Circuits, Systems, and Signal Processing
https://doi.org/10.1007/s00034-021-01657-1

A Hybrid Deep Ensemble for Speech Disfluency Classification

Sheena Christabel Pravin[1] · **M. Palanivelan**[1]

Received: 30 May 2020 / Revised: 17 January 2021 / Accepted: 22 January 2021

Abstract
In this paper, a novel Hybrid Deep Ensemble (HDE) is proposed for automatic speech disfluency classification on a sparse speech dataset. Categorizations of speech disfluencies for diagnosis of speech disorders have so long relied on sophisticated deep learning models. Such a task can be accomplished by a straightforward approach with high accuracy by the proposed model which is an optimal combination of diverse machine learning and deep learning algorithms in a hierarchical arrangement which includes a deep autoencoder that yields the compressed latent features. The proposed model has shown considerable improvement in downgrading processing time overcoming the issues of cumbersome hyper-parameter tuning and huge data demand of the deep learning algorithms with high classification accuracy. Experimental results show that the proposed Hybrid Deep Ensemble has superior performance compared to the individual base learners, and the deep neural network as well. The proposed model and the baseline models were evaluated in terms of Cohen's kappa coefficient, Hamming loss, Jaccard score, F-score and classification accuracy.

Keywords Hybrid Deep Ensemble · Speech disfluency classification · Sparse speech dataset · Deep autoencoder · Latent features

1 Introduction

Disfluencies are a major characteristic of spontaneous spoken speech. Children often do not speak well-formed and structured sentences. They speak in phrases with false starts, prolongation, repetition, interjections, filled and unfilled pauses. Rates of disfluency per word in English spontaneous speech among children have a varying

✉ Sheena Christabel Pravin
sheena.s@rajalakshmi.edu.in

M. Palanivelan
palanivelan.m@rajalakshmi.edu.in

[1] Department of ECE, Rajalakshmi Engineering College, Chennai, India

Published online: 11 February 2021

disfluency rate between 5 and 10% for natural conversations [14, 28]. There is also considerable variation across different speaking conditions and disfluency types [41]. There are both physiological and psychological interpretations for the manifestation of disfluencies [11, 15, 27]. Though often disfluencies are regarded as noise and sporadic speech junctures, disfluencies do show unique characteristics to some extent that aid in perfect modelling and classification of the same. Disfluency recognition and classification were weighed to be complex but such systems aid in the prognosis of respiratory system diseases [1] and speech disorders [12, 40] as well. Disfluency in case of speech disorders like stuttering is accompanied frequently by variation in breathing, vocalization, articulation, rhythm, speaking rate and accent of speech [3], which cause many complications in their accurate classification. For a speech therapist, the automatic speech disfluency classification can support as a pragmatic tool for prognosis of specific speech impairment, severity assessment and inspection of the therapy progression of the subjects with disfluent speech.

Speech acoustic features carry a straightforward relationship with the speaker's vocal anatomy characterized by one's pitch (F_0) which depicts the length and mass of a speaker's vocal cords [20]. Some of the valuable pitch-related speech acoustic features are pitch (F_0) values extracted from the speech preceding the boundary and following the boundary, the difference in F_0 across the boundary and F_0 derivatives. Extraction of prosodic cues from the segmented speech including spectral stability, stable spectral duration, the spectral centre of gravity, silence region before and after the filled pause has been experimented for disfluency classification [43]. Duration, pitch and energy features for each syllable were found to be valuable for edit or false starts in spontaneous speech [50]. Duration features like the duration of pauses, vowels and voiced segments could help detecting filled pauses, repetitions, false starts and repair disfluencies [42]. Apparently, for non-tonal languages, the pitch-based features were found to be less effective for the classification of filled pauses from the non-filled pauses. Further, the pitch feature extractor adds complexity to the front end of the speech recognizer. The average value of pitch, energy and their derivatives serve as key features in tracing disfluent regions. Natural language processing (NLP)-based disfluency detection and classification models [2, 26, 47–49] have also tried to classify disfluencies by processing the texts from the speech recognizers. In this work, the spontaneous disfluent speech signals are analysed for suitable feature retrieval and classification.

Based on the prevailing literature, the standard machine learning models like the hidden Markov models (HMMs) [23], Gaussian mixture models [36], conditional random field (CRF) models [9, 24], multi-layer perceptron (MLP) [46] were expended for disfluency classification. Tree models were used specifically in supervised learning of disfluencies [29, 42]. They recursively partition the input space and assign a label to the final node. Well-known tree models including classification and regression trees (CARTs) have been popularized in repetition disfluency detection [25, 42]. A key advantage of tree-based models is their fine interpretability, since the predictions are given by a set of rules. It is also common to use an ensemble of multiple trees such as the random forest [6] and XGBoost [8] to enhance the performance of the classifier on a trade-off with interpretability. With machine learning-based syndrome prognostics becoming prevalent and poignant in many facets of

our day-to-day lives [13, 31], the pivotal point of research has moved to ensemble models which have attracted research curiosity since their introduction [45]. They are powerful enough to dispense prediction accuracy on par with the deep learning models [33, 44]. The conventional deep learning networks comprise of a cascading hierarchical architecture with a layer-by-layer processing of features wherein the information processed by a layer is fed to the next layer for further dispensation, while the ensembles use an ordered arrangement of machine learning models [33]. Many deep learning architectures have been adopted for speech disfluency classification, including the deep belief networks [18], recurrent neural networks [38], convolutional neural networks (CNNs) [30], LSTM [35, 51]. Though traditional machine learning algorithms can be used in the classification of spontaneous speech disfluencies, they do not match the performance heights achieved by deep learning models. But an ensemble of the conventional machine learning models with the deep learning models conferred as the proposed hybrid model has shown to yield high classification accuracy and better statistical evaluation results. The key contributions of this paper are as follows:

- A Hybrid Deep Ensemble (HDE) model is proposed for speech disfluency classification. The HDE architecture is built with an ensemble of deep learning and machine learning models.
- The proposed HDE model is trained on the deep autoencoder-based latent feature vector and exhibits high classification accuracy and precision in speech disfluency classification over a sparse dataset. On statistical evaluation, the proposed hybrid model was observed to yield exemplary performance than the standalone baseline machine learning models, essentially above the deep learning models that demand massive data.
- The proposed model is shown to have lesser tuning parameters and decreased run time compared to the deep learning models.

From this point forward, the paper is organized as follows: Sect. 2 briefs on the disfluent speech corpus used for the classification task. Section 3 discusses on various disfluency characterizations. Section 4 elucidates the various speech acoustic parameters that are tapped for detecting speech disfluencies. Section 5 brings a detailed view of the proposed Hybrid Deep Ensemble, while Sect. 6 outlines the baseline experimentation and Sect. 7 elaborates the experimental results. Section 8 concludes the paper with the prospective future work.

2 Disfluent Speech Corpus

The disfluent speech data corpus embodies the taped English spontaneous speech samples from 27 bilingual children (14 boys and 13 girls), speaking Tamil (mother tongue) and English (second language), ranging in age between 4 and 7 years. Bilingual children exhibited many disfluencies when they rendered speech in their second language. For each of the subjects, parents signed an informed compliance before speech sample recordings. A head-saddled array

Table 1 Specimen queries

Vacations	When I Grow Up
1. When was your last vacation?	1. What do you aspire to be when you grow up?
2. Describe your favourite vacation spot	2. Will you help the needy?
3. With whom do you like to spend time with?	3. How will you spend your fortune?
4. What souvenirs do you buy for your friends?	4. Which is your dream country to pursue a career?
5. How long do you go on vacation?	5. What is your dream portfolio?

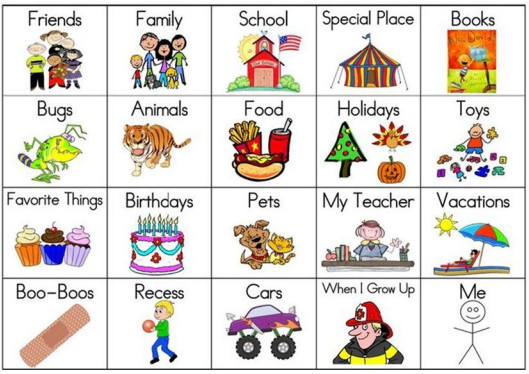

Fig. 1 Spontaneous speech stimulus material [17]

microphone with noise-cancellation features was maintained a couple of inches away from the child's mouth. Spontaneous speech from children was recorded at every subject's home in a room sans ambient noise to keep the child at ease and to bring down their anxiety. After warming up, the children were asked to speak out spontaneously on the topics of choice from the visual stimuli, an instance of which is displayed in Table 1 and Fig. 1. English spontaneous speech samples were taped for a total of 125 min, about 4.5 min per subject with one recording per subject.

