http://dx.doi.org/10.1016/j.neunet.2023.10.018

Automatic Selection of Spoken Language Biomarkers for Dementia Detection

Xiaoquan Ke[a], Man Wai Mak[a,*], Helen M. Meng[b]

[a]*Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong SAR*
[b]*Department of Systems Engineering and Engineering Management, The Chinese University of Hong Kong, Hong Kong SAR*

Abstract

This paper analyzes diverse features extracted from spoken language to select the most discriminative ones for dementia detection. We present a two-step feature selection (FS) approach: Step 1 utilizes filter methods to pre-screen features, and Step 2 uses a novel feature ranking (FR) method, referred to as dual dropout ranking (DDR), to rank the screened features and select spoken language biomarkers. The proposed DDR is based on a dual-net architecture that separates FS and dementia detection into two neural networks (namely, the operator and selector). The operator is trained on features obtained from the selector to reduce classification or regression loss. The selector is optimized to predict the operator's performance based on automatic regularization. Results show that the approach significantly reduces feature dimensionality while identifying small feature subsets that achieve comparable or superior performance compared with the full, default feature set. The Python codes are available at `https://github.com/kexquan/dual-dropout-ranking`.

Keywords: feature selection, feature ranking, dementia detection, spoken language biomarkers

*Corresponding author
Email addresses: `xiaoquan.ke@connect.polyu.hk` (Xiaoquan Ke), `enmwmak@polyu.edu.hk` (Man Wai Mak), `hmmeng@cuhk.edu.hk` (Helen M. Meng)

1. Introduction

Dementia is a severe cognitive impairment that may seriously affect the health and daily lives of the afflicted individuals. The greatest known risk factor for dementia is increasing age, and the most common form of dementia is Alzheimer's Disease (AD). According to a report from the World Health Organization,[1] more than 55 million people live with dementia worldwide, and there are nearly 10 million new cases every year. In 2019, the estimated global societal cost of dementia was $1.3 trillion, and these costs are expected to surpass $2.8 trillion by 2030. This has a huge impact on the quality of life, not only for individuals with dementia but also for their families and caretakers. Fortunately, with effective detection of early dementia, disease-modifying medications and interventions are possible. Early detection of dementia will facilitate intervention to slow disease progression.

Dementia can be diagnosed through several means, including neuropsychological assessments, brain scans, blood tests, etc. These diagnosis methods are generally intrusive and costly. Dementia also manifests itself as spoken language deficits. Studies had found that dementia-induced language impairment could be found in patients years before the disease was diagnosed [1], and patients with progressive cognitive decline exhibit subtle linguistic impairment even in the pre-symptomatic stages of the disease [2]. These findings suggest that dementia can be detected using spoken language processing (SLP) techniques.

1.1. Related Works

Recently, automatic detection of dementia through speech and language analyses has gathered attention in the research community. Some studies investigated different types of speech-based features that contain a variety of acoustic characteristics of the speakers for dementia detection. More recently, Haider *et al.* [3] compared different types of paralinguistic features – including eGeMAPS [4], ComParE 2013 [5], Emobase [5], and MRCG [6] – for dementia detection. As the paralinguistic features are high-dimensional, Pearson's correlation (PeaCorr) tests were performed to reduce the feature dimensions. Nasreen *et al.* [7] distinguished AD patients from healthy control of similar age using two types of speech-based features: interactional features

[1] `https://www.who.int/news-room/fact-sheets/detail/dementia`

and acoustic features. The former characterizes the temporal and interactional aspects of conversations; the latter characterizes the acoustics of speakers using pitch, amplitude, energy, and cepstral coefficients (MFCC). They achieved 87% accuracy using the interactional features alone. In addition to eGeMAPS features, Gauder *et al.* [8] investigated different speech-based embeddings for the automatic detection of AD. The speech-based embeddings are high-level representations extracted from TRILL [9], Allosaurus [10], and wav2vec 2.0 [11] models.

