applied sciences

MDPI

Article

Automatic Speech Disfluency Detection Using wav2vec2.0 for Different Languages with Variable Lengths

Jiajun Liu [1,2], Aishan Wumaier [2,3,*], Dongping Wei [2,3] and Shen Guo [2,3]

[1] College of Software, Xinjiang University, Urumqi 830046, China; liujiajun@stu.xju.edu.cn
[2] Key Laboratory of Multilingual Information Technology in Xinjiang Uyghur Autonomous Region, Urumqi 830046, China; waooo@stu.xju.edu.cn (D.W.); guoshen@stu.xju.edu.cn (S.G.)
[3] College of Information Science and Engineering, Xinjiang University, Urumqi 830046, China
* Correspondence: hasan1479@xju.edu.cn

Abstract: Speech is critical for interpersonal communication, but not everyone has fluent communication skills. Speech disfluency, including stuttering and interruptions, affects not only emotional expression but also clarity of expression for people who stutter. Existing methods for detecting speech disfluency rely heavily on annotated data, which can be costly. Additionally, these methods have not considered the issue of variable-length disfluent speech, which limits the scalability of detection methods. To address these limitations, this paper proposes an automated method for detecting speech disfluency that can improve communication skills for individuals and assist therapists in tracking the progress of stuttering patients. The proposed method focuses on detecting four types of disfluency features using single-task detection and utilizes embeddings from the pre-trained wav2vec2.0 model, as well as convolutional neural network (CNN) and Transformer models for feature extraction. The model's scalability is improved by considering the issue of variable-length disfluent speech and modifying the model based on the entropy invariance of attention mechanisms. The proposed automated method for detecting speech disfluency has the potential to assist individuals in overcoming speech disfluency, improve their communication skills, and aid therapists in tracking the progress of stuttering patients. Additionally, the model's scalability across languages and lengths enhances its practical applicability. The experiments demonstrate that the model outperforms baseline models in both English and Chinese datasets, proving its universality and scalability in real-world applications.

Keywords: speech disfluency detection; stuttering; limited data; wav2vec2.0; entropy invariance

Citation: Liu, J.; Wumaier, A.; Wei, D.; Guo, S. Automatic Speech Disfluency Detection Using wav2vec2.0 for Different Languages with Variable Lengths. *Appl. Sci.* **2023**, *13*, 7579. https://doi.org/10.3390/app13137579

Academic Editor: Dong Wang and Andrew Abel

Received: 17 May 2023
Revised: 17 June 2023
Accepted: 24 June 2023
Published: 27 June 2023

1. Introduction

As an important means of communication for human beings in terms of exchanging ideas, expressing emotions, and transmitting information, speech has driven the development of human civilization and social change. In daily life, speech is essential for normal communication between people, and it has been proven to be the most efficient and widely used method of communication. Generally, three basic dimensions are used to measure a person's oral expression ability, which include accuracy, clarity, and fluency. Fluency determines the ability of speech to convey messages. Fluency is defined by the normal flow of speech [1], which connects different phonemes to generate information. Fluency reflects the speaker's language organization and conversion ability and more directly reflects the true level of oral expression. Continuity, rate, rhythm, and effort are four basic factors used to distinguish between fluent and disfluent speech [2]. Continuity refers to the logical connectivity of message transmission. Rate is the speed of message delivery. Rhythm and effort, respectively, refer to the regularity and energy of sound in the message delivery process. However, not everyone has normal oral expression ability, and speech disfluency often occurs in communication. There are two main types of speech disfluency: normal disfluency and stuttering, which are, respectively, targeted at the normal population and the

stuttering population. Normal disfluency refers to the situation of speech disfluency that exists in people's daily lives, such as interruptions, pauses, or repetitions, which are related to people's emotional excitement, anxiety, or nervousness. Its main characteristics include phrase repetition, syllable repetition of polysyllabic words, and the use of interjections, rephrasing, and repeated revisions. This situation is a common phenomenon in daily life and does not affect daily communication and emotional expression. Speech disfluency also occurs in children, but as children grow older and develop stronger expressive abilities, this disfluency tends to improve. Stuttering [3] is a language disorder characterized by symptoms such as speech disfluency, repetition or prolongation of syllables, pauses, and difficulty organizing language. The World Health Organization defines stuttering as a disorder of speech rhythm [4] in which the stutterer cannot express themselves clearly due to involuntary repetition, prolongation, or interruption of sounds. According to epidemiological survey data, there are currently over 70 million stuttering patients worldwide, with a far greater proportion of male patients than female patients [5,6]. Therefore, stuttering is an important chronic speech disorder that deserves widespread attention.

Therefore, if normal disfluency is present, it is necessary to comprehensively improve oral expression ability. Speakers may participate in speaking exams, such as the TOEFL iBT speaking test for English or the Mandarin Proficiency Test (Putonghua Shuiping Ceshi, PSC) for Chinese, to improve their oral abilities, where fluency is an important aspect of the exam. However, the assessment of fluency, especially for spontaneous speech expression tasks, is still manually scored, which is highly subjective and requires a significant amount of manpower. Furthermore, these assessments usually only return a corresponding fluency level or score without indicating specific disfluency features, making it difficult for speakers to receive timely feedback and make corrections. If stuttering is present, it needs to be taken seriously, and the individual should be trained to reshape fluency to reduce the impact of stuttering on their life and psychology. Speech-language pathologists diagnose stutterers and manually measure their fluency, calculate the incidence of disfluency, and evaluate the stutterer's response throughout the entire treatment process. However, assessments of stuttering of this nature may be subjective, lack consistency, and be susceptible to errors. This paper suggests that an automatic method is needed to detect speech disfluencies, which can help people overcome speaking difficulties, improve their language expression abilities, and assist therapists in tracking the progress of stuttering patients.

Existing automated methods for detecting speech disfluency require large-scale disfluency-labeled speech data during training and are designed for specific languages, making it difficult to extend their use to the detection of speech disfluency in other languages. Additionally, in practical detection scenarios, the length of disfluent speech is not fixed. However, most existing research on fixed-length speech disfluency detection has overlooked the detection of variable-length speech disfluency. In recent studies, researchers have started to consider building disfluency detection models with limited data and have attempted to use pre-training and fine-tuning models for disfluency detection. However, these methods still have many limitations. For example, the latest models are still designed and trained for specific languages and dialects, making it difficult to detect disfluencies in other languages or dialects. Moreover, these models may not generalize well to other datasets or contexts. Additionally, many of the latest models are based on fixed-length modeling, which may not be suitable for detecting disfluencies of varying lengths. Furthermore, many of the latest studies still adopt simple machine learning methods, although the use of deep learning methods may potentially achieve better results in disfluency detection given the recent advances in deep learning in the field of speech. In summary, while there have been some recent advances in automatic speech disfluency detection, there are still several limitations and challenges that need to be addressed. To address these issues, this study proposes a disfluency detection method designed for limited disfluent speech data. The method extracts disfluency features from the wav2vec2.0 model for disfluency detection. In the model construction process, the entropy invariance of attention mechanisms is used to enable the model to generalize to disfluent speech of different lengths, allowing for variable-length

speech disfluency detection. Finally, this paper conducts experiments on both an English open-source database and a self-built database for Chinese and finds that the model can be applied in different language environments with strong scalability.

In summary, the main contributions of this paper can be summarized as follows:

- This paper addresses the shortage of Chinese disfluent speech data by creating the PSC-PS-DF dataset, which includes four disfluent features: interjections, blocks, prolongations, and repetitions.
- A classification network is developed in this paper for automated speech disfluency detection by combining CNN and Transformer and utilizing context embeddings from the pre-training model, wav2vec2.0. The network outperforms the baseline model in terms of detection accuracy and training time, even when trained with limited data.
- Considering that the length of disfluent speech data varies in practical detection scenarios, this paper improves the model based on the entropy invariance of attention mechanisms, allowing the model's results to generalize to speech data of different lengths, which means that even if the training and testing data have different disfluent data lengths, the model can still achieve good results.
- To ensure that the proposed model can achieve good disfluency detection results in different language environments, this paper conducts experiments on the self-built PSC-PS-DF dataset for Chinese and the open-source SEP-28k dataset for English disfluent speech. The results demonstrate the potential of the proposed model to detect speech disfluency in various language environments.

The paper is structured as follows. Section 2 provides an overview of related work, while Section 3 presents the proposed method, including the model architecture and the entropy invariance of attention mechanisms. Section 4 describes the dataset and the experimental setup, while Section 5 analyzes the experimental results. Finally, Section 6 concludes the paper.

2. Related Work

Speech disfluency detection models are generally based on the concept of speech recognition systems, which involve extracting speech features and classifying speech samples as fluent or disfluent. Speech disfluency detection models typically have three stages: preprocessing, speech feature extraction, and feature classification [7]. The preprocessing stage involves preprocessing the raw speech signal to prepare for subsequent feature extraction and classification. The speech feature extraction stage involves converting the preprocessed speech signal into a set of feature vectors for subsequent classification. Speech signals are time-domain signals that can be analyzed in the time domain and frequency domain to extract features. Feature classification involves classifying the extracted speech feature vectors into two categories: fluent and disfluent.

For speech disfluency detection, the focus is mainly on the extraction of disfluent features and the classification of disfluent speech, which involves several main categories of disfluent features, including repetitions, prolongations, interjections, and blocks. Different feature extraction and classification techniques for speech disfluency detection have been introduced in the literature [1,8–11], and the accuracy of different methods has been compared and analyzed. Many of these studies were conducted on non-public datasets or public datasets with non-public annotations [12,13]. There are now two publicly available datasets with public annotations: the stuttering events in podcasts SEP-28k dataset [14] and the KSoF dataset [15]. In the following sections, we will provide a detailed introduction to the related work on feature extraction and disfluent speech classification tasks.

2.1. Feature Extraction

Common disfluent speech features in existing speech disfluency detection methods include the Mel frequency cepstral coefficient (MFCC) [16–24], linear predictive coding (LPC) [25–27], the linear prediction cepstral coefficient (LPCC) [20,28], perceptual linear prediction (PLP) [23,27], and spectrograms [29–31]. In recent years, wav2vec features have

shown strong performance in multiple speech tasks, such as speech recognition, speech emotion recognition, and mispronunciation detection [32–34]. As an emerging speech feature extraction method, wav2vec2 is capable of learning richer and more abstract speech representations and has also been widely used in speech disfluency detection [35–39].

2.2. Disfluency Detection with Machine Learning

In 1995, Howell et al. [40] proposed using the autocorrelation function and envelope parameters as input features and constructed an artificial neural network (ANN) to detect disfluencies in repetitions and prolongations. In subsequent research, many scholars have also used ANNs as classifiers for identifying stuttering events in speech [41,42]. In recent years, many researchers have used an increasing number of machine learning models as classifiers for detecting disfluency events in speech, such as hidden Markov models (HMM) [16,26], support-vector machines (SVM) [17,21,27,43], k-nearest neighbors (KNN) [18,20,22,27,28], linear discriminant analysis (LDA) [18,20,27,28], dynamic time warping (DTW) [19,44], and multilayer perceptrons (MLP) [45,46]. When using machine learning models for stuttering event classification, it is generally necessary to manually design some features to represent different aspects of speech, such as the spectrum, energy, and speaking rate of the audio. Then, these features are used as inputs, passed to a machine learning algorithm, and the algorithm learns how to map the input features to different stuttering event categories, thereby training the classifier.