All speech recordings were made at 16,000 Hz sampling rate using the Audacity tool in Windows 10 laptop. Additionally, the subjective ratings of fluency were manually recorded in both the languages spoken by every child, observant of secondary behaviours during their speech rendition in both the languages spoken by them. Annotations of the recordings were carried out using the PRAAT tool [4]. The disfluent segments of speech were then framed at a window length of 25 ms with an overlap window of 10 ms. A total of 488 acoustic features were collected from each speech frame. Constrained time alignment was exercised on the disfluent segments and their transcriptions together. Manual disfluency annotation was carried out for 5 different disfluencies, viz. filled pauses, word repetition, word-medial repetition, revision, prolongation following the annotation waylaid in [21].

3 Disfluency Characterizations

The spontaneous speech of individuals, even those who do not stutter, exhibits several disfluencies. Typical disruptions in fluency are characterized by filled pauses, repetition and its types such as word repetitions and intra-word or word-medial repetitions, revisions and prolongations. Disfluency characterization and classification can help the Speech Language Pathologist (SPL) in accurate diagnosis of the definite speech disorder and conclude on a suitable speech therapy.

3.1 Filled Pauses

Filled pauses (FPs) serve as discourse markers in spontaneous speech. FPs in spontaneous or conversational speech have a natural durational attribute of voice lengthening. Filled pauses exhibit nasal effect, profoundly exhibited by the filled pauses such as 'ah', 'uh', 'um', 'em' and 'hem'. These nasal sounds exhibit resonance characteristics influenced by lingual cavity characteristics, the velum and the nasal tract as well. Filled pauses are predominantly vowel sounds, which may or may not be followed by a nasal. Filled pauses have posed glitches to speech recognizers since they are often confused and recognized as small functional words.

3.2 Word Repetition

Repetition is a recurrent disfluency where speakers repeat their words while giving thought to the next word to be uttered. A repetition can either be an identical repetition, where speakers exactly repeat a word or phrase or a broken repetition, where they correct themselves using similar words. As per a study on repetition frequency statistics [47] given in Table 2, repetitions in spontaneous speech are frequent and hence need attention. The variants of repetition disfluency are briefed in the ensuing subsections.

Word repetition disfluency, which is simple repetitions observed in spontaneous speech encompasses a word (W1), an optional pause (SIL) followed by the second occurrence of the repeated word (RW1) and then the spontaneous speech, is continued. A template of word repetition is pictured in Fig. 2. It was observed that when a word was simply repeated, the repeated word (RW1) was observed to have shorter duration and a lowered pitch value.

Table 2 Repetition frequency statistics in spontaneous speech collection [36]

Speech corpus	Unitary word repetition (%)	Multiple word repetition (%)	Total no. of words
Switchboard	20.2	40	3.1million
Call home	10.5	22	181,000
Fisher	17.2	33	17.8 million

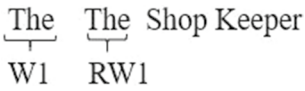

Fig. 2 Template for word repetition

3.3 Word-Medial Repetition

Word-medial repetition comprises of phonetic or syllabic repetitions made before completing one whole word, in which the 'phonation' or airflow is stopped within a word [16]. It also relates to single consonant repetition at the end of an utterance. The phonological arrangement of a syllable is styled best in terms of its principal elements: the onset, nucleus and coda as in Fig. 3. The onset is the set of consonants prior to the nucleus which is essentially a vowel and coda shapes the end of a syllable which is an optional consonant. The nucleus and coda collectively form the rhyme. The word-medial repetition can be at the onset (e.g. b-b-baby for baby) or at the nucleus (e.g. pe-en for pen) or at the coda (e.g. scientist-ist). Conventional word-medial disfluencies in subjects with stuttering disorder were observed at the onset of the syllable.

3.4 Revisions

Revisions signify the speech markings of a speaker who abandons the inception word of a spontaneous utterance and restarts all over again. Figure 4 is the typical example of revision with the abbreviations FS: false start, IP: interruption point, R: revision.

The principal difference between spontaneous speech and read speech is the use of false starts, wherein the speaker impedes the outflow of speech to restart the utterance all over again constraining revisions. Simple restarts in particular were without an inserted or substituted word which could be discerned acoustically via an analysis of word duration, pitch and other spectral features in the proximity of a pause. In our analysis, the false starts were spotted to be the most difficult type of disfluency to discern, since there was absolutely no correlation between the dropped-out word and the word that followed. However, the acoustic cues like the critical band energies, formants

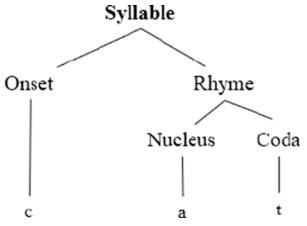

Fig. 3 Syllabic constituents

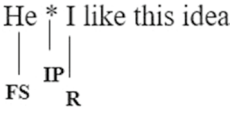

Fig. 4 An example of revision disfluency

and Mel-frequency cepstral coefficients (MFCCs) were found to improve the detection of false starts by the proposed model.

3.5 Prolongation

Lengthened speech segments in a word are termed as prolongations which is characterized by an involuntary lengthening of speech sound. A speech segment that lasts greater than 200 ms is termed a prolongation even if the prolongation was driven by the prosody of the discourse. Some authors consider the portion of the speech with a duration longer than 300 ms as prolongations [5]. An example of prolongation disfluency with P1: first prolongation and P2: second prolongation is signified in Fig. 5.

Prolongation can be utilized in the prediction of developmental stuttering. When early stuttering is predominated by prolongation, the chance of fluent speech recovery is lower than with the dominance of other disfluency characterizations [10]. Prolongation can be of a vowel, nasal, lateral, an approximant, a fricative and a plosive. In the experimental analysis, it was examined that the prolonged utterances have the same frequency structures for a longer period of time as the normal utterance.

4 Parametrization of Disfluencies

Conventionally, Speech Language Pathologists (SLPs) manually count the occurrence of disfluencies such as repetitions and prolongations in the stuttered speech to scale the severity index of stuttering. However, these forms of stutter ratings are subjective, uncertain, exhausting and prone to error. Therefore, machine learning-based automatic assessment of specific disfluencies and their recurrence would assist the SLP in agile diagnosis of speech pathology and choose a suitable therapy. It can also help the patient in his post-recovery phase.

4.1 Acoustic Features

Acoustic features characterized by the formant frequencies and energy features are of prime importance in discriminating simplest classes of disfluency phenomena, namely the filled pauses, revisions, repetitions and prolongations. Revisions were observed by averaging time duration with standard deviation and mean values of the first and second formants (F1 and F2). Acoustic features including the MFCCs were highly advantageous in analysing stuttered events in speech. Essentially, the features like the number of phones, syllables per word and energy slopes were proved to be greatly significant for marking the interruption points for identifying revision disfluencies.

Fig. 5 An example of prolongation disfluency

In this work, a total of 488 acoustic features were extracted in an empirically optimized window as in [32] which included the Bark band energy (BBE) in 22 critical bands during the onset and offset durations of the speech utterances. The MFCCs and its derivatives, the first two formants and their derivatives along with the statistical measures of the acoustic features, namely the mean, standard deviation, skewness and kurtosis, were also extracted. The afore-said four statistical functionals of the 122 features listed in Table 3 were computed, and together they expanse to a high-dimensional feature set comprising of 488 feature columns. The comparison of different classes based on the acoustic features is plotted in Fig. 6. The second formant frequency was observed to soar high for prolongation disfluency than the other disfluencies, while the revision disfluency exhibited the least of the second formant frequencies as shown in Fig. 6a. The revision and whole word repetitions exhibited lowered BBE than the other disfluencies as presented in Fig. 6c. Interestingly, the double delta MFCCs could distinguish prolongations from filled pause with two significant closely spaced spectral peaks as shown in Fig. 6d. In all of the disfluencies, the BBE at the offset was lower than at the onset of the utterance.

4.2 Dimensionality Reduction

The dimensionally high acoustic feature set led to cumbersome training of the hybrid ensemble. The higher the feature dimensionality, the harder it gets to visualize the feature set and their de-correlation. Some of these features were observed to be correlated, and hence redundant. Dimensionality reduction-based feature engineering helps in feature compression, reduced computation time and lowered redundancy that was resorted to use the tree ensembles like the random forests which were often used for feature selection. A large set of trees was constructed counter to a feature, and then, each feature's utility statistics was used to find the most relevant set of features. If a feature was often selected, it is most likely a pertinent feature to hold. Random forest naturally ranks the features by how well they decrease the impurity of each node called as the 'Gini' impurity. Nodes with less 'Gini' impurity were placed at the start of the trees, while nodes with higher impurity were positioned at the end of trees. Thus, by pruning trees below a particular node, a subset of the most important features was discerned followed by feature ranking.

