Interspeech 2022
18-22 September 2022, Incheon, Korea

Enabling Off-the-Shelf Disfluency Detection and Categorization for Pathological Speech

Amrit Romana[1], *Minxue Niu*[1], *Matthew Perez*[1], *Angela Roberts*[2], *Emily Mower Provost*[1]

[1]Computer Science and Engineering, University of Michigan, Ann Arbor, Michigan, USA
[2]Communication Sciences and Disorders, Northwestern University, Evanston, Illinois, USA

aromana@umich.edu, sandymn@umich.edu, mkperez@umich.edu,
angela.roberts@northwestern.edu, emilykmp@umich.edu

Abstract

A speech disfluency, such as a filled pause, repetition, or revision, disrupts the typical flow of speech. Disfluency modeling has grown as a research area, as recent work has shown that these disfluencies may help in assessing health conditions. For example, for individuals with cognitive impairment, changes in disfluencies may indicate worsening symptoms. However, work on disfluency modeling has focused heavily on detection and less on categorization. Work that has focused on categorization has suffered with two specific classes: repetitions and revisions. In this paper, we evaluate how BERT (Bidirectional Encoder Representations from Transformers) compares to other models on disfluency detection and categorization. We also propose adding a second fine-tuning task where BERT learns to distance repetitions and revisions from their repairs with triplet loss. We find that BERT and BERT with triplet loss outperform previous work on disfluency detection and categorization, particularly for repetitions and revisions. In this paper we present the first analysis of how these models can be fine-tuned on widely available disfluency data, and then used in an off-the-shelf manner on small corpora of pathological speech.

Index Terms: pathological speech, disfluencies, BERT

1. Introduction

A speech disfluency is a disruption in the typical flow of speech. These disfluencies include inserting filler words, prolonging sounds, and repeating or revising portions of a sentence. Automatic disfluency detection and categorization has grown as an area of research for two primary reasons: 1) these disfluencies may inhibit natural language understanding applications such as with voice assistants [1–3] and 2) these disfluencies may highlight speaker characteristics such as age or health condition [4, 5]. In our paper, we focus on automatically detecting and categorizing disfluencies with clinical applications in mind.

In clinical practice, there is a need to quantify the number and types of disfluencies for individuals impacted by a range of conditions, including stuttering, Parkinson's disease, Alzheimer's disease, and aphasia [6–11]. These evaluations help clinicians assess current treatment and plan any necessary adjustments. However, these assessments require manual transcription by trained experts, which limits scalability and can introduce subjectivity [5]. An automated disfluency detection and categorization approach could provide clinicians with more fine-grained assessments of disfluencies in a range of settings, such as daily assessments of disfluencies in one's home. However, off-the-shelf tools that can provide both the detection and categorization of disfluencies do not currently exist.

The majority of previous work in automatic disfluency analysis has focused on detecting disfluencies, rather than categorizing them by type [12–15]. However, recent work by Riad et al. illustrated a method for detecting and categorizing disfluencies [16]. The authors introduced a set of hand-crafted features, and compared the use of these features in several models to detect filled pauses, repetitions, and revisions (examples in Table 1). In our preliminary analysis, we found that while these hand-crafted features performed well in detecting filled pauses, they underperformed in detecting repetitions and revisions. These categorizations are key in clinical applications [7, 17].

Table 1: *Examples of each type of disfluency class*

Filled pause (F)	um for most of my life I would..
Repetition (R)	and I get my message message across
Revision (V)	I don't think that think the goal...
Partial word (P)	...any kind of th therapy

Repetitions and revisions are collectively known as reparanda, and reparanda are typically followed by repairs. In a repetition, the repair matches the reparandum. In a revision, the reparandum differs from but often has lexical or semantic overlap with the repair. A key characteristic in disfluency analysis is that while the reparandum is disfluent, the repair is fluent. In this paper we leverage the relationship between reparanda and their repairs in order to improve disfluency detection.