2.3. Disfluency Detection with Deep Learning

With the promising progress of deep learning in speech-related fields, such as speech recognition and emotion classification, the application of DL in disfluency detection has also increased in recent years. Zayats et al. [47] used a bidirectional long short-term memory neural network (BLSTM) model for speech disfluency detection in 2016. The model not only used word sequences but also included pattern matching features as inputs and used integer linear programming in the final output to combine the constraints of the network structure, achieving advanced performance. In their 2019 work on the same dataset, Zayats et al. [48] improved the speech disfluency detection model by using BLSTM with prosodic cues and achieved better classification results. In the same year, Santoso et al. [49] applied BLSTM and attention mechanisms to weight each frame based on its importance rather than directly measuring the overall information from the speech, resulting in more accurate classification. Kourkounakis et al. [29] used spectrograms as features and achieved high accuracy in speech disfluency classification with a BLSTM model. Wang et al. [50] proposed a method that combines multiple self-supervised tasks and fine-tunes a pre-trained network using labeled disfluency detection training data, achieving good detection results with limited data. Chen et al. [51] proposed a controllable time-delay Transformer (CT-Transformer) model that jointly performs punctuation prediction and disfluency detection tasks. The experimental results showed that the method outperformed existing state-of-the-art models in terms of F-score and achieved a competitive inference speed. Sheikh et al. [24] proposed a novel deep learning-based stutter detection model, StutterNet, which uses time-delay neural network (TDNN) to capture disfluent speech in a contextual aspect and significantly reduces the number of training parameters while obtaining reliable results. Mohapatra et al. [35] constructed a DisfluencyNet network based on contextual embeddings of the wav2vec2.0 model. The main building blocks of this network are convolution layers with max-pooling and fully connected layers. The experiment showed that this method trained a powerful network within just a few minutes of data and achieved excellent disfluency detection results. Al-Banna et al. [31] proposed a new stutter event detection model based on log melspectrograms and a 2D atrous CNN . The experimental results showed that the model outperformed the state-of-the-art methods in prolongation. In general, deep learning models have better performance than traditional machine learning models because they can directly learn features from raw speech signals in an end-to-end

manner and can use more complex models to model the time and frequency information in speech signals, resulting in more accurate training of speech disfluency detection classifiers.

In summary, most of the existing work on disfluency detection in speech focuses on English datasets, and few studies have validated the effectiveness of the models on datasets in different languages. Additionally, the existing work mostly includes experiments on large-scale datasets and is only suitable for detecting disfluencies of fixed lengths, while the length of disfluent speech data is not fixed. Therefore, there is a lack of a disfluency detection model that is applicable to different languages and variable-length disfluent speech data. In this paper, we have constructed a disfluency detection model combining wav2vec2.0 , a CNN, and a Transformer, which is inspired by [35]. The model can achieve good results in different language scenarios with minute-level data training, and the entropy invariance of attention mechanisms is employed to enable the model to generalize to disfluent speech of different lengths, making it more applicable to real-world disfluency detection scenarios.

3. Proposed Method

In this section, we introduce the network architecture of the disfluency detection model shown in Figure 1, which mainly consists of three modules: wav2vec2.0 , a CNN, and a Transformer. In the Transformer module, we introduce a length-scaling factor when computing the attention matrix and demonstrate from an entropy perspective the model's ability to generalize to disfluent speech of different lengths. This makes the model applicable for disfluency detection tasks of varying speech lengths.

3.1. Model Architecture

This paper proposes a disfluency detection network designed for disfluency classification using limited, disfluent speech data. The paper uses the wav2vec2.0 model to extract contextual speech embeddings from raw audio and achieves good disfluency classification results with limited disfluent data. As shown in Figure 1, the contextual representation of the speech input obtained through the wav2vec2.0 model is then used to more effectively extract disfluency features through a CNN and Transformer model, and finally, a fully connected layer is used to obtain binary classification results for fluent and disfluent speech. Specifically, the disfluency detection network first employs a single-layer CNN to extract specific disfluent features from speech signals, which are then passed to the Transformer layer for further processing. The subsequent sections describe in detail the structure of the proposed disfluency detection network.

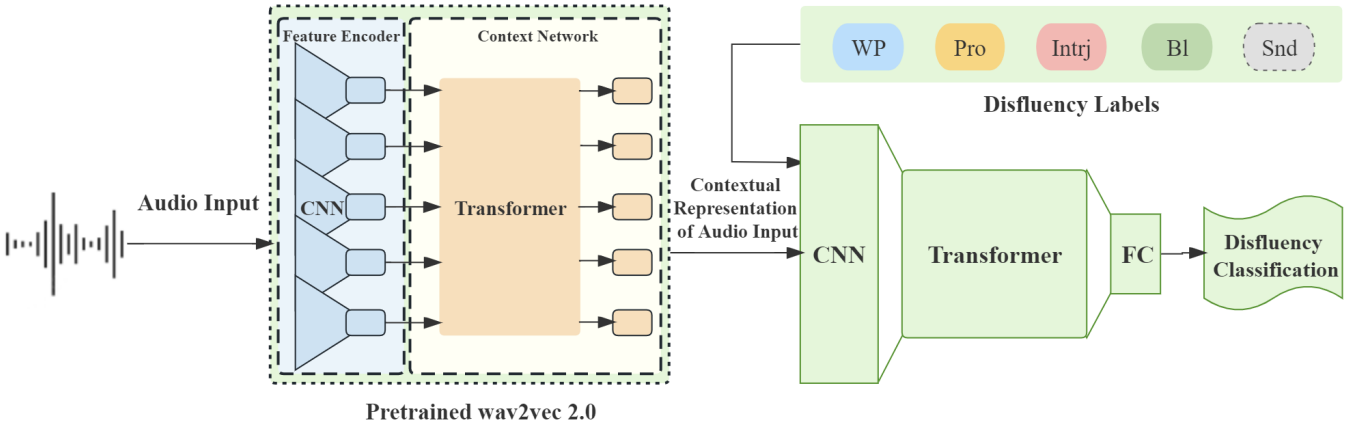

Figure 1. The architecture of the proposed model.

First, the raw audio is input into the pre-trained wav2vec2.0 model, obtaining the audio feature representation of the last hidden layer of the wav2vec2.0 model. As shown

in Formula (1), where $X \in R^{L \times d}$ represents the raw speech signal, L represents the length, and d represents the dimension. $f_{\text{wav2vec2.0}}$ represents the pre-trained wav2vec2.0 model.

$$Z = f_{\text{wav2vec2.0}}(X) \tag{1}$$

Afterwards, local features can be extracted through the CNN by sliding convolution kernels, and these feature representations contain local structures and speech features in the audio, which can be further used for disfluency detection tasks. Specifically, 768-dimensional audio features are inputted into the CNN network structure, which has 50 output channels. The CNN layer applies 50 different filters to the input, each with a width of 2 and with no padding added to the input, producing 50 output feature maps. These feature maps reflect disfluent information in speech signals, such as repetitions, blocks, and prolongations. After the CNN, the obtained features are fed into the Transformer model for more effective feature extraction. The encoder part of the Transformer can be seen as a combination of a multi-head self-attention mechanism and a feed-forward neural network. These network layers can effectively learn the relationships and sequence information between features, generating a more abstract and high-level representation. This part has an input vector dimension of 50 and is composed of two stacked Transformer encoder layers. Each Transformer encoder layer uses the same set of parameters, including 10 multi-head self-attention mechanisms to capture different semantic information.

$$\hat{Z} = \text{Transformer}(\text{ConvlD}(Z)) \tag{2}$$

Finally, the output of the encoder is classified through a fully connected layer to obtain the results of the disfluency classification.

Due to the denser and more complex nature of speech data compared to text data, the combined structure of wav2vec2.0 + CNN + Transformer can fully explore more details and complexities in the speech signal, effectively extracting and classifying speech features. This approach can improve the performance of disfluency detection models and has a certain level of generality and applicability.

3.2. Entropy Invariance of Attention Mechanisms

By using attention mechanisms, it becomes possible to identify the most relevant segments of input data and concentrate on them. In the scaled dot-product attention mechanism, the scaling operation is used to ensure the entropy invariance of the attention distribution. This is achieved by multiplying the attention distribution by a scaling factor to adjust its magnitude, thus better controlling the learning efficiency and stability of the model. In the scaled dot-product attention, the input sequence is first mapped to query the vector Q, key vector K, and value vector V. The dot product of Q and K is then calculated and scaled, and the scaled result is weighted and averaged with V to obtain the output of the self-attention mechanism. Specifically, assuming there are n input vectors, each with dimension d, the formula for calculating the scaled dot-product attention is

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V \tag{3}$$

The softmax function is used to normalize the attention distribution into a probability distribution. $\sqrt{d}$ is the scaling factor used to prevent the dot-product result from being too large or too small, keeping the attention distribution within a suitable range and avoiding over-concentration, thus improving the model's robustness and generalization ability. In this paper, we introduce a method to maintain attention entropy invariance by using the scaling operation in the attention mechanism in the Transformer model. In Transformer, each attention head calculates attention for the input, and the results of multiple heads are fused to obtain the final output. During attention calculation, scaling the attention vector

can maintain the entropy invariance of the attention distribution, better controlling the size and uniformity of the attention distribution.

This paper addresses the shortage of disfluent speech data resources and the issue that the length of disfluent speech in practical scenarios is not fixed. We propose a model optimization method based on attention mechanism entropy invariance, which enables the model to generalize to speech data of different lengths, and achieve good results even when the length of fine-grained disfluent speech data used for training and testing is different. In this section, we optimize the model using attention mechanism entropy invariance to enhance its extrapolation ability, that is, the ability to infer the whole from the part, and enable a model trained on shorter disfluent speech data to be tested on longer data sequences without fine-tuning, achieving good results. Optimizing the extrapolation ability of the model length not only solves the problem of inconsistent lengths during training and prediction but also improves the model's generalization ability. Moreover, it allows an effective model to be trained on shorter data for use on longer data when disfluent speech data resources are limited, thus improving the practicality of the model.