The random forest model ranked the top 120 significant features out of 488 speech cues, which were chosen for model training. Due to space constraint, a feature ranking plot visualizing the 40 top ranking features is displayed in Fig. 7. The Bark band energy (BBE) extracted from the 22 critical bands has been observed to be the most significant feature followed by the Mel-frequency cepstral coefficients (MFCCs) and their derivatives. The first derivative of MFCC displays evidence on the rate of speech, and the second derivate of MFCC yields information on speech acceleration. Since these features reveal the dynamic characteristics of speech signal, they are deemed significant to categorise disfluencies into their respective classes. The first and the second formant frequencies and their statistical functionals were ranked next only to the MFCCs.

Table 3 Acoustic features extracted for disfluency classification

Acoustic features	No. of feature coefficients	Feature description
BBE onset	22	Bark band energies at the onset of voiced segments from unvoiced segments
MFCC onset	12	Mel-frequency cepstral coefficients (MFCCs) at the onset of voiced segments from unvoiced segments
Δ MFCC onset	12	Delta Mel-frequency cepstral coefficients at the onset of voiced segments from unvoiced segments
ΔΔ MFCC onset	12	Delta–delta Mel-frequency cepstral coefficients at the onset of voiced segments from unvoiced segments
BBE offset	22	Bark band energies at the offset of voiced segments
MFCC offset	12	Mel-frequency cepstral coefficients at the offset of voiced segments from unvoiced segments
Δ MFCC offset	12	Delta Mel-frequency cepstral coefficients at the offset of voiced segments from unvoiced segments
ΔΔ MFCC offset	12	Delta–delta Mel-frequency cepstral coefficients at the offset of voiced segments from unvoiced segments
F1	1	First formant frequency
Δ F1	1	Delta F1
ΔΔ F1	1	Delta–Delta F1
F2	1	Second formant frequency
Δ F2	1	Delta F2
ΔΔ F2	1	Delta–Delta F2

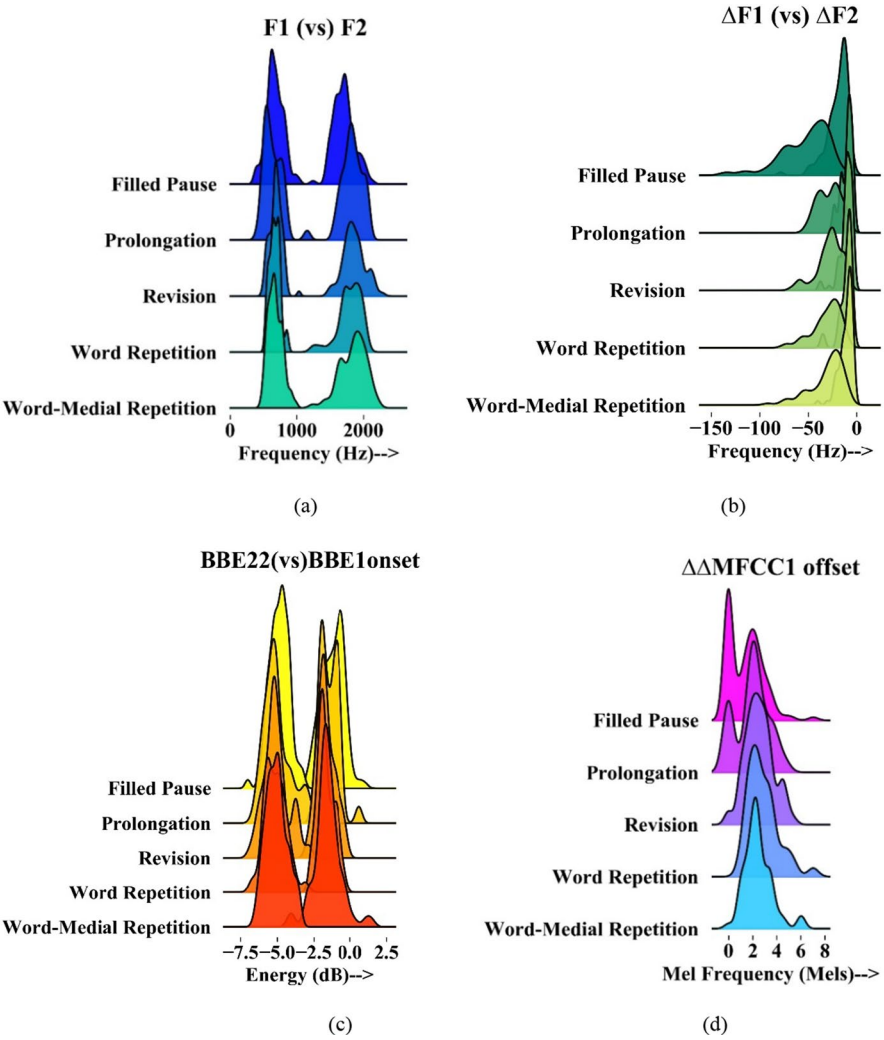

Fig.6 Comparison of disfluency classes with respect to acoustic features

5 The Proposed Hybrid Deep Ensemble

The proposed Hybrid Deep Ensemble (HDE) is the extension of the Super Learner Ensemble [45], augmenting the weights of the base learners, downplaying the loss function given the cross-validated score of each of the base learners. In the proposed work, an ensemble of conventional machine learning models and deep learning models such as the deep autoencoder and multi-layer perceptron was built to have diversity among the component learners for a strong generalization ability of the model. The deep autoencoder was introduced to extract the latent feature vector which is the compressed form of the input speech acoustic features that effectively

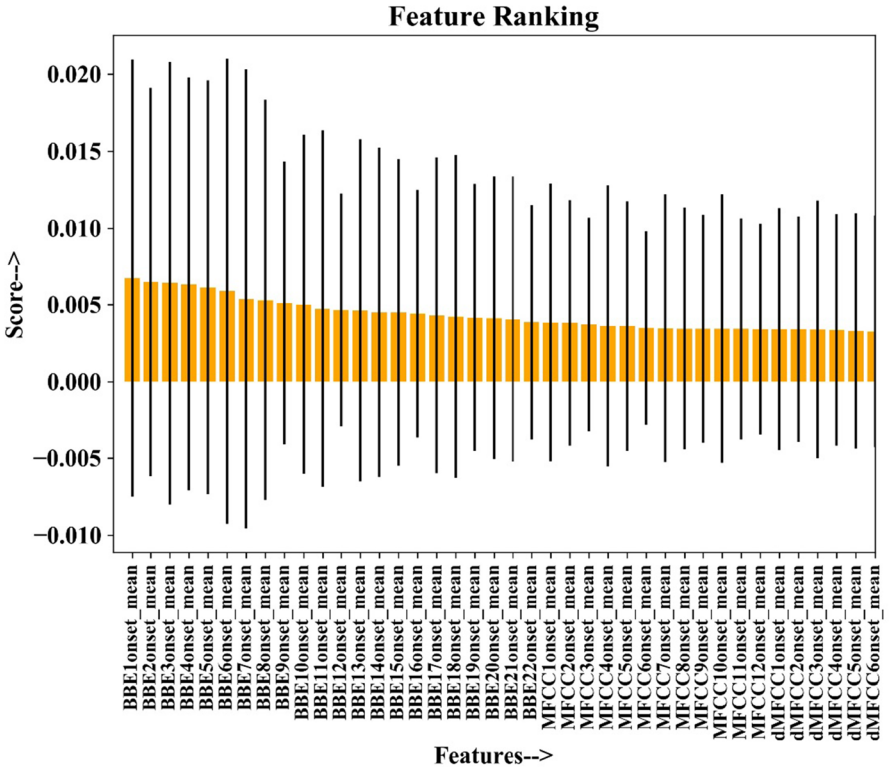

Fig.7 Feature ranking using random forest model

improve the classification performance. A detailed architecture of the proposed Hybrid Deep Ensemble model is given in Fig. 8.