In our work, we create language-focused disfluency detection and categorization models designed to be deployed in an off-the-shelf manner. We develop these models on the large-scale Switchboard annotated dataset of disfluencies [18] and evaluate them on two smaller disfluency datasets: one with adults who stutter [19] and another with individuals who have Parkinson's disease. Additionally, we augment the typical fine-tuning process for BERT [20], a pre-trained transformer for language understanding, to better detect reparanda. Specifically, we train BERT with triplet loss (TL) to push the word embeddings for reparanda and repairs further apart, while pulling embeddings for repairs and other fluent words closer together. We anticipate that this training step will help the downstream BERT classifier differentiate reparanda from fluent words. To the best of our knowledge, no previous work has applied TL, or any contrastive loss between reparanda and repairs, to disfluency detection. Overall, we find that BERT outperforms models that process hand-crafted features on each disfluency class. Additionally, we find that BERT with TL slightly outperforms BERT in detecting repetitions on both target datasets, and that it slightly outperforms BERT in detecting revisions on one target dataset. This work is an important step toward being able to automatically detect and identify disfluencies in pathological speech. In the long term, this work will lead to more fine-grained and scalable disfluency assessments for those with speech pathologies.

10.21437/Interspeech.2022-10971

2. Related work

Riad et al. address word-level disfluency detection and categorization in analyzing the Fluency Bank Adults Who Stutter (AWS) dataset (see Section 3) [16]. The authors introduce a set of hand-crafted features covering token characteristics, span information, and acoustics. They then show that these features in a simple support vector machine (SVM) classifier outperform more complex temporal models, such as Long Short-Term Memory (LSTM) models or Bidirectional-LSTMs (BLSTMs) previously used in the field [12, 21].

BERT is a language model that has been shown to be useful on a range of tasks including question answering, next sentence prediction, and sentence acceptability classification [20]. Researchers have also had success fine-tuning BERT on tasks that require learning relationships between tokens, such as part of speech tagging or named entity recognition [22]. However, BERT has not been widely explored on the task of disfluency detection. Rocholl et al. showed that fine-tuned BERT models achieve high accuracy in detecting disfluencies in the Switchboard dataset [15]. Additionally, Bach et al. show that a BERT model fine-tuned on Switchboard can achieve high accuracy on unseen but similarly structured data, such as a dataset of telephone conversations between family and close friends [14].

However, these works leave several open questions. First, can BERT categorize disfluency types, and how does its detection accuracy vary by type? Second, how do these accuracies by disfluency type compare to models trained with hand-crafted features? Finally, can BERT achieve high accuracy when tested on pathological speech where the distribution of disfluencies likely differs? In our paper we focus on addressing these questions. We also explore a fine-tuning technique specifically aimed at better detecting repetitions and revisions.

3. Data

We evaluate how disfluency detection and categorization models trained on Switchboard, a large-scale corpus, perform on two smaller datasets of pathological speech: a Fluency Bank dataset of stuttering experience assessments with adults who stutter (we refer to this as the AWS-Assessment dataset), and a dataset of picture description tasks from individuals with Parkinson's disease (PD-PicSeq dataset). Both of these datasets contain semi-structured speech with CHAT format transcripts. The CHAT format includes codings to indicate where and which disfluencies occur [23]. In the AWS-Assessment dataset, we analyze three classes of disfluencies: filled pauses, repetitions, revisions. In the PD-PicSeq dataset, we analyze an additional class: partial words. We do not include partial words for the AWS-Assessment dataset because these were not explicitly annotated. Table 1 shows examples of each disfluency class and Figure 1 shows the distribution in each dataset.

The AWS-Assessment dataset is a portion of Fluency Bank, an open source dataset used by clinical researchers to study fluency [19]. In the AWS-Assessment portion of the dataset, participants were recorded as they respond to questions in the Overall Assessment of the Speaker's Experience of Stuttering (OASES) elicitation protocol [5]. These data were manually transcribed by the data providers. We follow [16] and exclude participants whose annotations were lacking. In the end, we obtain 1,511 utterances from 22 speakers. Of these 1,511 utterances, 728 contain at least one disfluency. In our preliminary analysis (Section 4.1), we train and test models with these data. In our final analysis, we only use these data to test the off-the-shelf models that we train on the Switchboard corpus.