This paper references previous research on the entropy invariance of attention mechanisms [52–54] and, based on this perspective, redefines a new scaling factor for extrapolating length and introduces a constant λ to improve the length generalization ability of the attention mechanism. This enables the model to perform better in length extrapolation without changing its existing training performance. Information entropy is a measure of uncertainty, and in research, it is necessary to minimize uncertainty, i.e., minimize entropy. In our study, this uncertainty is considered the concentration of attention. Generally, training on longer data requires attention to be evenly distributed among all parts, making the attention more dispersed and resulting in greater entropy. Conversely, training on shorter data leads to more concentrated attention, resulting in smaller entropy. Entropy invariance mainly refers to reducing the correlation between this uncertainty and length, so that regardless of how the length changes, the entropy remains essentially unchanged. From the perspective of entropy invariance, the formula for scaled dot-product attention can be rewritten as Formula (4).

$$a_{ij} = \frac{e^{\lambda q_i \cdot k_j}}{\sum_{j=1}^{n} e^{\lambda q_i \cdot k_j}} \tag{4}$$

where q_i and k_j represent the i-th query vector and j-th key vector, respectively, of the input sequence, and $q_i \cdot k_j$ is the dot product of the two vectors, which reflects the correlation between them. λ is the scaling factor, which is independent of q_i and k_j. a_{ij} is the conditional distribution of a random variable, and the expression of entropy is shown in Formula (5).

$$\mathcal{H}_i = -\sum_{j=1}^{n} a_{ij} \log a_{ij} \tag{5}$$

Substituting a_{ij} into Formula (5) yields Formula (6).

$$\begin{aligned} \mathcal{H}_i &= -\sum_{j=1}^{n} a_{i,j} \log a_{i,j} \\ &= \log \sum_{j=1}^{n} e^{\lambda q_i \cdot k_j} - \lambda \sum_{j=1}^{n} a_{ij} e^{\lambda q_i \cdot k_j} \\ &= \log n + \log \frac{1}{n} \sum_{j=1}^{n} e^{\lambda q_i \cdot k_j} - \lambda \sum_{j=1}^{n} a_{ij} e^{\lambda q_i \cdot k_j} \end{aligned} \tag{6}$$

Formula (6) comprises three terms, with the $\log \frac{1}{n} \sum_{j=1}^{n} e^{\lambda q_i \cdot k_j}$ operation in the second term involving taking the exponential function first and then averaging. To simplify the computation, the mean-field approximation method can be used in place of this operation.

This method involves first averaging the values and then exponentiating the result, thus resulting in Formula (7).

$$\log \frac{1}{n} \sum_{j=1}^{n} e^{\lambda q_i \cdot k_j} \approx \log \exp \left(\frac{1}{n} \sum_{j=1}^{n} \lambda q_i \cdot k_j \right) = \log \lambda \overline{q_i \cdot k_j} \tag{7}$$

Since the softmax function emphasizes the position of the maximum value, an approximate value for the third term in Formula (6), $\lambda \sum_{j=1}^{n} a_{ij} e^{\lambda q_i \cdot k_j}$, can be obtained, as shown in Formula (8).

$$\lambda \sum_{j=1}^{n} p_{ij} e^{\lambda q_i \cdot k_j} \approx \lambda \max_{1 \le j \le n} \left(e^{\lambda q_i \cdot k_j} \right) \tag{8}$$

It is important to note that the semi-quantitative estimations in Formulas (7) and (8) are used to determine the appropriate scaling factor for compensating for the impact of length on entropy rather than completely disregarding it. Therefore, by substituting Formulas (7) and (8) into Formula (6), we can obtain

$$\begin{aligned} \mathcal{H}_i &\approx \log n + \log \lambda \overline{q_i \cdot k_j} - \lambda \max_{1 \le j \le n} \left(e^{\lambda q_i \cdot k_j} \right) \\ &\approx \log n - \lambda \left(\max_{1 \le j \le n} \left(e^{\lambda q_i \cdot k_j} \right) - \log \lambda \overline{q_i \cdot k_j} \right) \end{aligned} \tag{9}$$

The concept of entropy invariance aims to minimize the impact of sequence length n on entropy. Based on Formula (9), we can set

$$\lambda \propto \log n \tag{10}$$

Therefore, based on the principle of entropy invariance and some reasonable assumptions, we can derive a new scaling factor and a corresponding scaled dot-product attention mechanism, as shown in Formula (11). Specifically, we compute the scaling factor by taking the logarithm of the length of the target sequence and multiplying it with the dot-product score matrix. This scaling factor rescales the values in the score matrix to better adapt to the length of the target sequence. Although this scaling factor is just a scalar, it scales every element in the score matrix, resulting in a scaled attention weight matrix. Finally, the weighted average of the input value vectors with the scaled attention weights generates the final context vector.

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{\log n Q K^T}{\sqrt{d}} \right) V \tag{11}$$

In conclusion, this study obtained a new scaling factor, which makes the model more robust to changes in input length. Based on the entropy invariance of the attention mechanism, this study optimized the attention mechanism of the Transformer module in the disfluency detection model while keeping the other steps the same. This enables the model to generalize to speech data of different lengths and makes it more suitable for real-world disfluency detection scenarios.

4. Experiments

4.1. Datasets

To verify the effectiveness of the proposed method for detecting disfluent speech, we conducted experiments on the open-source English dataset SEP-28k and our self-built Chinese dataset PSC-PS-DF. This section will provide a detailed introduction to both datasets.

4.1.1. SEP-28k

SEP-28k is an open-source English stuttering speech corpus consisting of data from 385 public online podcast collections. These podcast collections come from 8 shows centered around stuttering themes, mainly featuring interviews between people who stutter. The dataset is composed of 28,177 segments, each consisting of 40–250 three-second clips extracted from each podcast episode. The SEP-28k dataset has two different types of annotation: stuttering and non-stuttering. Three annotators label stuttering segments with five types of disfluencies: blocks, prolongations, sound repetitions, word repetitions, and interjections. Additionally, the dataset annotates an additional 4144 stuttering segments from the FluencyBank dataset using the same scheme. All audio segments in SEP-28k are 3 s long and are sampled at 16 kHz. In this study, we mainly focus on stuttering annotations and sample data from the SEP-28k dataset for training our model. We evaluate the model using data from both SEP-28k and FluencyBank and validate the model using data of different sizes. The data distribution used in our experiments is identical to that of Reference [35].

The current study utilizes the data processing method described in Reference [35] for the SEP-28k dataset. Data points that all annotators agreed upon as unambiguous were chosen for sampling, ensuring that both the training and test sets contained high-quality data, thus increasing the reliability of the experimental results. Table 1 provides definitions of the five different types of disfluencies along with specific examples.

Table 1. Definition and examples of different speech disfluencies in the SEP-28k dataset.

Disfluency Labels	Definition	Examples
Sound repetitions (Snd)	Repetitions of syllables	I (wh-wh-) whispered a secret
Word repetitions (WP)	Repetitions of words	I know (know) a secret
Prolongations (Pro)	Extended syllables	I kn(nnnnn)ow
Interjections (Intrj)	Filler words or non-words	I (um) know a (uh) secret
Blocks (Bl)	Long stuttered pauses	I know (pause) a secret

4.1.2. PSC-PS-DF

To address the issue of a lack of resources for Chinese disfluent speech data, this paper used the construction method of the SEP-28k dataset as a reference to construct the Chinese disfluent speech dataset PSC-PS-DF. PSC-PS-DF consists of propositional speaking files for the Mandarin Proficiency Test (Putonghua Shuiping Ceshi, PSC). Previous work has shown that, due to the nature of Chinese propositional speaking, which requires the speaker to freely describe a topic for three minutes without any reference text, the speech files contain a large number of disfluent features, such as “um”, “ah”, and “uh” interjections, blocks, prolongations, and repetition, but such disfluent features are rarely marked and used in research [55]. In this study, disfluent features were annotated in Chinese propositional speaking data to obtain spontaneously spoken disfluent features in Chinese for the detection of disfluent Chinese-language speech. A total of 4414 3 s disfluent segments were extracted from 400 propositional speaking data in the PSC-PS-DF dataset. The disfluent segments were labeled by three annotators with four types of disfluencies, namely WP, Pro, Intrj, and Bl, whose definitions are the same as those in the SEP-28k dataset and can be referred to in Table 1. The Chinese dataset does not include the annotation of Snd, and the repetition of Chinese speech is limited to the annotation of repetitive words and phrases, which are uniformly annotated with WP. The four types of disfluent annotations for the PSC-PS-DF dataset can be visualized by referring to Figure 2. Fluent segments in the PSC-PS-DF dataset are composed of speech unanimously identified by the three annotators as having a fluency level of level 1 in the propositional speaking assessment results. The length of all audio segments in the PSC-PS-DF was set to 3 s, with a sampling frequency of 16 kHz. The data distribution is shown in Table 2.

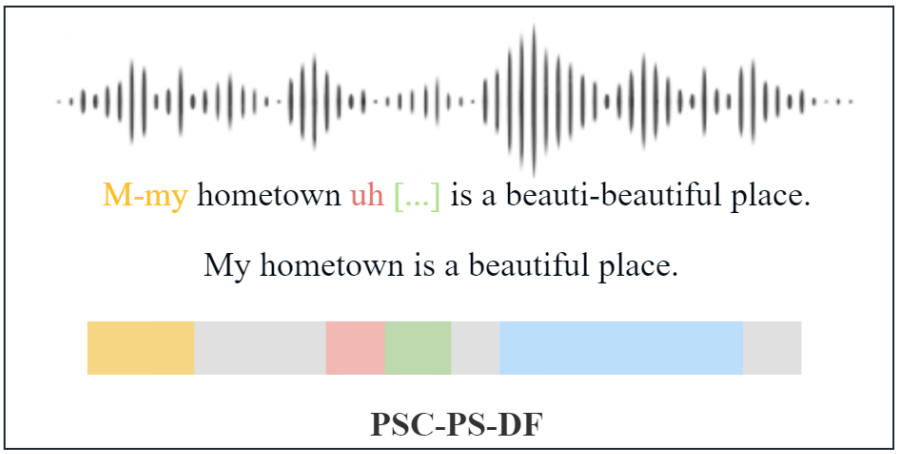

Figure 2. Example figure of the PSC-PS-DF dataset with prolongations (orange), interjections (red), blocks (green), repetitions (blue), and normal parts (gray).

Table 2. Data distribution of the PSC-PS-DF dataset.

Disfluency Labels	Train	Dev	Test	Total	Train Data Size in Minutes
WP	958	96	320	1374	48
Pro	576	58	218	852	29
Intrj	442	45	146	633	22
Bl	1086	109	360	1555	54

When annotating disfluent features in PSC-PS-DF, researchers labeled common words such as "um", "ah", "uh", and "well" as interjections, while word repetition generally occurred when the speaker hesitated, supplemented, or modified their speech. As there is no strict definition for blocks, this study followed Raupach's [56] research viewpoint and uses 0.3 s as the boundary for speech pauses within or between sentences. In addition, this study uses 0.8 s as the boundary for prolongations. When annotating Chinese disfluent features, researchers extract blocks longer than 0.3 s and prolongations longer than 0.8 s as disfluent markers with the labels Bl and Pro, respectively. This study also analyzes Chinese disfluent features, and Figures 3–10 show the distribution of their speech waveforms and the speech spectrograms of the four types of features. The different disfluency features can be more easily identified by analyzing the speech waveforms and speech spectrograms for the five types of disfluencies. Figures 3–10 depict the waveforms (amplitude vs. time) on the left and a time-frequency plot of the wavelet decomposition using these data on the right. From the plots, we can observe that in the PSC-PS-DF dataset, the WP feature is represented by multiple identical speech waveforms, which exhibit shorter articulation time and narrower speech waveforms for words. The Pro feature in the plots exhibits a fuller time-domain waveform with longer articulation durations for single words and syllables. The difference between the Intrj and Pro features is that the Intrj feature has a lower energy and a shorter duration than the Pro feature. The Bl feature contains large pauses and is most easily distinguished by its lower energy.