5.1 DAE-Based Latent Feature Extraction

The HDE was trained on the latent features generated by the deep autoencoder (DAE), constructed with one input layer, five hidden layers and one output layer. The encoder part of the DAE was built with two layers indicated as hidden layer 1 and hidden layer 2 in Fig. 8. The bottleneck layer is indicated as the hidden layer 3, and further, the decoder part of the DAE was constructed with two layers marked as hidden layer 4 and hidden layer 5, respectively. The hidden layer 3 (i.e., the bottleneck layer) was built with lesser neurons to facilitate feature compression. The hidden layers comprise of 64 neurons each, excepting the bottle neck layer, which was built with just 10 neurons. The latent feature vector was derived from the DAE in an unsupervised fashion. The latent representation is a dimensionally reduced, compressed version of the input features with nonlinear makeover.

The first layer in the DAE being the input layer has 'd' units; 'd' represents the size of the dimensionality reduced speech acoustic feature vector. The input layer

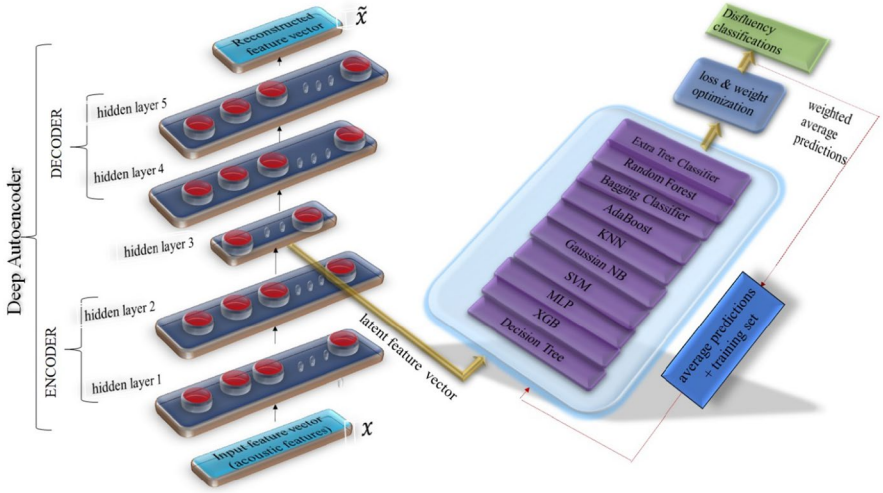

Fig. 8 The proposed Hybrid Deep Ensemble architecture

maps the input feature vector, $x \in R^d$ to the hidden layers, which is encoded to the compressed latent vector 'L', by a rectified linear unit (ReLU) activation function $\emptyset$, with the training weight matrix $W^{(1)}$ and bias vector $V^{(1)}$as given in Eq. (1).

$$L = \emptyset\left(W_t^{(1)}x + V^{(1)}\right) \tag{1}$$

where $\emptyset = \begin{cases} \max(0,x) & \text{if } x < 0 \\ x & \text{if } x > 0 \end{cases}$.

The deep autoencoder was implemented with the 'ReLU' activation function which takes care of the random initialization of the weights evading the vanishing gradient and the exploding gradient issues owing to random initializations. The decoder reconstructs the latent feature vector to $\tilde{x}$ expending the learned weight matrix $W_t^{(2)}$and bias vector $V^{(2)}$ as follows:

$$\tilde{x} = \emptyset\left(W_t^{(2)}x + V^{(2)}\right) \tag{2}$$

Since the DAE is assumed to have equal weights in this work to condense the tuning parameters, the restriction given in Eq. (3) was applied.

$$W_t^{(1)} = W_t^{(2)} = W \tag{3}$$

The tuning parameters of the AE now comprise of (W, $V^{(1)}$, $V^{(2)}$). The proposed DAE-based latent feature extraction is illustrated by a pseudocode given under Algorithm 1.

Algorithm 1: Pseudocode for the DAE-based Latent Feature Extraction

begin
  initialize input feature vector size 'd', training epochs, learning rate, number of hidden layers 'l', number of neurons in the hidden layers and the number of classes C
  build an AutoEncoder(AE) with 'd' input units and 'l' hidden layers each with 64 neurons;
  set the number of neurons at the latent hidden layer as 10;
  set the number of output units of the AE equal to 'd';
  initialize AE weights W and Biases $\mathbf{V^{(1)}}$, $\mathbf{V^{(2)}}$ to zero;
  for each training epoch
    for each batch of data
      compute feature reconstruction:
      $\tilde{\mathbf{x}}=\emptyset(\mathbf{W}.\ \emptyset(\mathbf{W}.\mathbf{x}+\boldsymbol{V}^{(1)})+\boldsymbol{V}^{(2)})$
      compute the loss function:
      $\boldsymbol{D}_{KL}(\boldsymbol{P}(\tilde{\boldsymbol{x}})|\boldsymbol{P}(\boldsymbol{x}))=-\sum_{\tilde{\boldsymbol{x}}\,\epsilon\,\mathbf{x}}\boldsymbol{P}(\boldsymbol{x})\,\mathbf{log}(\boldsymbol{P}(\boldsymbol{x})|\mathbf{P}(\tilde{\boldsymbol{x}}))$
      update AE Weights
    end
  end
  remove the output layer;
  extract the latent feature vector from the latent hidden layer;
end

The autoencoder learning parameter, $\varepsilon=(W, \mathrm{V}^{(1)}), V^{(2)})$, is learned using back-propagation algorithm by optimizing the Kullback–Leibler (KL) divergence loss function, measured between the probability distribution of the input $P(x)$ and the probability distribution of the reconstructed feature vector $P(\tilde{x})$ as shown in Eq. (4)

$$D_{KL}(P(\tilde{x})|P(x)) = -\sum_{\tilde{x}\,\mathrm{X}} P(x)\log\left(P(x)|\mathrm{P}(\tilde{x})\right) \tag{4}$$

The KL divergence loss measures the similarity of the predicted probability distribution $P(\tilde{x})$ with the desired target probability distribution $P(\tilde{x})$ in the probability space X. A 'null'-valued KL divergence loss indicates that the distributions are identical. This loss function aids the DAE model to approximate a more complex function for learning a dense feature representation to reconstruct the original input feature vector. The objective of training the DAE is to minimize the loss function as given in Eq. (5).

$$\arg\min_{W,V^{(1)},V^{(2)}}\left[D_{\mathrm{KL}}(P(\tilde{x})|P(x))\right] \tag{5}$$

The AE weights were updated with a learning rate '$\propto$', as given in Eqs. (6)–(8).

$$W_{\mathrm{new}} = W_{\mathrm{old}} - \propto \frac{\partial D_{\mathrm{KL}}(P(\tilde{x})|P(x))}{\partial W} \tag{6}$$

$$V^{(1)}_{\mathrm{new}} = V^{(1)}_{\mathrm{old}} - \propto \frac{\partial D_{\mathrm{KL}}(P(\tilde{x})|P(x))}{\partial V^{(1)}} \tag{7}$$

$$V_{\text{new}}^{(2)} = V_{\text{old}}^{(2)} - \propto \frac{\partial D_{\text{KL}}(P(\tilde{x})|P(x))}{\partial V^{(2)}} \tag{8}$$

The loss function is measured for each batch of data after which the DAE weights are updated. After a prolific feature reconstruction, the decoder was removed and the latent feature vector was extracted from the latent hidden layer for training the proposed Hybrid Deep Ensemble model.

5.2 HDE-Based Speech Disfluency Classification

The proposed Hybrid Deep Ensemble (HDE) was built with a total of ten base learners including nine machine learning (ML) models and one deep learning (DL) model. The ML base learners include the decision trees, XGBoost, support vector machine (SVM), Gaussian Naive Bayes classifier, *k*-nearest neighbours (k-NN), AdaBoost, bagging classifier, random forest, extra-trees classifier and the DL base learner multi-layer perceptron (MLP) which are incorporated into the Hybrid Deep Ensemble architecture. The tree models were included to provide extreme diversity in ensemble classification. The k-NNs, SVM and Gaussian Naive Bayes have been picked and chosen because of their variant classification ideologies. Thus, a combination of boosting and bagging algorithms was stacked one upon the other.

The proposed HDE was trained on the latent feature vector from the deep autoencoder. During the training phase, at each epoch the class prediction probabilities for the base learners were obtained. The weights and the loss function were optimized to get the average probability score across all the base learners by multiplying the prediction probabilities with the respective weights. When the loss value was observed lesser than that from the previous epoch, the average probabilities were appended to the feature set in the next training iteration of the ensemble to stabilize the loss function and the average probabilities obtained in the current epoch were saved.

Cross-validation strategy was used to assess the performance of the machine learning models on unexplored data, wherein the training dataset was split into '*k*' parallel sets. The number of cross-validation folds can influence the extent of under/overfitting of the model. With more folds, there is a substantial overlap in the training data between the folds leading to a potential over-fit and more runtime. Fivefold was used in this work as experimentation proved it to be a good balance of fit and runtime.