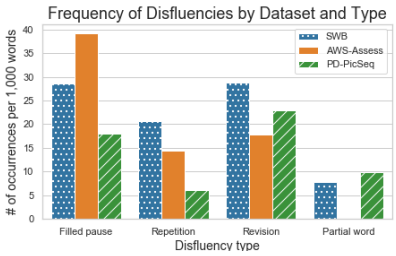

Figure 1: *Number of occurrences of each disfluency type in the Switchboard (SWB), AWS-Assessment, and PD-PicSeq datasets*

The PD-PicSeq dataset is a dataset of participants who have Parkinson's disease as they are recorded completing a picture sequence description task. These data were manually transcribed in the CHAT format by trained listeners in the Language & Communication in Aging and Neurodegeneration Research Group, led by Dr. Angela Roberts, at Northwestern University. This dataset includes 1,182 utterances from 37 participants. Of the 1,182 utterances, 303 contain at least one disfluency. In our preliminary analysis, we train and test models with these data. In our final analysis, we only use these data to test the off-the-shelf models that we train on the Switchboard corpus.

The Switchboard corpus, is a widely used telephone corpus with disfluencies labeled [18]. We work with the silver annotations provided by Zayats et al [24]. We apply a simple rule-based approach to convert the silver annotation disfluency labels to the label set in the pathological speech datasets. Specifically, we label occurrences of "um" and "uh" as filled pauses, we label words ending with "-" (such as "th-") as partial words, and given the provided reparandum and repair annotations, we label a reparandum as a repetition if it matches its repair and as a revision otherwise. While our silver annotation processing results in about 200k utterances, only 60k of these include disfluencies. We retain all sentences that have disfluencies and randomly sample an equal number of sentences that do not have any disfluencies. This step is supported by previous work which shows that undersampling fluent data can help with disfluency detection [16]. In the end, our Switchboard corpus has 114,324 utterances, where 57,162 contain at least one disfluency. We perform an 60/20/20 train/dev/test split, ensuring that disfluent samples are equally represented in each set.

4. Methodology

4.1. Preliminary Analysis

Zayats et al. and Riad et al. introduce similar sets of hand-crafted features for detecting disfluent words in the Switchboard (typical speech) and AWS-Assessment (pathological speech) datasets, respectively [12, 16]. These hand-crafted features include token characteristics (word embeddings and part of speech tags) and span information which captures each tokens similarity with nearby tokens. Zayats et al. trains a BLSTM to detect disfluencies, while Riad et al. illustrates that an SVM can outperform temporal models such as BLSTMs when working with pathological speech datasets.

In our preliminary analysis we extracted the token (320-dim) and span (76-dim) features introduced by Riad et al for the AWS-Assessment dataset [16]. We used these features to train SVMs in a leave-one-subject-out manner on the AWS-Assessment and PD-PicSeq datasets. We implement these mod-

Table 2: *F1-scores for detecting and categorizing disfluencies with an SVM trained on the AWS-Assessment and PD-PicSeq datasets, evaluated in a leave-one-subject-out manner. Note: O = outside of disfluency (fluent), D = disfluent(any), F = filled pauses, R = repetitions, V = revisions, P = partial words.*

Dataset	O	D	F	R	V	P
AWS-Assess	97.8	63.8	74.4	74.0	0.0	–
PD-PicSeq	98.0	50.8	83.6	0.0	0.0	67.0

els in Scikit-learn [25] with a linear kernel and C=0.025 as in [16]. Table 2 lists the F1-scores for detecting and categorizing each disfluency class across the two datasets. We found that this approach, while able to detect disfluencies (column D), often did not categorize them correctly. Revisions in particular (column V) are undetected, even though these compose the largest disfluency class for the PD-PicSeq data (see Figure 1).

These models struggle in part because of the small and imbalanced datasets. This motivates the use of larger corpora, with which we can explore deep networks. We evaluate the use of BLSTM and BERT models trained on the Switchboard corpus and evaluated on pathological speech.