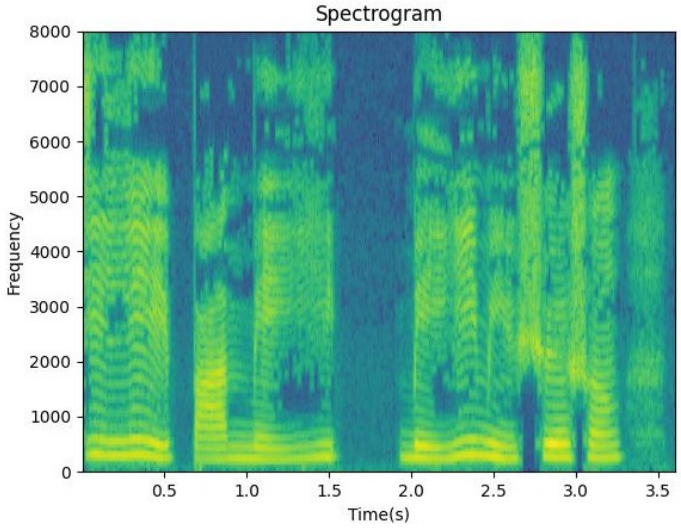

Figure 3. Speech waveform of WP for PSC-PS-DF.

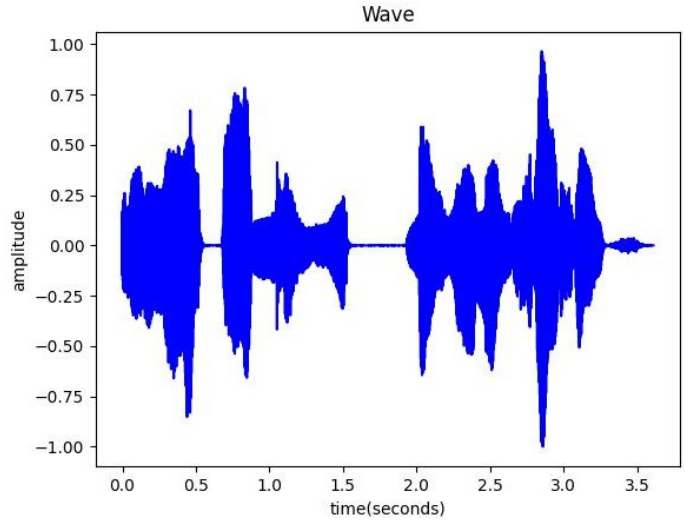

Figure 4. Speech spectrogram of WP for PSC-PS-DF.

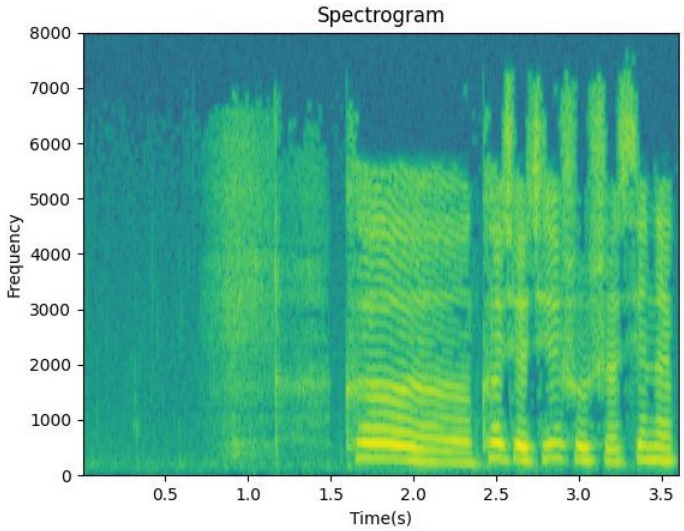

Figure 5. Speech waveform of Pro for PSC-PS-DF.

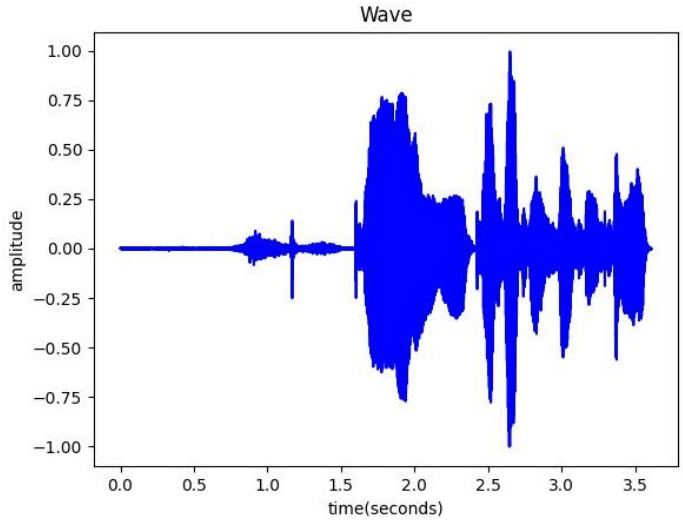

Figure 6. Speech spectrogram of Pro for PSC-PS-DF.

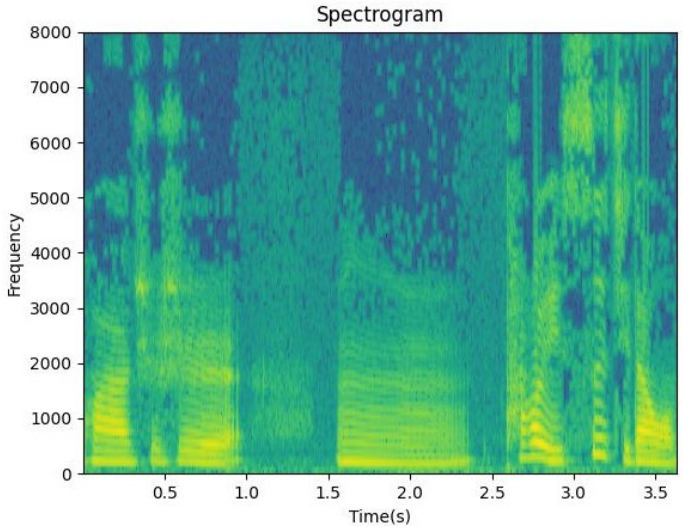

Figure 7. Speech waveform of Intrj for PSC-PS-DF.

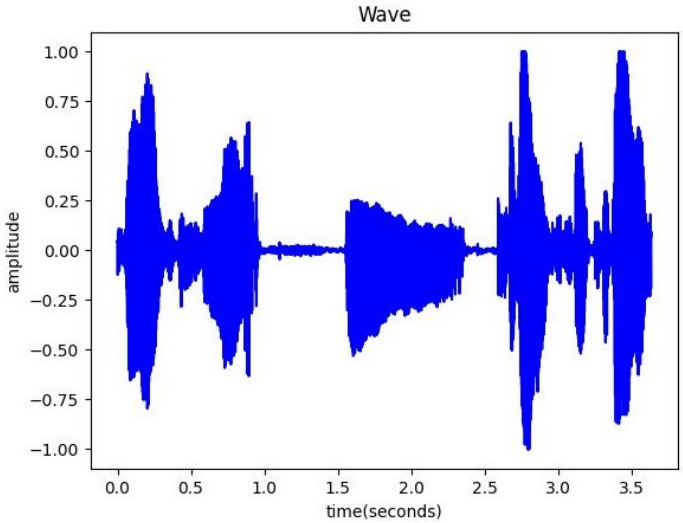

Figure 8. Speech spectrogram of Intrj for PSC-PS-DF.

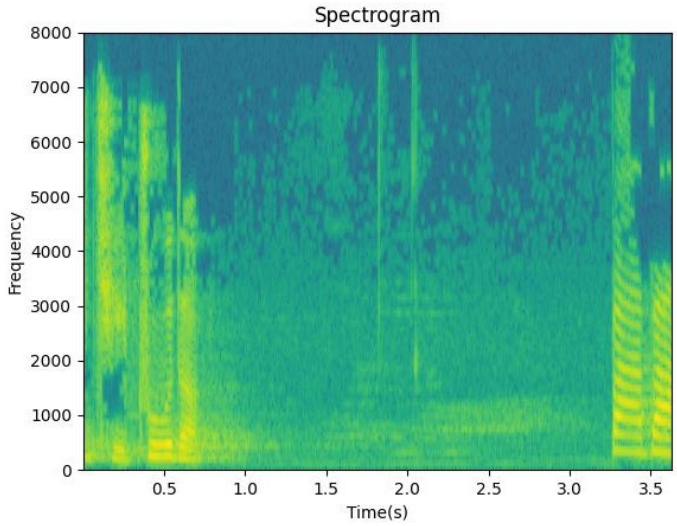

Figure 9. Speech waveform of Bl for PSC-PS-DF.

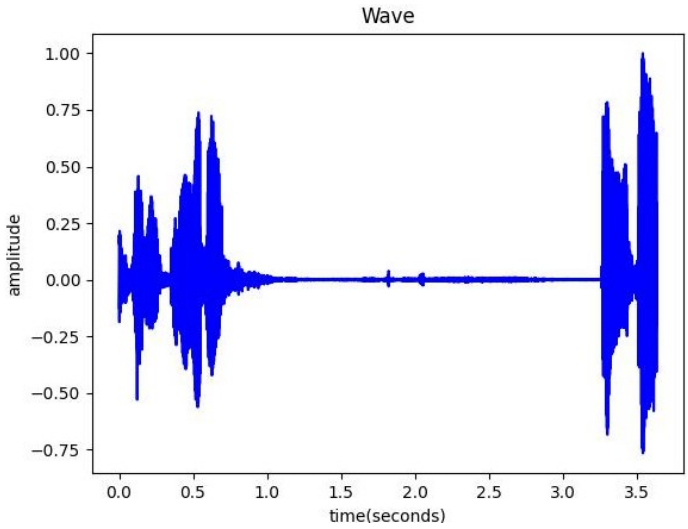

Figure 10. Speech spectrogram of Bl for PSC-PS-DF.

4.2. Basic Settings

The experimental hyperparameters setting of the proposed model are presented in Table 3. The implementation of the wav2vec2.0 model is based on the Huggingface Transformer code repository [57]. As the convergence speed of different fine-grained fluency labels varies during training, the number of epochs in each experiment needs to be adjusted accordingly. Based on the correspondence between the actual and predicted results, samples were classified into four categories: true positives (*TP*), false positives (*FP*), false negatives (*FN*), and true negatives (*TN*). The formulas for calculating the four evaluation metrics are as follows:

$$Precision = \frac{TP}{TP + FP} \tag{12}$$

$$Recall = \frac{TP}{TP + FN} \tag{13}$$

$$F1 = \frac{2}{(1/Precision) + (1/Recall)} \tag{14}$$

$$Accuracy = (TP + TN)/(TP + TN + FN + FP) \tag{15}$$

Table 3. The hyperparameters set in the experiments.