During the testing phase, the proposed model was put on pilot mode and the test instances were passed through the base learners and the average probability score across all the base learners was utilized to classify the disfluency labels. The pseudocode of the classification procedure using the proposed HDE is detailed under Algorithm 2.

Algorithm 2: Pseudocode for the Proposed Hybrid Deep Ensemble

```
begin
    Extract latent features from the deep autoencoder in an unsupervised fashion;
    for iteration in 1 to max epochs do
        split training data into k folds;
        for each fold do
            for each base-learner in the HDE ensemble do
                train the base-learner on the latent feature vector in the fold;
                get class probabilities from base learner on the validation set;
                build class prediction probabilities for the base learner;
            end
        end
        optimize weights and loss function with prediction labels and true labels;
        get average probability score across all the base-learners by multiplying predictions
         with respective weights;
        if the loss value is less than that from the previous epoch then
            append average probabilities to the feature set;
        else
            save average probabilities obtained in the current epoch;
            break for;
        end
    end
    for each instance in the test set do
        for each base-learner in the HDE ensemble do
            get class probabilities from base learner on the test set;
            build class prediction probabilities for the base learner;
        end
    end
    get average probability score across all the base-learners;
    classify the disfluency labels with the average probability score;
    generate the confusion matrix;
end
```

6 Experimentation with the Baseline Models

The performance of the proposed HDE was compared with individual machine learning base learners and deep learning networks like the MLP, DNN and Keras deep learning classifier against the proposed Hybrid Deep Ensemble model.

6.1 Stand-Alone Machine Learning Base Learners

The machine learning base learners embedded in the HDE architecture were tested individually using the Scikit-learn Python library [34] for the task of disfluency classification using identical loss function and evaluation metrics. The tuning parameters of the base learners are displayed in Table 4. The tree-based models have been constructed to have maximum depth, balanced class weights and 'maximum features' taken as the default value for the number of features. The SVM classifier was implemented with radial basis function and 'one-vs-rest' decision function. The AdaBoost classifier, bagging classifier, random forest and extra-trees classifier are

Table 4 Summary of the ML base learners and their tuning parameters

Sl. No	ML base learners	Tuning parameters
1	Decision tree	*Splitter:* best *Depth:* Maximum *Maximum features:* default no. of features *Class Weight:* Balanced
2	XGB	*Booster:* Tree-based models *Depth:*10 *Subsampling of observations:*1 *Column sampling by tree:* 1 *Learning rate:*0.2
3	SVM	*Kernel Coefficient: 'rbf'* *Decision function shape: 'ovr'* *gamma = 1/(No. of features * variance of the feature vector)* *C = 1*
4	Gaussian NB	Parameters were set to default settings
5	k-NN	*Neighbours:*5
6	AdaBoost	*Base estimator:* Decision Tree *No. of estimators:*50 *Learning rate:*1.0
7	Bagging classifier	*Base estimator:* One-vs-Rest classifier *No. of Base estimators:*10
8	Random forest	*Base estimator:* Decision Tree *Trees:*250 *Criterion:*'Gini' *Maximum Features:* default of features *Depth:* maximum
9	Extra-trees classifier	*Trees:*250 *Criterion:*'Gini' *Maximum features:* default no. of features *Depth:* maximum

ensembles themselves and were implemented with varying number of base estimators. The bagging classifier was tested with the one-vs-rest classifier, while the rest of the ensemble models were tested with the default decision tree base estimator.

6.2 Deep Learning Networks

Deep learning networks were also experimented to establish standards for performance comparison. In our investigation with few deep learning architectures such as the multi-layer perceptron (MLP), deep neural network (DNN) and Keras deep learning classifier (KDLC) [7], some tuning parameters were modified to achieve better performance indicative of the innate-feature engineering capabilities of the deep learning networks. In Table 5, various tuning parameters of the experimented deep learning networks are displayed in detail. All the baseline deep learning networks were run for 100 epochs, measuring the model loss using the cross-entropy loss function with different batch sizes which typically had no impact on the

Table 5 Tuning parameters of deep learning networks

Tuning parameters for the deep learning architecture	Deep learning architectures		
	MLP	DNN	KDLC
Feature standardization	Standard scaler	Standard scaler	Max absolute scaler
Epochs	100	100	100
Learning rate	0.001	0.01	0.1
Maximum steps per epoch	200	200	50
Batch size	32	50	50
Train/test split	80:20	80:20	80:20
Cross-validation strategy	K-fold of 5 splits with shuffle split	K-fold of 5 splits with shuffle split	Stratified K-fold with fivefold CV
Loss function	Cross-entropy	Cross-entropy	Cross-entropy
Activation	ReLU	ReLU, Softmax	ReLU, Softmax
Optimizer	LBFGS	Adam	SGD
Hidden layers	3	10	3
Dense layer1	30 neurons, Activation: ReLU	10	64 neurons, Activation: ReLU
Dense layer2	30 neurons, Activation: ReLU	20	8 neurons, Activation: ReLU
Drop out	0.1	0.2	0.2
Output layer	5 Nodes Activation: Softmax	5 Nodes Activation: Softmax	5 Nodes Activation: Softmax

classification accuracy. Each of the model was verified with fivefold cross-validation. The ReLU activation function was chosen for all the models for faster learning and quick convergence. The 'softmax' layer was appended at the output layer of the DNN and KDLC to list out probabilities pertaining to the potential labels for classification. Each of the models were trained with different learning rates which were found apt for each of the models on an experimental basis. Also, diversified gradient optimizers, namely the limited-memory Broyden–Fletcher–Goldfarb–Shanno (LBFGS), the Adam and stochastic gradient descent (SGD) optimizers, were applied to the three models, respectively, which were individually tested for quick convergence.

The number of hidden layers and neurons was variably included in the models based on the dense nature intended for the respective model. Each of the models was built with 5 nodes at the output layer since the disfluency classification labels amount to five.

7 Experimental Results

The proposed Hybrid Deep Ensemble (HDE) model was compared with the standalone ML base learner models, deep learning networks with respect to statistical scores such as the F1-score, zero–one loss also called as the misclassification loss function, Jaccard similarity index (JSI) score and the Cohen's kappa coefficient (κ) to approximate the generalization inaccuracies of the models and to measure the proficiency of the models to categorize different disfluency classes measured by Eqs. (9)–(12)

$$\text{F1 - score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

$$\text{Zero - one - loss} = \begin{cases} 0 \text{ if PL} = \text{AL} \\ 1 \text{ if PL} \neq \text{AL} \end{cases} \tag{10}$$

$$\kappa = \frac{\text{AL} - \text{PL}}{(1 - \text{PL})}; \quad 0 < \kappa < 1 \tag{11}$$

$$\text{JSI(PL, AL)} = \frac{|\text{PL} \cap \text{AL}|}{|\text{PL} U \text{AL}|}; \quad 0 < \text{JSI(PL, AL)} < 1 \tag{12}$$

where AL: the actual label and PL: the predicted label.

The proposed HDE model achieved statistically significant scores, viz. lower zero–one loss, higher JSI score and higher Kappa's coefficient score. The HDE outperforming the machine learning models and the other deep learning networks signposts the impact of latent features extracted from the deep autoencoder on the statistical scores and the classification accuracy as well. Though deep learning models performed better than the conventional models such as the SVM, Gaussian NB,

Table 6 Statistical evaluation of the classifiers

Classifier	Model description	F1-score	Zero–one loss	Jaccard score	Cohen's Kappa coefficient	Run time (min)
Proposed HDE	HDE	0.97	0.03	0.94	0.95	1.45
DNN	DLN	0.89	0.1	0.81	0.86	4.25
KDLC	DLN	0.88	0.099	0.81958	0.87	4.09
MLP	DLN	0.86	0.106	0.807	0.85	1.55
DT	MLM	0.86	0.1266	0.775	0.83	0.59
XGB	MLBA	0.93	0.066	0.875	0.91	1.02
SVM	MLM	0.93	0.08	0.851	0.89	1.25
Gaussian NB	MLM	0.62	0.35	0.47	0.53	0.58
k-NN	MLE	0.65	0.325	0.51	0.57	0.52
AdaBoost	MLE	0.32	0.50	0.5092	0.57	1.04
Bagging classifier	MLE	0.92	0.06	0.875	0.91	1.20
Random forest	MLE	0.92	0.066	0.875	0.91	1.24
Extra-trees classifier	MLE	0.94	0.053	0.898	0.93	1.29

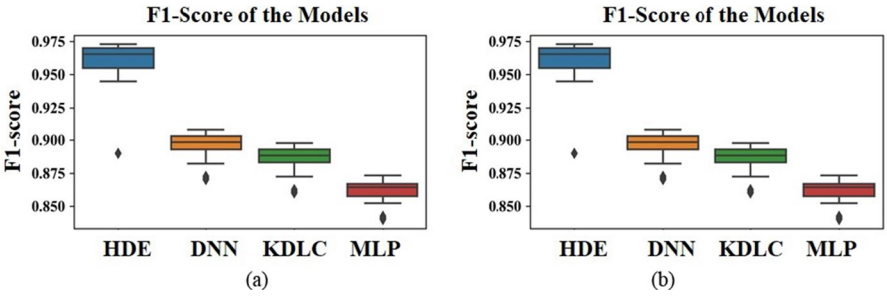

Fig. 9 (**a**) Average Kappa coefficients. (**b**) Average F1-score of the models

k-NN and much better than the AdaBoost classifier, it was interestingly noted that the ensemble machine learners, namely the decision trees, random forest, bagging classifier and the XGBoost algorithm, excelled the deep learners in exhibiting lower generalization loss and higher F1-score which is the harmonic mean of the precision and recall of the models.