4.2. BLSTM model with hand-crafted features

We implement a BLSTM with the hand-crafted token and span features from [16]. We train the model to detect and categorize disfluencies on our Switchboard train set, we and evaluate its performance on the pathological speech datasets. Our BLSTM architecture consists of 1 layer and 16 hidden states implemented in Pytorch [26]. We train the model with cross entropy loss, an Adam optimizer, a batch size of 32, a learning rate of 1e-3, and up to 20 epochs with early stopping (patience of 5) based on performance on the held-out Switchboard development set. We train the model with six different random initializations, and we evaluate the resulting models on the detection and categorization task with pathological speech.

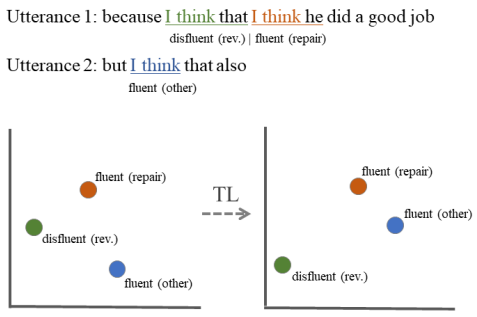

Figure 2: *Example of data for triplet loss and illustration of the resulting movement of embeddings. Both utterances are from the Switchboard training set. The first utterance contains a revision, where the revision and repair both include the phrase "I think." The second utterance is a randomly selected sentence that doesn't include disfluencies but does include the phrase "I think." Triplet loss penalizes the model if the embeddings for the fluent phrases are closer to the disfluent phrase than they are to each other. We anticipate that this training step will help the downstream BERT classifier differentiate disfluent and fluent words when they are lexically similar.*

4.3. BERT model

In this paper we compare the use of the hand-crafted features described above to BERT. We append a dense output layer to the BERT base uncased pre-trained model. We fine-tune the model on our Switchboard train set and evaluate how it performs in detecting and categorizing disfluencies in the pathological speech datasets. We implement the BERT model using PyTorch and the HuggingFace library [27]. We optimize the model's hyperparameters over three random seeds. Specifically, we use the Adam optimizer with cross entropy loss, set a batch size of 32, sweep over three learning rates (2e-5, 3e-5, 5e-5), and stop training after 2, 3, or 4 epochs. These hyperparameters were those suggested by the original BERT authors [20]. We select a final set of hyperparameters based on the average development loss, and we use this set to fine-tune BERT models over six new random seeds. This is a condensed version of the training using in previous analysis with BERT [28].

4.4. BERT model with triplet loss

We hypothesize that we can leverage the lexical overlap between reparanda and associated repairs in order to improve detection of repetitions and revisions. Specifically, we provide these reparanda and repairs as samples while we aim to minimize a cost function, triplet loss (TL). TL takes in three inputs (an anchor, a positive sample, and a negative sample), and aims to pull the positive sample toward the anchor while pushing a negative sample away from the anchor. This is accomplished with the following equation.

$$L(a, p, n) = max(d(a_i, p_i) - d(a_i, n_i) + margin, 0)$$

where d is a pairwise distance function, a is an anchor, p is a positive sample, n is a negative sample, and $margin$ defines how far away the dissimilarities need to be. We set $margin = 1$ as in previous work that used TL for BERT [29]. For our task we also enable $swap$ which specifies that, if the positive sample is closer to the negative sample than the anchor is, the role of the anchor and positive sample is reversed, so that the negative sample is pushed away from the positive sample.

Within our Switchboard training set we define triplet samples as follows. For each reparanda, we identify the lexical overlap between that reparanda and its repair. For example, in the sentence "because I think that I think he did a good job," the reparandum is "I think that," the repair is "I think he," and the overlapping text is "I think." We define the "I think" in the reparanda to be the negative sample, and the "I think" in the repair to be the anchor. We then randomly select a fluent sentence from our training set that has this phrase (for example "But I think that also") and define that "I think" as our positive sample. With this approach we define 17,697 triplets. During fine-tuning, we pass the two sentences associated with each triplet as inputs to our model, separate the mean embeddings associated with the anchor, positive, and negative samples, calculate TL, and then update the embeddings accordingly. The goal of this step is to teach the model that even though the words within the anchor, positive, and negative sample are the same, based on the context in which they appear, the anchor (repair) and positive (other fluent) embeddings should be closer to each other than to the negative (disfluent reparandum) embedding. In future work, we will explore expanding our triplet set to include lexical differences between the anchor, positive, and negative samples (such as the full phrases "I think that" and "I think he"). We summarize our current process in Figure 2.