Hyperparameters	Setting
Learning rate	1×10^{-4}
Batch size	512
Optimizer	Adam
Loss function	CrossEntropyLoss
Audio feature dimension	768
Attention dimension/number of heads	50/10
CNN hidden layer dimension	50
wav2vec2.0 (Chinese)	TencentGameMate/chinese-wav2vec2-base
wav2vec2.0 (English)	facebook/wav2vec2-base-960h

5. Results and Analysis

To validate the performance of the proposed speech disfluency detection model, experiments are conducted on both the open-source English dataset SEP-28k and the self-built PSC-PS-DF dataset. The proposed method is experimentally validated in four aspects: evaluation of its performance on the limited data of both datasets, comparison experiments with baseline models, ablation study, and length-scaled attention experiments.

5.1. Evaluation on Limited Data

Given the limited resources of the disfluent speech dataset, this study aims to train a more effective model using the limited disfluent speech data. Consequently, the dataset is divided, and the proposed disfluent speech detection model is compared on the complete SEP-28k dataset, 1/2 of the dataset, and 1/4 of the dataset, and the results are shown in Table 4. The data in Table 4 indicates that the *F*1 of the proposed disfluency detection model decreases as the number of datasets decreases. However, with only 1/4 of the dataset, the *F*1 for all disfluent features, except for the Bl feature, remains above 0.7. This indicates that the proposed model can effectively detect disfluencies even with limited data resources. Figure 11 visualizes the *F*1 of different features under different dataset sizes.

Table 4. Results for all disfluencies on the SEP-28k dataset.

Disfluency	Dataset	Data Size in Minutes	*F*1	*Precision*	*Recall*	*Accuracy* (%)
Snd	1/1	75	0.90	0.82	0.99	89.08
	1/2	37	0.81	0.74	0.89	78.88
	1/4	19	0.74	0.70	0.77	72.33
WP	1/1	148	0.90	0.84	0.96	86.76
	1/2	74	0.79	0.78	0.81	76.36
	1/4	37	0.75	0.74	0.77	70.17
Pro	1/1	75	0.89	0.82	0.98	88.11
	1/2	37	0.78	0.71	0.87	75.97
	1/4	19	0.72	0.68	0.77	70.39
Intrj	1/1	248	0.84	0.87	0.81	83.82
	1/2	124	0.83	0.85	0.81	78.18
	1/4	62	0.80	0.84	0.76	76.90
Bl	1/1	45	0.79	0.77	0.81	78.57
	1/2	22	0.72	0.68	0.78	70.41
	1/4	11	0.69	0.64	0.73	66.33

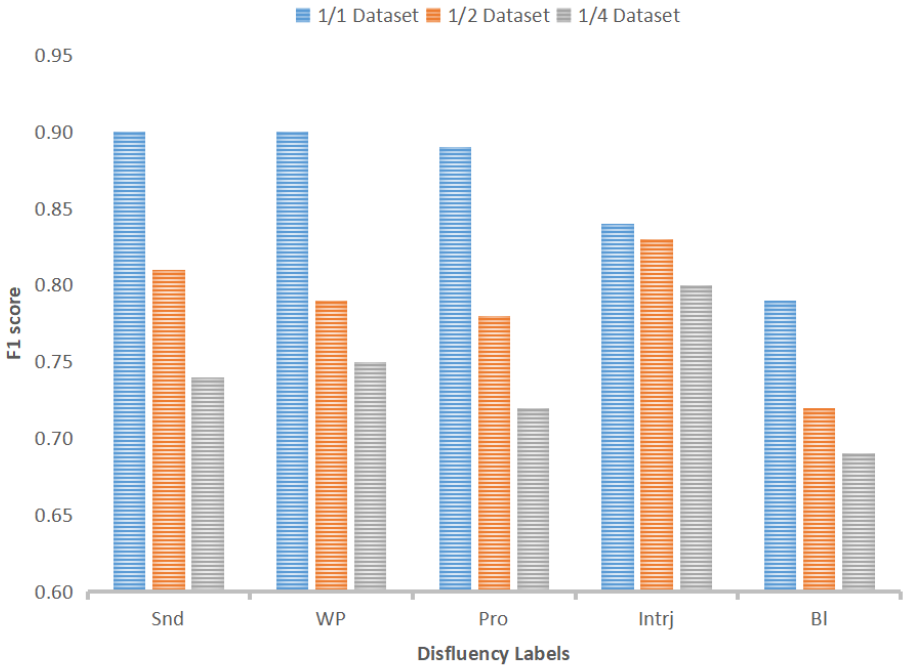

Figure 11. The F1-score of all disfluencies on the SEP-28k dataset at different data sizes.

In addition, this paper also conducts experiments on different disfluency features on the self-built PSC-PS-DF dataset, and the experimental results are shown in Table 5. The experimental results demonstrate that the proposed model also achieves excellent detection results in the Chinese disfluent dataset, with the Bl, WP and Intrj features performing well in Chinese speech disfluency detection, demonstrating that blocks and repetitions as well as interjection features are more easily detectable types of disfluencies in spoken language. In contrast, the performance of the Pro feature is not as good as the other disfluent features, indicating that the prolongations feature of speech does not provide a clear distinction between fluent and disfluent speech.

Table 5. Results for all disfluencies on the PSC-PS-DF dataset.

Disfluency	Data Size in Minutes	*F1*	*Precision*	*Recall*	*Accuracy* (%)
WP	48	0.98	0.99	0.98	98.44
Pro	29	0.84	0.73	0.98	81.73
Intrj	22	0.94	0.97	0.92	94.52
Bl	54	0.99	0.99	0.99	98.89

5.2. Comparison with Baseline Models

In this section, the proposed model is compared with the long short-term memory (LSTM) model, MLP model, and the baseline model DisfluencyNet [35] on the 1/4 SEP-28k dataset, and the results are shown in Table 6. The results indicate that the proposed model in this paper yields superior detection results for the Snd, WP, Intrj, and Bl disfluency features when compared to other baseline models. Regarding the Pro disfluency feature, the proposed model in this paper achieves an F1-score that is 0.01 lower than that of the baseline model DisfluencyNet. This suggests that there is still scope for improving the detection of disfluencies such as prolongations. For a clearer comparison of how different baseline models perform on 1/4 of the SEP-28k dataset, please refer to Figure 12.

In addition, this study also compares the convergence speed of the proposed model with the baseline model DisfluencyNet during the training process. Figures 13–16 show the training loss curves for the Snd, WP, Pro, and Bl labels at 400 epochs in the full

SEP-28k dataset, respectively, from which it can be seen that the proposed model converges significantly faster than the baseline model DisfluencyNet.

Table 6. Performance of baseline models on 1/4 of the SEP-28k dataset.

Disfluency	Model	*F1*	*Precision*	*Recall*	*Accuracy* (%)
Snd	LSTM	0.66	0.65	0.67	65.05
	MLP	0.70	0.68	0.72	69.17
	DisfluencyNet	0.72	0.67	0.79	70.00
	Ours	0.74	0.70	0.77	72.33
WP	LSTM	0.72	0.71	0.74	67.70
	MLP	0.71	0.72	0.69	69.68
	DisfluencyNet	0.71	0.75	0.66	71.00
	Ours	0.75	0.74	0.77	70.17
Pro	LSTM	0.60	0.64	0.60	60.44
	MLP	0.63	0.62	0.65	62.14
	DisfluencyNet	0.73	0.80	0.76	75.70
	Ours	0.72	0.68	0.77	70.39
Intrj	LSTM	0.69	0.67	0.71	70.00
	MLP	0.68	0.81	0.59	72.73
	DisfluencyNet	0.79	0.79	0.79	74.50
	Ours	0.80	0.84	0.76	76.90
Bl	LSTM	0.49	0.50	0.49	50.00
	MLP	0.56	0.53	0.59	53.06
	DisfluencyNet	0.58	0.54	0.61	55.00
	Ours	0.69	0.64	0.73	66.33

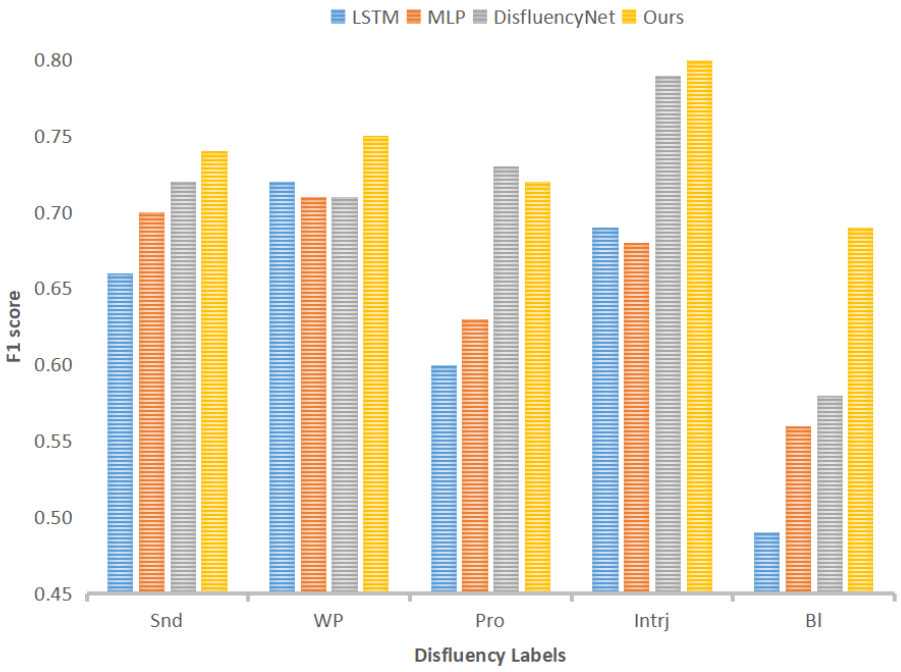

Figure 12. Performance of baseline models on 1/4 of the SEP-28k dataset.

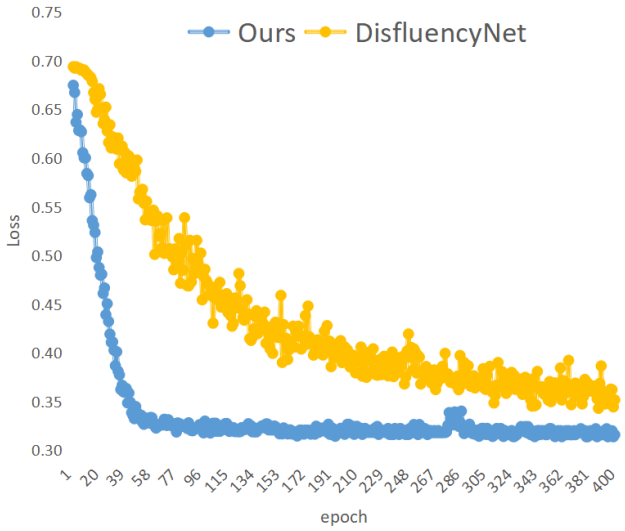

Figure 13. Training Loss Curves for Snd.