The model descriptions abbreviated in Table 6 are as follows: HDE: Hybrid Deep Ensemble, DLN: deep learning network, MLM: machine learning model, MLBA: machine learning boosting algorithm, MLE: machine learning ensemble. The Kappa coefficients and F1-scores for few of the models under comparison are presented graphically in Fig. 9a, b.

Classification performance of the proposed model against the baselines was assessed by measuring average validation and test accuracies along with their weighted precision and recall average scores, taken across 50 trials and presented in Table 7. The fivefold cross-validation report of the proposed HDE model and the baselines is graphically presented in Fig. 10a. The average test accuracy of the models is presented in Fig. 10b. Though the classification accuracy of few of the machine learning base learners was observed to soar high with the training data, they scored less on the test-set classification. The proposed HDE model yielded the maximum test classification accuracy, maintaining the precision and recall standards as well than the rest of the experimented baselines.

The classification performance of the models is displayed in the form of their true positive and true negative classification accuracy as confusion matrices shown in Fig. 11a–d.

With a total of 980 disfluencies in the sparse dataset, 20% of the data, viz. 196 samples, were taken as test instances. The proposed hybrid model could classify all the 66 filled pauses, 36 revisions, 28 word repetitions accurately without any classification error, while 46 out of 49 word-medial repetitions and 14 out of 17 prolongations were rightly classified by the proposed model, yielding an overall average classification accuracy of 97%. Due to the conventional 'random_state' hyper-parameter set to 1 in all the models for an arbitrary split in the data samples from different classes in the train/validation/test set. Thus, the number of test samples under each disfluency class varies thinly in the confusion matrices of all the evaluated models but the total number of test samples amount to 196 in every model.

The receiver operating characteristics (ROCs) of the classifiers were plotted to show the area under the precision–recall curve of each model as shown in Fig. 12a–d. ROC tests the sensitivity of the proposed model in terms of true positive

Table 7 Accuracy, precision, recall of the classifiers

Classifier	Validation accuracy (%)	Test accuracy (%)	Weighted precision average	Weighted recall average
Proposed HDE	98.08	97.00	0.97	0.97
DNN	94.57	90.00	0.90	0.90
KDLC	89.63	89.38	0.90	0.88
MLP	87.22	85.21	0.86	0.85
DT	100	87.15	0.88	0.87
XGB	100	93.09	0.94	0.93
SVM	100	92.12	0.93	0.92
Gaussian NB	69.32	64.02	0.66	0.64
k-NN	86.12	60.03	0.59	0.60
AdaBoost	69.06	67.02	0.72	0.67
Bagging classifier	100	93.00	0.94	0.93
Random forest	100	93.00	0.94	0.93
Extra-trees classifier	100	95.02	0.95	0.95

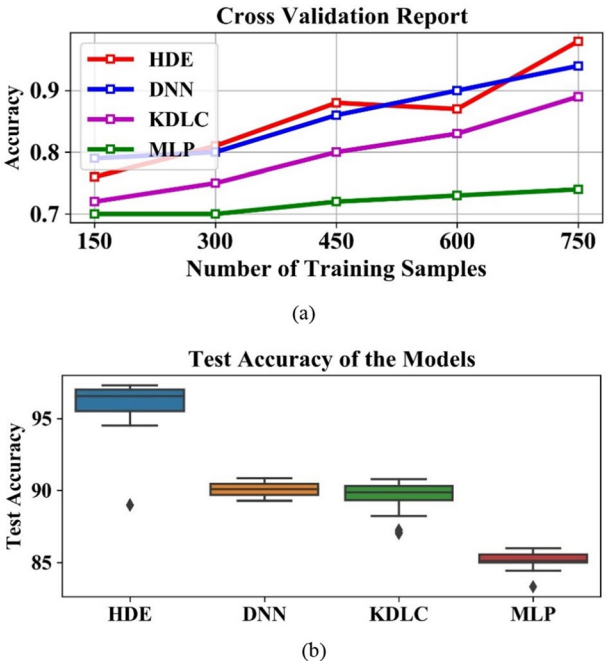

Fig.10 (**a**) Cross-validation report and. (**b**) Test accuracy of the models

rate versus the false positive rate. Compared to the baseline models, the superior diagnostic proficiency of the proposed model is evident in terms of its higher AUC.

A comparative study of the classifiers in the existing literature is presented in Table 8 with the abbreviations, *P*: precision, *R*: recall, *F*: F-score.

The efficacy of various classifiers in the literature with respect to the common evaluation metrics, namely classification accuracy, precision, recall, F-score, missed detection rate as available from the existing literature presented in Table 8, indicates the superior performance of the proposed Hybrid Deep Ensemble model than the recent models proposed for speech disfluency classification. The HDE when implemented on the UCLASS dataset [19] and the Fluency Bank dataset [37] performs slightly lower in accuracy than its performance on the self-annotated disfluency corpus but is still observed to be better than the existing models in the literature. Precision and recall values being high for the proposed model indicate its notable true positive classification.

8 Conclusions and Future work

In this paper, a Hybrid Deep Ensemble (HDE) model is proposed for speech disfluency classification on the available sparse data. HDE is an ensemble of proficient machine learning and deep learning base learners that are trained on the latent representation of the speech acoustic features. The deep autoencoder aids

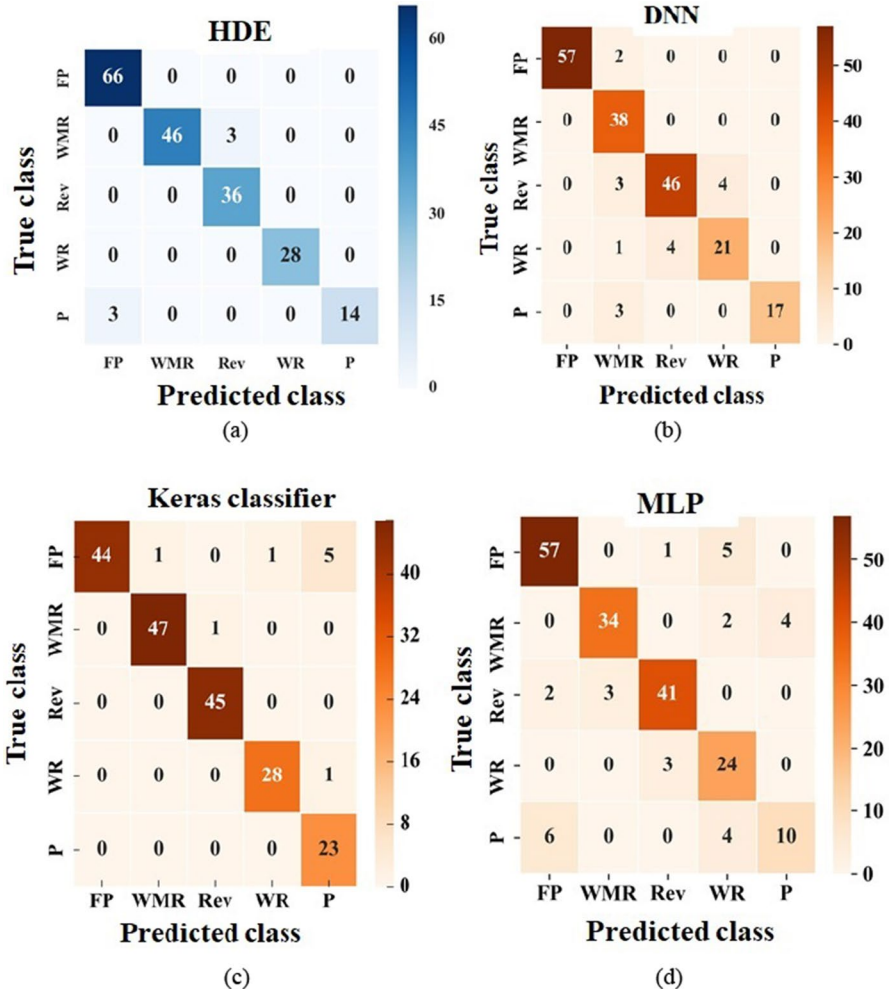

Fig. 11 Confusion matrix of (**a**) proposed HDE, (**b**) DNN, (**c**) KDLC, (**d**) MLP