Table 3: *F1-score for detecting and categorizing disfluencies in our off-the-shelf experiments. We train these models on the Switchboard corpus, and then evaluate how these models perform, without any additional adaptation, on the AWS-Assessment and PD-PicSeq datasets. Results shown are the mean and standard deviation of F1-scores across 6 random seeds. The best results for each class are in bold. Note: O = outside of disfluency (fluent), D = disfluent (any), F = filled pause, R = repetition, V = revision, P = partial word.*

Dataset	Model	O	D	F	R	V	P
AWS-Assessment	BLSTM	97.4 (0.1)	69.9 (0.4)	86.9 (0.3)	56.9 (1.3)	24.7 (0.9)	–
	BERT	**98.1** (0.0)	77.9 (0.4)	**95.3** (0.0)	69.6 (1.2)	43.2 (0.1)	–
	BERT w/TL	**98.1** (0.0)	**78.2** (0.4)	**95.3** (0.0)	**69.7** (0.8)	**44.1** (1.1)	–
PD-PicSeq	BLSTM	98.6 (0.1)	71.8 (0.5)	86.1 (0.4)	41.5 (1.2)	39.4 (2.7)	66.8 (0.7)
	BERT	**99.2** (0.0)	**85.9** (0.4)	**87.3** (0.0)	74.7 (0.9)	**65.8** (1.0)	**84.7** (0.1)
	BERT w/TL	**99.2** (0.0)	85.2 (0.4)	**87.3** (0.0)	**75.3** (2.2)	64.8 (0.9)	83.5 (0.7)

TL has previously been used to fine-tune BERT embeddings in [29], but to the best our our knowledge it has not been explored in disfluency analysis. Following [29], we fine-tune our model with TL for one epoch with a learning rate of 2e-5. We then append the dense output classification layer, evaluate hyperparameters over three random seeds, and fine-tune BERT over six random seeds as described in the previous section. Based on our results with the Switchboard development set, we find that Baseline BERT performs best with a learning rate of 5e-5 over 2 epochs, and BERT with TL (BERT w/TL) performs best with a learning rate of 3e-5 over 2 epochs.

5. Results and discussion

Table 3 lists the F1-scores resulting from our off-the-shelf experiments. We find that these off-the-shelf models outperform the within-dataset model presented in Table 2 on all disfluecy types except repetitions in the AWS-Assessment data. Of the off-the-shelf models, BERT generalizes best to both AWS-Assessment and PD-PicSeq. In terms of the different disfluency types, BERT primarily boosts the detection of repetitions and revisions. For the AWS-Assessment data, BERT detects repetitions with an F1-score of 69.6 (compared to 56.9 with BLSTM), and for the PD-PicSeq data, this F1-score increases to 74.7 (compared to 41.5 with BLSTM). For the AWS-Assessment data, BERT detects revisions with an F1-score of 43.2 (compared to 24.7 with BLSTM), and for the PD-PicSeq data this F1-score increases to 65.8 (compared to 39.4 with BLSTM).

Overall, BERT w/TL performs similarly to BERT. We hypothesized that TL would help BERT detect revisions and repetitions and we see slight improvements for these classes. Most notably, in the AWS-Assessment dataset, the F1-score for detecting revisions increases from 43.2 with BERT to 44.1 with BERT w/TL. In the PD-PicSeq dataset, the F1-score for detecting repetitions increases from 74.7 with BERT to 75.3 with BERT w/TL. However, in both of these classes the standard deviation of the F1-score also increases. We use McNemar's test with the Bonferroni correction to evaluate whether these difference are significant. We find that BERT w/TL significantly outperforms BERT on detecting reparanda in half of the runs with the AWS-Assessment dataset (p-value < 0.1), but that there is no significant difference between BERT and BERT w/TL in the PD-PicSeq dataset. This is likely due in part to the differences in reparanda for each dataset which we explore next.