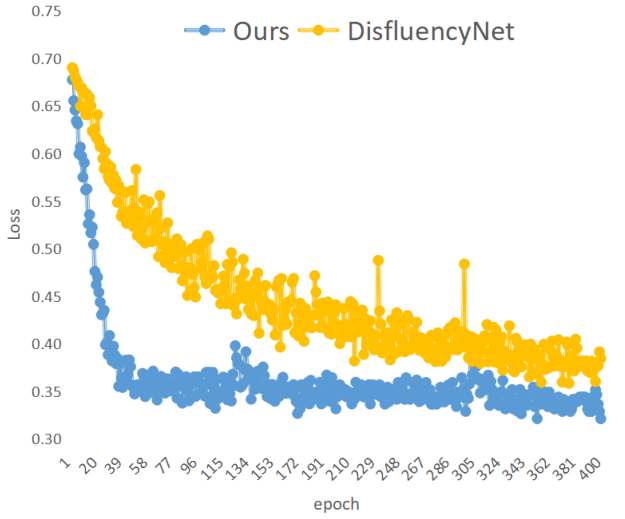

Figure 14. Training Loss Curves for WP.

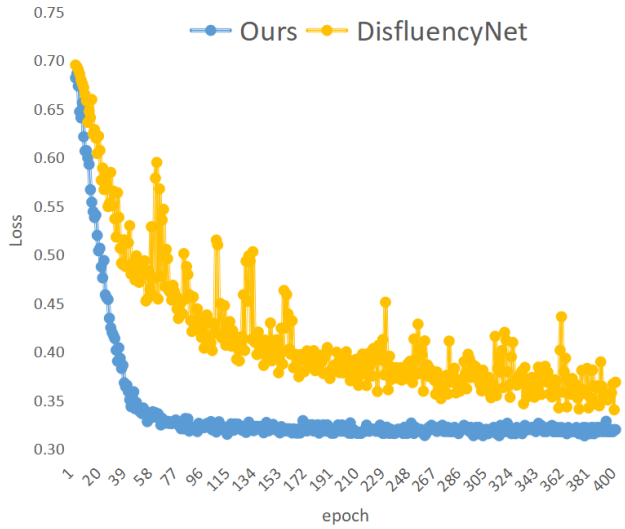

Figure 15. Training Loss Curves for Pro.

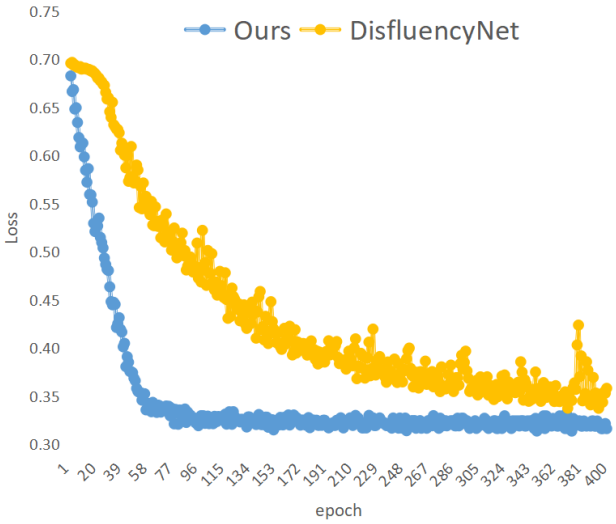

Figure 16. Training Loss Curves for Bl.

In this study, comparison experiments with different baseline models were also conducted on the self-built PSC-PS-DF dataset, and the results are shown in Table 7. The findings indicate that the model proposed in this paper outperforms other baseline models in detecting Pro, Intrj, and Bl labels in Chinese. However, there is no significant difference between the proposed model and other models for WP labels. The performance of the different models on the PSC-PS-DF dataset can be seen in Figure 17.

Table 7. Performance of baseline models on the PSC-PS-DF dataset.

Disfluency	Model	*F1*	*Precision*	*Recall*	*Accuracy* (%)
WP	LSTM	0.97	0.97	0.97	97.19
	MLP	0.98	0.98	0.98	97.81
	DisfluencyNet	0.98	0.99	0.98	98.44
	Ours	0.98	0.99	0.98	98.44
Pro	LSTM	0.82	0.71	0.95	78.44
	MLP	0.82	0.71	0.97	78.44
	DisfluencyNet	0.83	0.83	0.83	82.57
	Ours	0.84	0.73	0.98	81.73
Intrj	LSTM	0.92	0.94	0.90	92.47
	MLP	0.93	0.96	0.90	93.15
	DisfluencyNet	0.93	0.94	0.92	93.15
	Ours	0.94	0.97	0.92	94.52
Bl	LSTM	0.95	0.94	0.96	95.00
	MLP	0.96	0.95	0.98	96.11
	DisfluencyNet	0.97	0.98	0.97	97.50
	Ours	0.99	0.99	0.99	98.89

5.3. Ablation Study

To verify the impact of the CNN and Transformer modules on the performance of the model proposed in this paper, we conducted ablation experiments on the SEP-28k and PSC-PS-DF datasets. Figure 18 demonstrates the impact of removing the CNN and Transformer modules from the model on the SEP-28k dataset, and Tables 8 and 9 show the results of the ablation experiments on different datasets. First, we evaluated the impact of removing the CNN module on the model. For the Snd, WP, Pro, Intrj, and Bl labels in the SEP-28k dataset, removing the CNN module resulted in a decrease in F1-score of 0.05, 0.04, 0.04, 0.03, and 0.08, respectively. The effect of removing the CNN module on the PSC-PS-DF dataset was not significant, with only a 0.04 decrease in the Bl label. Second, we

removed the Transformer module, and the performance of the model on both the SEP-28k and PSC-PS-DF datasets significantly decreased. For the Snd, WP, Pro, Intrj, and Bl labels in the SEP-28k dataset, the F1-score decreased by 0.08, 0.05, 0.10, 0.13, and 0.13, respectively. For the WP, Pro, and Bl labels in the PSC-PS-DF dataset, the F1-score decreased by 0.01, 0.02, and 0.02, respectively. This indicates that the Transformer module has a significant impact on the overall performance of the model. The ablation study verifies the functions of each module in our model and validates the contributions of each module to the performance of the model.

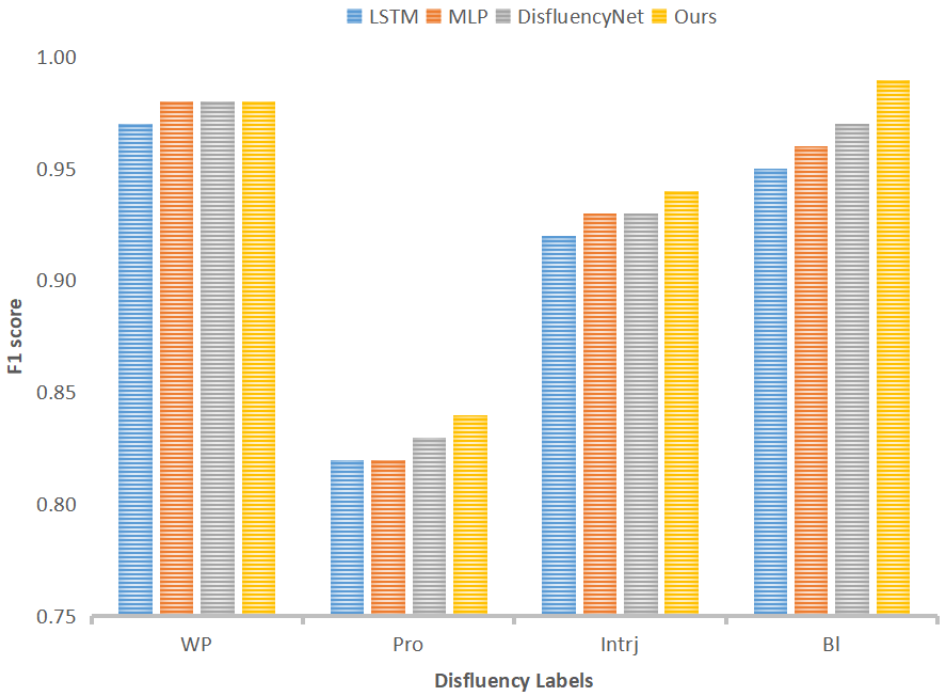

Figure 17. Performance of baseline models on the PSC-PS-DF dataset.

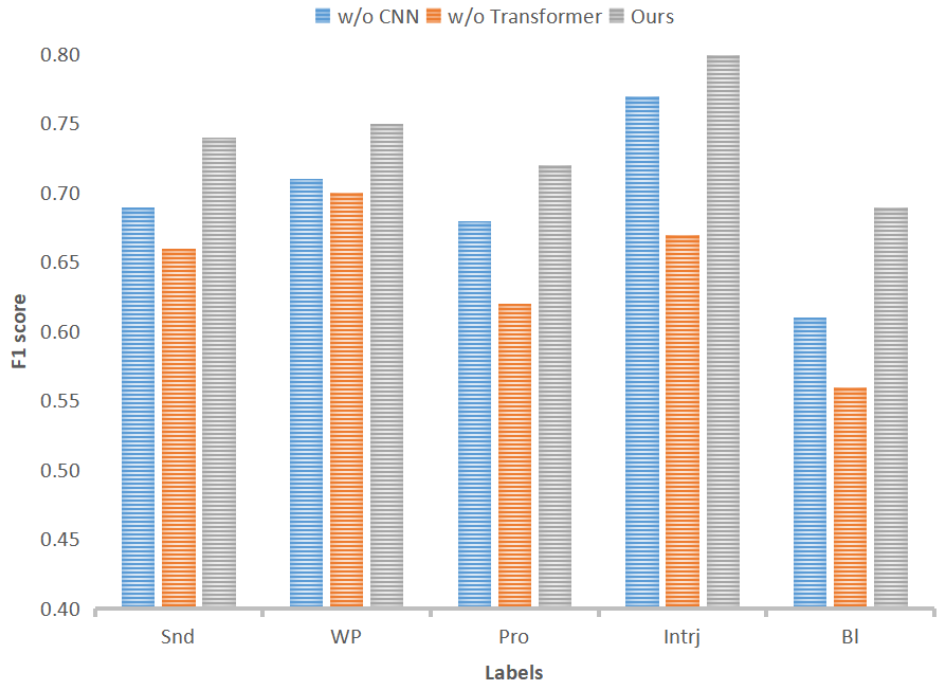

Figure 18. Results of the ablation experiment on the SEP-28k dataset.

Table 8. Results of the ablation experiment on the SEP-28k dataset.

Disfluency	Model	*F1*	*Precision*	*Recall*	*Accuracy* (%)
Snd	w/o CNN	0.69	0.69	0.69	68.69
	w/o Transformer	0.66	0.68	0.65	67.23
	Ours	0.74	0.70	0.77	72.33
WP	w/o CNN	0.71	0.66	0.76	69.18
	w/o Transformer	0.70	0.68	0.71	68.31
	Ours	0.75	0.74	0.77	70.17
Pro	w/o CNN	0.68	0.63	0.73	65.53
	w/o Transformer	0.62	0.61	0.63	61.65
	Ours	0.72	0.68	0.77	70.39
Intrj	w/o CNN	0.77	0.83	0.71	74.91
	w/o Transformer	0.67	0.71	0.63	71.45
	Ours	0.80	0.84	0.76	76.90
Bl	w/o CNN	0.61	0.56	0.67	57.14
	w/o Transformer	0.56	0.56	0.55	56.12
	Ours	0.69	0.64	0.73	66.33

Table 9. Results of the ablation experiment on the PSC-PS-DF dataset.