in the compression of the acoustic features, which on reaching a desirable feature reconstruction yields a latent feature vector from its bottle neck layer, on which the proposed model is trained. The proposed HDE has better classification accuracy, higher Kappa coefficient and Jaccard similarity scores than the stand-alone machine learning and deep learning base learners when tested on the self-annotated disfluency corpus used in this study and also on the conventional datasets such as the UCLASS dataset on stuttered speech and the Fluency Bank dataset. The proposed model has proved to be on par with the performance of the deep learning networks, viz. multi-layer perceptron, deep neural network and Keras deep learning classifier. Significantly, the proposed HDE is adaptable with fewer hyper-parameters and performs well on a sparse dataset. High classification

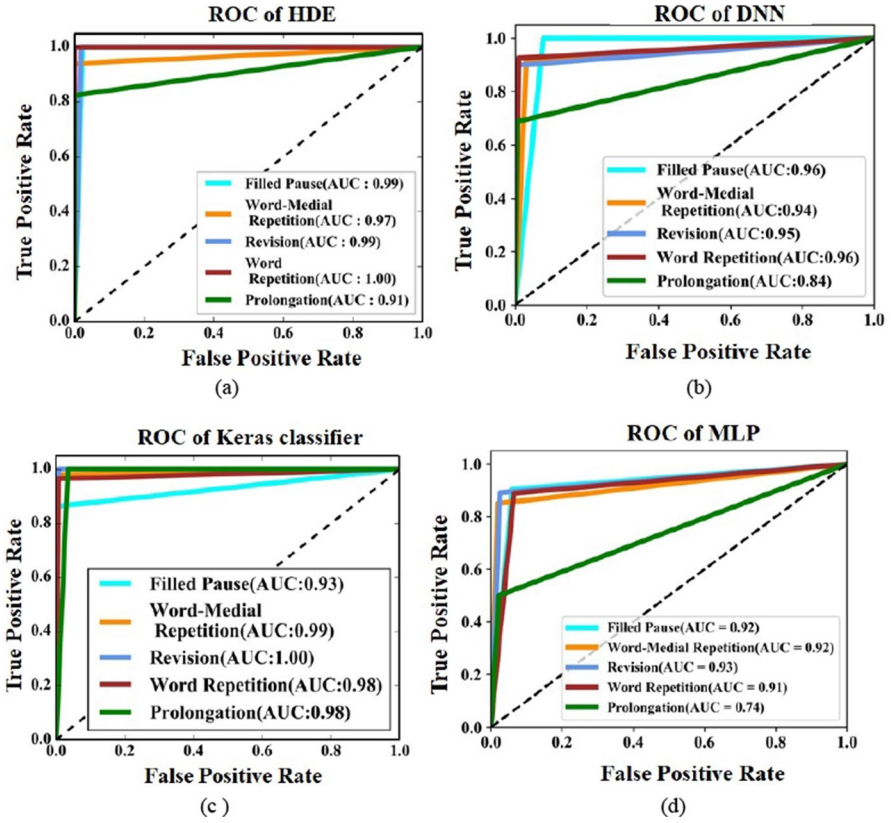

Fig. 12 ROC of (**a**) proposed HDE, (**b**) DNN, (**c**) KDLC, (**d**) MLP

accuracy with a sparse dataset makes the proposed model advantageous for the SLPs to use it for the patients under test by collecting few speech samples. Thus, the proposed model is easily trained unlike the deep learning architectures that demand vast data and elaborate hyper-parameter tuning. Further, the proposed model is advantageous in allowing transparency into results and has relatively faster convergence with good performance in the task of disfluency classification. The frequency count of each disfluency aids in publishing the fluency score of the subjects which can serve as a self-improvement aid for the children with speech disorders.

In future, we aspire to bring out a handy and easy-to-deploy tool with the proposed model for the Speech Language Pathologists for swift diagnosis of specific speech/language impairments. Also, the authors intend to resynthesize disfluent

Table 8 Comparison of classifiers based on their evaluation metrics

Classifiers	Dataset	Number of disfluent samples	Disfluency detected	Evaluation metrics (%)		
Deep residual network With bidirectional long short-term memory [22]	University College London's Archive of Stuttered Speech (UCLASS) Release One [19] dataset	800 disfluent audio samples	Sound repetition, word repetition, phrase repetition, revision, interjection, prolongation	Average missed rate:10.03 Average accuracy:91.15		
DNN + Audio Span Features + Signal features [39]	Adult-who-Stutter sub-dataset of Fluency Bank [37]	1429 speech utterances	Filled pause, single word repetition, multi-repetitions, phrase repetitions, revisions	P:86.4, error rate:100		
Proposed HDE	Self-annotated disfluency corpus	980 disfluent speech samples	Filled pause	P:96	R:100	F:98
			Word-medial repetition	P:100	R:94	F:97
			Revision	P:92	R:100	F:96
			Word repetition	P:100	R:100	F:100
			Prolongation	P:100	R:82	F:90
			Average accuracy:97.00			
Proposed HDE	University College London's Archive of Stuttered Speech (UCLASS) release one dataset [19]	756 disfluencies	Filled pause	P:92	R:83	F:87.5
			Word-medial Repetition	P:95	R:92	F:93.5
			Revision	P:94	R:100	F:97
			Word repetition	P:100	R:100	F:100
			Prolongation	P:95	R:83	F:89
			Average accuracy:93.25			

Table 8 (continued)

Classifiers	Dataset	Number of disfluent samples	Disfluency detected	Evaluation metrics (%)		
Proposed HDE	Adult-who-Stutter sub-dataset of Fluency Bank [37]	1129 disfluencies	Filled pause	P:91	R:81	F:86
			Word-medial repetition	P:98	R:94	F:96
			Revision	P:97	R:94	F:95.5
			Word repetition	P:100	R:100	F:100
			Prolongation	P:99	R:88	F:93.5
			Average accuracy:94.11			

speech into fluent renditions using the generative models such as the variational autoencoders and the generative adversarial networks in future.

Acknowledgements The authors gratefully acknowledge the anonymous reviewers for their valuable comments and suggestions which helped us improve the manuscript. This project is funded by AICTE, India, under the Research Progress Scheme (RPS). The Grant Reference No. is: 8-40/RIFD/RPS/Policy-1/2017-18, dated 15 March 2019. The authors are the joint investigators of the project.

Data Availability The disfluent speech dataset generated and analysed during the current study is available from the corresponding author on reasonable request.

Compliance with ethical standards

Conflict of interest The authors alone are responsible for the content and writing of the paper, and they report no conflict of interest.

References

1. R. Behroozmand, F. Almasganj, Optimal selection of wavelet-packed-based features using genetic algorithm in pathological assessment of patients' speech signal with unilateral vocal fold paralysis. Comput. Biol. Med. **37**(1), 474–485 (2007)
2. M. Black, J. Tepperman, S. Lee, P. Price, S. Narayanan, Automatic detection and classification of disfluent reading miscues in young children's speech for the purpose of assessment. Proc. INTERSPEECH **2007**, 206–209 (2007)
3. O. Bloodstein: A handbook on stuttering, San Diego, CA, p. 178–181 (1995).
4. P. Boersma, D. Weenink, PRAAT: doing phonetics by computer [Computer program], Version 5.3.51. http://www.praat.org/ (2013), Accessed 2 Jan 2019.
5. A. Braun, A. Rosin, On the speaker-specificity of hesitation markers, in *Proceedings of the 18th International Congress of Phonetic Sciences*, U.K., 0731.1-5 (2015).
6. L. Breiman, Random forests. Mach. Learn. **45**(1), 5–32 (2001)
7. P.W.D. Charles: Project Title. GitHub repository, https://github.com/charlespwd/project-title (2013), Accessed 03 Jan 2020.
8. T. Chen, C. Guestrin, XGBoost: A scalable tree boosting system, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, USA, p. 785–794 (2016).
9. E. Cho, T.H. Ha, A. Waibel, CRF-based disfluency detection using semantic features for German to English spoken language translation, in *Proceedings of International Workshop on Spoken Language Translation*, Germany (2013).
10. E.G. Conture, *Stuttering*, 2nd edn. (Prentice-Hall, Englewood Cliffs, 1990).
11. M. Corley, L.J. MacGregor, D.I. Donaldson, It's the Way that you, er, say it: hesitations in speech affect language comprehension. Cognition **105**(3), 658–668 (2007)
12. A. Czyzewski, A. Kaczmarek, B. Kostek, Intelligent processing of stuttered speech. J. Intell. Inf. Syst. **21**(2), 143–171 (2003)
13. K.C. Fraser, J.A. Meltzer, F. Rudzicz, Linguistic features identify Alzheimer's disease in narrative speech. J. Alzheimer's Disorder **49**(2), 407–422 (2016)
14. J.E. Fox Tree, The effects of false starts and repetitions on the processing of subsequent words in spontaneous speech. J. Memory Lang. **34**(1), 709–738 (1995)
15. S.H. Fraundorf, D.G. Watson, The disfluent discourse: effects of filled pauses on recall. J. Memory Lang. **65**(2), 161–175 (2011)
16. B. Guitar, T. J. Peters: Stuttering: an integrated approach to its nature and treatment. Baltimore, (1998).
17. L. Guo, J.B. Tomblin, V. Samelson, Speech disruptions in the narratives of English-speaking children with specific language impairment. J. Speech Lang. Hearing Res. **51**(3), 722–738 (2008)