Across the datasets, we find that each model performs worse in detecting revisions in the AWS-Assessment dataset compared to the PD-PicSeq dataset. We explore which characteristics may be driving this change in performance. We do not find a noticeable difference in the length of revisions across these datasets: in each dataset roughly 40% contain 1 word, 25% contain 2 words, and the rest are longer. We also do not find a noticeable difference in the parts of speech found within the revisions: in each dataset, nouns, prepositions, and determiners are the most commonly found in revisions. We do find the revisions within the AWS-Assessment dataset are more commonly followed by filled pauses, which may impede the detection of the revision. We also find that the fraction of overlapping words between a revision and repair tend to be higher in the AWS-Assessment dataset compared to the PD-PicSeq dataset (45% versus 37%, respectively). This may make it difficult for the models to differentiate between when a disfluent word is in a repetition versus revision. This may also explain why BERT with TL, which was fine-tuned with overlapping reparanda and repair samples, improves the detection of revisions in the AWS-Assessment dataset but not the PD-PicSeq dataset. In future work we will aim to include a broader set of triplet samples for TL, where the anchors, positives, and negatives have some lexical variation to capture a wider set of revisions.

6. Conclusions

In this paper we present the first analysis of disfluency detection and categorization models designed to be deployed in an off-the-shelf manner with pathological speech. We compare the use of a BLSTM, BERT, and BERT with an additional TL training step. We find that BERT and BERT w/TL perform best across all disfluency types, and that BERT w/TL performs better than BERT in detecting reparanda in one of the two datasets with which we evaluated these models. In the long term, these works can lead to more fine-grained and scalable disfluency assessments for pathological speech.

7. Acknowledgements

This work was funded in part by Precision Health at University of Michigan. Funding for the data was provided by a National Institute of Deafness and Communication Disorders (NIDCD) grant to Dr. Angela Roberts (NIH/NIDCD 1R21DC017255-01). The PD-PicSeq data collection and analysis were approved by and conducted in accordance with current human subjects ethics guidelines at Northwestern University (PI: Dr. Angela Roberts). We thank the following people who were involved in the manual aspects of the annotations: Stephanie Gutierrez, Brenda Xu, Abigael Parrish, and Richard Richter. The PD-PicSeq data were made available for this project through a data use agreement executed between Dr. Angela Roberts (Northwestern University) and Dr. Emily Mower Provost (University of Michigan).

8. References

[1] K. Arjun, S. Karthik, D. Kamalnath, P. Chanda, and S. Tripathi, "Automatic correction of stutter in disfluent speech," *Procedia Computer Science*, vol. 171, pp. 1363–1370, 2020.

[2] V. Mitra, Z. Huang, C. Lea, L. Tooley, S. Wu, D. Botten, A. Palekar, S. Thelapurath, P. Georgiou, S. Kajarekar *et al.*, "Analysis and tuning of a voice assistant system for dysfluent speech," *arXiv preprint arXiv:2106.11759*, 2021.

[3] C. Lea, V. Mitra, A. Joshi, S. Kajarekar, and J. P. Bigham, "Sep-28k: A dataset for stuttering event detection from podcasts with people who stutter," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 6798–6802.

[4] H. Bortfeld, S. D. Leon, J. E. Bloom, M. F. Schober, and S. E. Brennan, "Disfluency rates in conversation: Effects of age, relationship, topic, role, and gender," *Language and speech*, vol. 44, no. 2, pp. 123–147, 2001.

[5] J. S. Yaruss and R. W. Quesal, "Overall assessment of the speaker's experience of stuttering (oases): Documenting multiple outcomes in stuttering treatment," *Journal of fluency disorders*, vol. 31, no. 2, pp. 90–115, 2006.

[6] J. S. Yaruss, "Clinical measurement of stuttering behaviors," *Contemporary Issues in Communication Science and Disorders*, vol. 24, no. Spring, pp. 27–38, 1997.

[7] A. M. Goberman and M. Blomgren, "Parkinsonian speech disfluencies: effects of l-dopa-related fluctuations," *Journal of fluency disorders*, vol. 28, no. 1, pp. 55–70, 2003.