Disfluency	Model	*F1*	*Precision*	*Recall*	*Accuracy* (%)
WP	w/o CNN	0.98	0.98	0.98	97.81
	w/o Transformer	0.97	0.98	0.97	97.50
	Ours	0.98	0.99	0.98	98.44
Pro	w/o CNN	0.84	0.73	0.98	80.73
	w/o Transformer	0.82	0.70	0.97	77.98
	Ours	0.84	0.73	0.98	81.73
Intrj	w/o CNN	0.94	0.96	0.92	93.84
	w/o Transformer	0.94	0.97	0.90	93.84
	Ours	0.94	0.97	0.92	94.52
Bl	w/o CNN	0.95	0.96	0.94	95.00
	w/o Transformer	0.97	0.96	0.97	96.67
	Ours	0.99	0.99	0.99	98.89

5.4. Length-Scaled Attention

This paper presents length-scaled attention as a solution to the problem of fixed training sequence length and varying testing sequence length. In this method, the training and testing sequences are standardized to a length of 3 s for standard input. To evaluate the effectiveness of length-scaled attention, we tested the method by clipping the original sequence to half its original length. Then, we conducted experiments to verify the impact of length-scaled attention on the experiments while keeping all other configurations unchanged. Tables 10 and 11, respectively, show the changes in model performance when the length of training and testing sequences varies in different datasets. When trained and tested on data of the same length, adding length-scaled attention leads to an improve-

ment in experimental results, demonstrating the importance of length-scaled attention in disfluency detection models.

In the SEP-28k dataset, WP and Intrj labels are enhanced by 0.14 and 0.03, respectively, using length-scaled attention compared to removing length-scaled attention for the same test length, while the training data length is reduced to half of the original length. In the PSC-PS-DF dataset, when the training data are half of their original length, compared to removing length-scaled attention, WP, Pro, and Intrj labels are enhanced by 0.03, 0.01, and 0.01, respectively, using length-scaled attention. Experimental results for both datasets show that the model without the addition of length-scaled attention performs worse in most disfluent label detection experiments when the training sequence length is reduced to half of the original length.

Furthermore, as shown in Tables 10 and 11, shorter testing sequences lead to a decrease in the performance of length-scaled attention. Models that use length-scaled attention do not show their advantages, possibly because the duration of disfluency speech varies, and to standardize the length, zeros are sometimes padded at the end of the sequence. These clipped data may consist of useless zero frames.

In conclusion, this experiment demonstrates that a model with length-scaled attention can improve the results of disfluency detection experiments and can be applied to stuttering detection in speech with inconsistent training and testing sequence lengths.

Table 10. F1-score comparisons on SEP-28k with different length distributions.

Disfluency	Type	Train All Test All	Train 1/2 Test All	Train All Test 1/2
Snd	Length-scaled	0.74	0.66	0.72
	w/o length-scaled	0.74	0.67	0.72
WP	Length-scaled	0.75	0.53	0.43
	w/o length-scaled	0.71	0.39	0.52
Pro	Length-scaled	0.72	0.67	0.70
	w/o length-scaled	0.70	0.67	0.71
Intrj	Length-scaled	0.80	0.79	0.50
	w/o length-scaled	0.79	0.76	0.79
Bl	Length-scaled	0.69	0.58	0.69
	w/o length-scaled	0.67	0.58	0.68

Table 11. F1-score comparisons on PSC-PS-DF with different length distributions.

Disfluency	Type	Train All Test All	Train 1/2 Test All	Train All Test 1/2
WP	Length-scaled	0.98	0.97	0.98
	w/o length-scaled	0.97	0.94	0.97
Pro	Length-scaled	0.84	0.84	0.77
	w/o length-scaled	0.79	0.83	0.70
Intrj	Length-scaled	0.94	0.94	0.88
	w/o length scaled	0.93	0.93	0.86
Bl	Length-scaled	0.99	0.96	0.99
	w/o length-scaled	0.98	0.97	0.98

6. Conclusions

Speech is the most efficient and widely used means of communication among humans. However, not everyone has normal communication abilities, and speech disfluency often occurs during communication. Speech disfluency can manifest as normal speech disfluency in daily life for the general population, which is related to people's emotions, urgency, or nervousness. It can also manifest as stuttering in individuals who involuntarily repeat, prolong, or interrupt sounds, making it difficult for them to express themselves clearly. Currently, detecting speech disfluency in daily life and stuttering detection both require the assistance of experts, and there is a lack of validated automated evaluation methods. The current methods for detecting speech disfluency heavily depend on annotated data, which can be expensive. Furthermore, these methods do not address the issue of variable-length disfluent speech, which restricts the scalability of detection methods. To address these limitations, this paper proposes a method for detecting speech disfluency using wav2vec2.0 , CNN, and Transformer. The main feature of this method is its applicability to different languages and its ability to handle variable-length disfluent speech signals. Firstly, this paper constructs the PSC-PS-DF dataset for Chinese disfluent speech, which consists of four disfluent features: interjections, blocks, prolongations, and repetitions. Then, the paper uses context embeddings from the pre-training model wav2vec2.0 and combines CNN and Transformer to build a classification network for automated speech disfluency detection. Considering that the length of disfluent speech data varies in practical detection scenarios, the paper improves the model based on the entropy invariance of attention mechanisms, allowing the model's results to generalize to speech data of different lengths. In the experiments, we tested the proposed method using speech signals of different languages and lengths, and the results showed that the proposed method performed well in these scenarios. The paper concludes that the proposed model achieved good disfluency detection results in both self-built datasets and the open-source dataset for English disfluent speech, indicating its potential for detecting speech disfluency in different languages and different lengths.

Our method can be applied to different fields such as speech recognition, speech synthesis, and human-computer interactions. In future research, we hope to improve the proposed speech disfluency detection model in various ways, as follows:

- Since the accuracy of speech disfluency detection methods is directly related to the size of the data, but the cost of collecting and labeling disfluent data is high, we can use more efficient data processing and extraction methods to obtain more reliable disfluency detection results with limited data.
- The wav2vec2.0 model has a good effect on extracting disfluent speech features. In this paper, we only used the context representation of the last hidden layer of the wav2vec2.0 model as the input to the model. In future research, we can consider fine-tuning the wav2vec2.0 model to obtain effective detection results. In addition, we can also try using other speech pre-training models such as HuBERT and WavLM instead of the wav2vec2.0 model used in this paper to obtain better results.
- This paper conduct experiments on self-built and open-source English datasets, including SEP-28k, to verify that the model can be applied in different language environments. In the future, we can conduct experiments on this model in more languages to verify its reliability.
- In the future, we can expand our work from single-task scenarios to multi-classification scenarios, not only detecting disfluency but also distinguishing between different types of speech disfluency events.

Author Contributions: Conceptualization, J.L. and A.W.; methodology, J.L. and A.W.; software, J.L.; validation, J.L., D.W. and S.G.; formal analysis, J.L.; investigation, J.L.; resources, J.L. and S.G.; data curation, J.L. and D.W.; writing—original draft preparation, J.L.; writing—review and editing, J.L. and D.W.; visualization, D.W.; supervision, D.W.; project administration, A.W.; funding acquisition, A.W. All authors have read and agreed to the published version of the manuscript.

Funding: The funding for this research was provided by two sources: the Central Guiding Local Science and Technology Development Special Fund Project, through grant 202204120018, and the Basic Research Program of Tianshan Talent Plan of Xinjiang, China, through grant 2022TSYCJU0005.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The SEP-28k dataset used in this study is currently accessible [14]. The process of expanding the self-built PSC-PS-DF dataset is ongoing, and it has not been made publicly available yet.

Acknowledgments: The authors gratefully acknowledge all anonymous reviewers and editors for their constructive suggestions for the improvement of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neural Network
PSC	Putonghua Shuiping Ceshi
MFCC	Mel Frequency Cepstral Coefficient
LPC	Linear Predictive Coding
LPCC	Linear Prediction Cepstral Coefficient
PLP	Perceptual Linear Prediction
ANN	Artificial Neural Network
HMM	Hidden Markov Model
SVM	Support Vector Machine
KNN	K-Nearest Neighbor
LDA	Linear Discriminant Analysis
DTW	Dynamic Time Warping
MLP	Multilayer Perceptron
BLSTM	Bidirection Long Short-Term Memory
CT-Transformer	Controllable Time-delay Transformer
TDNN	Time-delay Neural Network
LSTM	Long Short-Term Memory

References

1. Gupta, S.; Shukla, R.S.; Shukla, R.K. Literature survey and review of techniques used for automatic assessment of Stuttered Speech. *Int. J. Manag. Technol. Eng.* **2019**, *9*, 229–240.
2. Starkweather, C.W. *Fluency and Stuttering*; Prentice-Hall, Inc.: Upper Saddle River, NJ, USA, 1987.
3. Maguire, G.A.; Yeh, C.Y.; Ito, B.S. Overview of the diagnosis and treatment of stuttering. *J. Exp. Clin. Med.* **2012**, *4*, 92–97. [CrossRef]
4. Lawrence, M.; Barclay, D.M., III. Stuttering: A brief review. *Am. Fam. Physician* **1998**, *57*, 2175. [PubMed]
5. Yairi, E.; Ambrose, N. Epidemiology of stuttering: 21st century advances. *J. Fluen. Disord.* **2013**, *38*, 66–87. [CrossRef]
6. Seitz, S.R.; Choo, A.L. Stuttering: Stigma and perspectives of (dis) ability in organizational communication. *Hum. Resour. Manag. Rev.* **2022**, *32*, 100875. [CrossRef]
7. Manjula, G.; Kumar, M.S. Overview of analysis and classification of stuttered speech. *Int. J. Ind. Electron. Electr. Eng.* **2016**, *4*, 80–86.
8. Sheikh, S.A.; Sahidullah, M.; Hirsch, F.; Ouni, S. Machine learning for stuttering identification: Review, challenges and future directions. *Neurocomputing* **2022**, *514*, 385–402. [CrossRef]
9. Barrett, L.; Hu, J.; Howell, P. Systematic review of machine learning approaches for detecting developmental stuttering. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2022**, *30*, 1160–1172. [CrossRef]
10. Khara, S.; Singh, S.; Vir, D. A comparative study of the techniques for feature extraction and classification in stuttering. In Proceedings of the 2018 IEEE Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, 20–21 April 2018; pp. 887–893.
11. Sharma, N.M.; Kumar, V.; Mahapatra, P.K.; Gandhi, V. Comparative Analysis of Various Feature Extraction Techniques for Classification of Speech Disfluencies. *Speech Commun.* **2023**, *150*, 23–31. [CrossRef]
12. Howell, P.; Davis, S.; Bartrip, J. The UCLASS archive of stuttered speech. *J. Speech Lang. Hear. Res.* **2009**, *52*, 556–569. [CrossRef]