18. G.E. Hinton, Y.W. Teh, A fast learning algorithm for deep belief nets. Neural Comput. **18**(1), 1527–1554 (2006)
19. P. Howell, S. Davis, J. Bartrip, The university college London archive of stuttered speech (UCLASS). J. Speech Lang. Hear. Res. **52**(1), 556–569 (2009)
20. T. Hudson, G. de Jong, K. McDougall, P. Harrison, F. Nolan, F0 statistics for 100 young male speakers of standard southern British English, in *Proceedings of the 16th International Congress of Phonetic Sciences*, Germany, p. 1809–1812 (2007).
21. F.S. Juste, C.R. Furquim de Andrade, Speech disfluency types of fluent and stuttering individuals: age effects, international journal of phoniatrics, speech therapy and communication. Pathology **63**(2), 57–64 (2011)
22. T. Kourkounakis, A. Hajavi and A. Etemad, Detecting multiple speech disfluencies using a deep residual network with bidirectional long short-term memory, in *Proceedings of ICASSP2020*, Spain, p. 6089–6093 (2020).
23. Y. Liu, E. Shriberg, A. Stolcke, M. Harper, Comparing HMM, maximum entropy, and conditional random fields for disfluency detection, in Proceedings of INTERSPEECH 2005, Portugal, p. 3313–3316 (2005).
24. Y. Liu, A. Stolcke, E. Shriberg, M. Harper, Using Conditional Random Fields for sentence boundary detection in speech, in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, USA, p. 451–458 (2005).
25. W. Loh, Classification and regression trees. WIREs Data Min. Knowl. Discov. **1**(1), 14–23 (2011)
26. P.J. Lou, M. Johnson, Disfluency detection using a noisy channel model and a deep neural language model, in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vol. 2: Short Papers, Canada, p. 547–553 (2017).
27. K. McDougall, M. Duckworth, Profiling fluency: an analysis of individual variation in disfluencies in adult males. Speech Commun. **95**(1), 16–27 (2017)
28. H. Maclay, C.E. Osgood, Hesitation phenomena in spontaneous English speech. WORD **15**(1), 19–44 (1959)
29. H. Medeiros, H. Moniz, F. Batista, I. Trancoso, L. Nunes, Disfluency detection based on prosodic features for university lectures, in *Proceedings of INTERSPEECH'2013*, France, p. 2629–2633 (2013).
30. J. Mekyska, B. Beitia, N. Barroso, A. Estanga, M. Tainta, M. Ecay-Torres, Advances on automatic speech analysis for early detection of Alzheimer Disease: a non-literal multi-task approach. Curr. Alzheimer Res. **15**(2), 139–148 (2018)
31. S.O. Orimaye, J.S. Wong, C.P. Wong, Deep language space neural network for classifying mild cognitive impairment and Alzheimer-type dementia. PLoS ONE **13**(11), 1–31 (2018)
32. J.R. Orozco-Arroyave, J.C. Vásquez-Correa, J.F. Vargas-Bonilla, R. Arora, N. Dehak, P.S. Nidadavolu, H. Christensen, F. Rudzicz, M. Yancheva, H. Chinaei, A. Vann, N. Vogler, T. Bocklet, M. Cernak, J. Hannink, E. Nöth, NeuroSpeech: an open-source software for Parkinson's speech analysis. Digital Signal Process. NeuroSpeech **77**(1), 207–221 (2017)
33. A. Ortiz, J. Munilla, J.M. Gorriz, J. Ramırez, Ensembles of deep learning architectures for the early diagnosis of the Alzheimer's disease. Int. J. Neural Syst. **26**(7), 1650025 (2016)
34. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**(1), 2825–2830 (2011)
35. M. Pishgar, F. Karim, S. Majumdar, H. Darabi, Pathological voice classification using Mel-Cepstrum vectors and support vector machine, in *Proceedings of 2018 IEEE International Conference on Big Data*, USA, p. 5267–5271, (2018).
36. V. Rangarajan, S. Narayanan, Analysis of disfluent repetitions in spontaneous speech recognition, in *Proceedings of 14th European Signal Processing Conference*, Italy, p. 1–5 (2006).
37. N.B. Ratner, B. MacWhinney, Fluency bank: a new resource for fluency research and practice. J. Fluency Disorders **56**(1), 69–80 (2018)
38. M. Reisser, Recurrent Neural Networks in speech disfluency detection and punctuation prediction. Master's Thesis at the Department of Informatics, Interactive Systems Lab (ISL), Institute of Anthropomatics and Robotics, Karlsruhe Institute of Technology, p. 50–60 (2015).

39. R. Riad, A.C. Bachoud-Lévi, F. Rudzicz, E. Dupoux, Identification of primary and collateral tracks in stuttered speech, in *Proceedings of the 12th Language Resources and Evaluation Conference*, p. 1681–1688 (2020).
40. X. Shao, J. Barker, Stream weight estimation for multistream audio-visual speech recognition in a multispeaker environment. Speech Commun. **50**(1), 337–353 (2008)
41. E. Shriberg, Preliminaries to a theory of speech disfluencies, Ph.D. thesis, University of California, Berkeley, CA, (1994).
42. E. Shriberg, R. Bates, A. Stolcke, A Prosody only decision tree model for disfluency detection, in *Proceedings of Eurospeech'97*, Greece, p. 2383–2386 (1997).
43. F. Stouten, J. Duchateau, J.P. Martens, P. Wambacq, Coping with disfluencies in spontaneous speech recognition: acoustic detection and linguistic context manipulation. Speech Commun. **48**(1), 1590–1606 (2006)
44. G. Thomas Dietterich, Ensemble Methods in Machine Learning, in: Proceedings of the First International Workshop on Multiple Classifier Systems, Italy, 1–15 (2000).
45. M.J. Van der Laan, E.C. Polley, A.E. Hubbard, Super Learner, U.C. Berkeley Division of Biostatistics Working Paper Series, Working Paper 222 (2007).
46. B. Villegas, K. M. Flores, K. José Acuña, K. Pacheco-Barrios and D. Elias, A novel stuttering disfluency classification system based on respiratory biosignals, in *Proceedings of 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Germany, p. 4660–4663 (2019).
47. F. Wang, W. Chen, Z. Yang, Q. Dong, S. Xu, B. Xu, Semi-supervised dis-fluency detection, in *Proceedings of the 27th International Conference on Computational Linguistics*, USA, p. 3529–3538 (2018).
48. S. Wang, W. Che, Y. Zhang, M. Zhang, T. Liu, Transition-based dis-fluency detection using LSTMs, in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Denmark, p. 2785–2794, (2017).
49. S. Wang, W. Che, Q. Liu, P. Qin, T. Liu T., W.Y. Wang, Multi-Task self-supervised learning for disfluency detection, in *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, AAAI 2020, p. 9193–9200 (2020).
50. J.F. Yeh, C.H. Wu, Edit disfluency detection and correction using a clean-up language model and an alignment model. IEEE Trans. Audio Speech Lang. Process. **14**(5), 1574–1582 (2006)
51. V. Zayats, M. Ostendorf, H. Hajishirzi, Disfluency detection using a bidirectional LSTM, in *Proceedings of INTERSPEECH2016*, USA, 2523-2527 (2016).