[8] A. M. Goberman, M. Blomgren, and E. Metzger, "Characteristics of speech disfluency in parkinson disease," *Journal of Neurolinguistics*, vol. 23, no. 5, pp. 470–478, 2010.

[9] A. Romana, J. Bandon, M. Perez, S. Gutierrez, R. Richter, A. Roberts, and E. M. Provost, "Automatically detecting errors and disfluencies in read speech to predict cognitive impairment in people with parkinson's disease," in *22nd Annual Conference of the International Speech Communication Association, INTERSPEECH 2021*. International Speech Communication Association, 2021, pp. 156–160.

[10] K. López-de Ipiña, U. Martinez-de Lizarduy, P. M. Calvo, B. Beitia, J. García-Melero, E. Fernández, M. Ecay-Torres, M. Faundez-Zanuy, and P. Sanz, "On the analysis of speech and disfluencies for automatic detection of mild cognitive impairment," *Neural Computing and Applications*, vol. 32, no. 20, pp. 15 761–15 769, 2020.

[11] C. C. Oomen, A. Postma, and H. H. Kolk, "Speech monitoring in aphasia: Error detection and repair behaviour in a patient with broca's aphasia," in *Phonological encoding and monitoring in normal and pathological speech*. Psychology Press, 2005, pp. 221–237.

[12] V. Zayats, M. Ostendorf, and H. Hajishirzi, "Disfluency detection using a bidirectional lstm," *Interspeech 2016*, pp. 2523–2527, 2016.

[13] P. J. Lou, P. Anderson, and M. Johnson, "Disfluency detection using auto-correlational neural networks," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 4610–4619.

[14] N. Bach and F. Huang, "Noisy bilstm-based models for disfluency detection." in *INTERSPEECH*, 2019, pp. 4230–4234.

[15] J. C. Rocholl, V. Zayats, D. D. Walker, N. B. Murad, A. Schneider, and D. J. Liebling, "Disfluency detection with unlabeled data and small bert models," *arXiv preprint arXiv:2104.10769*, 2021.

[16] R. Riad, A.-C. Bachoud-Lévi, F. Rudzicz, and E. Dupoux, "Identification of primary and collateral tracks in stuttered speech," in *LREC 2020-12th Conference on Language Resources and Evaluation*, 2020.

[17] G. D. Riley, "A stuttering severity instrument for children and adults," *Journal of speech and hearing disorders*, vol. 37, no. 3, pp. 314–322, 1972.

[18] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "Switchboard: Telephone speech corpus for research and development," in *Acoustics, Speech, and Signal Processing, IEEE International Conference on*, vol. 1. IEEE Computer Society, 1992, pp. 517–520.

[19] N. B. Ratner and B. MacWhinney, "Fluency bank: A new resource for fluency research and practice," *Journal of fluency disorders*, vol. 56, pp. 69–80, 2018.

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[21] T. Tran, S. Toshniwal, M. Bansal, K. Gimpel, K. Livescu, and M. Ostendorf, "Parsing speech: a neural approach to integrating lexical and acoustic-prosodic information," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 69–81.

[22] I. Tenney, D. Das, and E. Pavlick, "Bert rediscovers the classical nlp pipeline," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 4593–4601.

[23] B. MacWhinney, *The CHILDES Project: Tools for analyzing talk. transcription format and programs*. Psychology Press, 2000, vol. 1.

[24] V. Zayats, T. Tran, R. Wright, C. Mansfield, and M. Ostendorf, "Disfluencies and human speech transcription errors," *Proc. Interspeech 2019*, pp. 3088–3092, 2019.

[25] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[26] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[27] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020, pp. 38–45.

[28] T. Zhang, F. Wu, A. Katiyar, K. Q. Weinberger, and Y. Artzi, "Revisiting few-sample bert fine-tuning," in *International Conference on Learning Representations*, 2020.

[29] S. Wang, L. Thompson, and M. Iyyer, "Phrase-bert: Improved phrase embeddings from bert with an application to corpus exploration," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 10 837–10 851.