13. Ratner, N.B.; MacWhinney, B. Fluency Bank: A new resource for fluency research and practice. *J. Fluen. Disord.* **2018**, *56*, 69–80. [CrossRef]
14. Lea, C.; Mitra, V.; Joshi, A.; Kajarekar, S.; Bigham, J.P. Sep-28k: A dataset for stuttering event detection from podcasts with people who stutter. In Proceedings of the ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, ON, Canada, 6–11 June 2021; pp. 6798–6802.
15. Bayerl, S.P.; von Gudenberg, A.W.; Hönig, F.; Nöth, E.; Riedhammer, K. KSoF: The Kassel State of Fluency Dataset–A Therapy Centered Dataset of Stuttering. *arXiv* **2022**, arXiv:2203.05383.
16. Tan, T.S.; Ariff, A.; Ting, C.M.; Salleh, S.H. Application of Malay speech technology in Malay speech therapy assistance tools. In Proceedings of the 2007 International Conference on Intelligent and Advanced Systems, Kuala Lumpur, Malaysia, 25–28 November 2007; pp. 330–334.
17. Ravikumar, K.; Rajagopal, R.; Nagaraj, H. An approach for objective assessment of stuttered speech using MFCC. In Proceedings of the The International Congress for Global Science and Technology, Ottawa, ON, Canada, 2–17 July 2009; Volume 19.
18. Chee, L.S.; Ai, O.C.; Hariharan, M.; Yaacob, S. MFCC based recognition of repetitions and prolongations in stuttered speech using k-NN and LDA. In Proceedings of the 2009 IEEE Student Conference on Research and Development (SCOReD), Kuala Lumpur, Malaysia, 16–18 November 2009; pp. 146–149.
19. Km, R.K.; Ganesan, S. Comparison of multidimensional MFCC feature vectors for objective assessment of stuttered disfluencies. *Int. J. Adv. Netw. Appl.* **2011**, *2*, 854–860.
20. Ai, O.C.; Hariharan, M.; Yaacob, S.; Chee, L.S. Classification of speech dysfluencies with MFCC and LPCC features. *Expert Syst. Appl.* **2012**, *39*, 2157–2165.
21. Pálfy, J. Analysis of dysfluencies by computational intelligence. *Inf. Sci. Technol.* **2014**, *6*, 45.
22. Jabeen, S.; Ravikumar, K. Analysis of 0dB and 10dB babble noise on stuttered speech. In Proceedings of the 2015 International Conference on Soft-Computing and Networks Security (ICSNS), Coimbatore, India, 25–27 February 2015; pp. 1–5.
23. Esmaili, I.; Dabanloo, N.J.; Vali, M. Automatic classification of speech dysfluencies in continuous speech based on similarity measures and morphological image processing tools. *Biomed. Signal Process. Control* **2016**, *23*, 104–114. [CrossRef]
24. Sheikh, S.A.; Sahidullah, M.; Hirsch, F.; Ouni, S. Stutternet: Stuttering detection using time delay neural network. In Proceedings of the 2021 29th European Signal Processing Conference (EUSIPCO), Dublin, Ireland, 23–27 August 2021; pp. 426–430.
25. Hariharan, M.; Chee, L.S.; Ai, O.C.; Yaacob, S. Classification of speech dysfluencies using LPC based parameterization techniques. *J. Med Syst.* **2012**, *36*, 1821–1830. [CrossRef]
26. Thiang, W. Speech Recognition Using LPC and HMM Applied for Controlling Movement of Mobile Robot. *Semin. Nas. Teknol. Inf.* **2010**, 97-031.
27. Fook, C.Y.; Muthusamy, H.; Chee, L.S.; Yaacob, S.B.; Adom, A.H.B. Comparison of speech parameterization techniques for the classification of speech disfluencies. *Turk. J. Electr. Eng. Comput. Sci.* **2013**, *21*, 1983–1994. [CrossRef]
28. Chee, L.S.; Ai, O.C.; Hariharan, M.; Yaacob, S. Automatic detection of prolongations and repetitions using LPCC. In Proceedings of the 2009 International Conference for Technical Postgraduates (TECHPOS), Kuala Lumpur, Malaysia, 14–15 December 2009; pp. 1–4.
29. Kourkounakis, T.; Hajavi, A.; Etemad, A. Detecting multiple speech disfluencies using a deep residual network with bidirectional long short-term memory. In Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6089–6093.
30. Kourkounakis, T.; Hajavi, A.; Etemad, A. Fluentnet: End-to-end detection of stuttered speech disfluencies with deep learning. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 2986–2999. [CrossRef]
31. Al-Banna, A.K.; Edirisinghe, E.; Fang, H. Stuttering Detection Using Atrous Convolutional Neural Networks. In Proceedings of the 2022 13th International Conference on Information and Communication Systems (ICICS), Dalian, China, 17–19. October 2022; pp. 252–256.
32. Baevski, A.; Zhou, Y.; Mohamed, A.; Auli, M. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 12449–12460.
33. Pepino, L.; Riera, P.; Ferrer, L. Emotion recognition from speech using wav2vec 2.0 embeddings. *arXiv* **2021**, arXiv:2104.03502.
34. Xu, X.; Kang, Y.; Cao, S.; Lin, B.; Ma, L. Explore wav2vec 2.0 for Mispronunciation Detection. In Proceedings of the Interspeech, Brno, Czech Republic, 30 August–3 September 2021; pp. 4428–4432.
35. Mohapatra, P.; Pandey, A.; Islam, B.; Zhu, Q. Speech disfluency detection with contextual representation and data distillation. In Proceedings of the 1st ACM International Workshop on Intelligent Acoustic Systems and Applications, Portland, OR, USA, 1 July 2022; pp. 19–24.
36. Bayerl, S.P.; Wagner, D.; Nöth, E.; Riedhammer, K. Detecting dysfluencies in stuttering therapy using wav2vec 2.0. *arXiv* **2022**, arXiv:2204.03417.
37. Bayerl, S.P.; Wagner, D.; Nöth, E.; Bocklet, T.; Riedhammer, K. The Influence of Dataset Partitioning on Dysfluency Detection Systems. In Proceedings of the Text, Speech, and Dialogue: 25th International Conference, TSD 2022, Brno, Czech Republic, 6–9 September 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 423–436.
38. Bayerl, S.P.; Wagner, D.; Hönig, F.; Bocklet, T.; Nöth, E.; Riedhammer, K. Dysfluencies Seldom Come Alone–Detection as a Multi-Label Problem. *arXiv* **2022**, arXiv:2210.15982.

39. Bayerl, S.P.; Gerczuk, M.; Batliner, A.; Bergler, C.; Amiriparian, S.; Schuller, B.; Nöth, E.; Riedhammer, K. Classification of stuttering–The ComParE challenge and beyond. *Comput. Speech Lang.* **2023**, *81*, 101519. [CrossRef]
40. Howell, P.; Sackin, S. Automatic recognition of repetitions and prolongations in stuttered speech. In Proceedings of the First World Congress on Fluency Disorders, Munich, Germany, 8–11 August 1995; University Press Nijmegen: Nijmegen, The Netherlands, 1995; Volume 2, pp. 372–374.
41. Geetha, Y.; Pratibha, K.; Ashok, R.; Ravindra, S.K. Classification of childhood disfluencies using neural networks. *J. Fluen. Disord.* **2000**, *25*, 99–117. [CrossRef]
42. Savin, P.; Ramteke, P.B.; Koolagudi, S.G. Recognition of repetition and prolongation in stuttered speech using ANN. In Proceedings of the 3rd International Conference on Advanced Computing, Networking and Informatics: ICACNI 2015, Bhubaneswar, India, 23–25 June 2015; Springer: Berlin/Heidelberg, Germany, 2016; Volume 1, pp. 65–71.
43. Hariharan, M.; Vijean, V.; Fook, C.; Yaacob, S. Speech stuttering assessment using sample entropy and Least Square Support Vector Machine. In Proceedings of the 2012 IEEE 8th International Colloquium on Signal Processing and Its Applications, Malacca, Malaysia, 23–25 March 2012; pp. 240–245.
44. Ramteke, P.B.; Koolagudi, S.G.; Afroz, F. Repetition detection in stuttered speech. In Proceedings of the 3rd International Conference on Advanced Computing, Networking and Informatics: ICACNI 2015, Bhubaneswar, India, 23–25 June 2015; Springer: Berlin/Heidelberg, Germany, 2016; Volume 1, pp. 611–617.
45. Świetlicka, I.; Kuniszyk-Jóźkowiak, W.; Smołka, E. Hierarchical ANN system for stuttering identification. *Comput. Speech Lang.* **2013**, *27*, 228–242. [CrossRef]
46. Szczurowska, I.; Kuniszyk-Jóźkowiak, W.; Smołka, E. The application of Kohonen and multilayer perceptron networks in the speech nonfluency analysis. *Arch. Acoust.* **2014**, *31*, 205–210.
47. Zayats, V.; Ostendorf, M.; Hajishirzi, H. Disfluency detection using a bidirectional LSTM. In Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH), San Francisco, CA, USA, 8–12 September 2016; pp. 2523–2527.
48. Zayats, V.; Ostendorf, M. Giving Attention to the Unexpected: Using Prosody Innovations in Disfluency Detection. In Proceedings of the NAACL-HLT, Online, 6–11 June 2019; pp. 86–95.
49. Santoso, J.; Yamada, T.; Makino, S. Classification of causes of speech recognition errors using attention-based bidirectional long short-term memory and modulation spectrum. In Proceedings of the 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), Lanzhou, China, 18–21 November 2019; pp. 302–306.
50. Wang, S.; Che, W.; Liu, Q.; Qin, P.; Liu, T.; Wang, W.Y. Multi-task self-supervised learning for disfluency detection. In Proceedings of the AAAI Conference on Artificial Intelligence, New York, NY, USA, 7–12 February 2020; Volume 34; pp. 9193–9200.
51. Chen, Q.; Chen, M.; Li, B.; Wang, W. Controllable time-delay transformer for real-time punctuation prediction and disfluency detection. In Proceedings of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 8069–8073.
52. Su, J. Entropy Invariance in Softmax Operation. Available online: https://kexue.fm/archives/9034 (accessed on 11 April 2022).
53. Chiang, D.; Cholak, P. Overcoming a Theoretical Limitation of Self-Attention. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland, 22–27 May 2022; pp. 7654–7664.
54. Liu, F.; Shen, S.Y.; Fu, Z.W.; Wang, H.Y.; Zhou, A.M.; Qi, J.Y. LGCCT: A light gated and crossed complementation transformer for multimodal speech emotion recognition. *Entropy* **2022**, *24*, 1010. [CrossRef]
55. Liu, J.; Wumaier, A.; Fan, C.; Guo, S. Automatic Fluency Assessment Method for Spontaneous Speech without Reference Text. *Electronics* **2023**, *12*, 1775. [CrossRef]
56. Raupach, M. Temporal variables in first and second language speech production. In *Temporal Variables in Speech*; De Gruyter Mouton: Berlin, Germany 2011; pp. 263–270.
57. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Online, 16–20 November 2020; pp. 38–45.