In addition to speech-based features, transcription-based features have also been used for dementia detection [12, 13, 14, 15, 16]. The transcription-based features are extracted from either the automatic or manual transcriptions, which capture the semantic, syntactic, and lexical aspects of the speaker's utterances. For example, Qiao *et al.* [17] combined disfluency features and linguistic complexity features for AD detection. The disfluency features (silent pauses, speed of articulation, filled pauses, and pronunciation) extracted from the automatic speech recognition (ASR) system capture the speakers' articulatory characteristics. The linguistic complexity features (syntactic complexity, lexical richness, register-based n-gram frequency, and information-theoretic measures) were generated by analyzing the transcriptions using the Complexity Contour Generator (CoCoGen) [12]. In [17], the BERT [13] and ERNIE [14] models were fine-tuned to capture the language characteristics of the speakers in ADReSSo 2021 challenge [18]. Syed *et al.* [15] compared the efficacy of BERT and its derivatives, including DistilBERT [19] and RoBERTa [20], for capturing the structural and linguistic properties of the transcriptions.

Several studies have investigated the relevance of various features for dementia detection. For example, Weiner *et al.* [21] extracted features from biographic interviews to predict the development of AD after 5 years. They reduced the dimensions of the original feature set by nested forward feature selection (FS) and found that FS can significantly improve prediction performance. Ammar *et al.* [22] employed information gain FS, KNN model-based FS, and SVM recursive feature elimination (SVM-RFE) to select informative linguistic features. Their results demonstrate the effectiveness of FS in improving the accuracy of AD detection. Alhanai *et al.* [23] extracted demographic, audio, and text features and used a binomial logistic-regression model regularized by an elastic-net to identify the discriminative features for cognitive impairment recognition. Their method ranks features according to the sparsity regularization coefficients of the regression model.

1.2. Innovations and Contributions

While various types of features have been used for dementia detection, it is still unclear which features or combination of features are more effective. Therefore, this study investigates FS methods to identify the most discriminative features (the spoken language biomarkers) for screening dementia. We propose a two-step FS approach in which the first step utilizes filter methods to pre-screen the features, and the second step uses a novel feature ranking (FR) method to rank and select the screened features through the dropout masks of a deep neural network. We regard the features selected in the second step as the spoken language biomarkers. The proposed FR utilizes a dual-net architecture, where two networks (called operator and selector) are alternatively and cooperatively trained to simultaneously perform FS and dementia detection. In particular, the selector has dropout masks in its input layer for which the trainable dropout rates are inversely proportional to the features' importance. We refer to the FR method as dual dropout ranking (DDR). DDR was comprehensively evaluated on a synthetic dataset, the MNIST hand-written digit dataset, and two dementia-related datasets. We then proposed a two-step FS approach to address the difficulty in selecting features from extremely high-dimensional vectors under limited training data scenarios. Our method shows significant performance improvement compared to several state-of-the-art deep-learning-based FS methods.

In [24], we demonstrated that a dual-net architecture with trainable dropout rates can discard 74% of the original features, leading to a substantial performance gain. In this paper, we further demonstrate the effectiveness of the dual-net architecture for FS through extensive experiments on two public-domain AD datasets: ADReSS [25] and AD2021 [26]. Our study is limited to FS as we aim to identify specific spoken language biomarkers of the disease. Our ultimate research goal is to comprehensively elucidate the spoken language biomarkers concerning both data analytics and biological aspects of brain functions. By identifying interpretable biomarkers, we can facilitate disease diagnosis and monitor disease progression.

We summarize our contributions as follows. 1) This study is the first to exploit deep-learning-based FS methods to identify spoken language biomarkers for dementia detection under limited training data scenarios. 2) We extend the trainable dropout rates to a dual-net architecture and propose a novel feature ranking method (DDR). 3) We propose a two-step FS approach to address the difficulty in selecting features from extremely high-dimensional

vectors. 4) We identify the spoken language biomarkers that can boost the performance of dementia detection on two public-domain datasets.

The remainder of this paper is organized as follows. Section 2 presents the technical details of DDR. Section 3 introduces diverse features for dementia detection. Section 4 details the two-step FS approach to selecting spoken language biomarkers. Section 5 describes experimental setup, after which Section 6 demonstrates the effectiveness of the two-step FS approach and then applies the two-step FS approach for dementia detection. Discussions and concluding remarks are given in Section 7.

2. Dual Dropout Ranking

2.1. Dropout for Feature Ranking

FR aims to rank the importance of individual features according to some criteria, where the criteria typically reflect the features' contributions to the learning performance [27].

In dropout [28], nodes are purged according to their dropout rates. Therefore, the *higher* the dropout rate, the *lower* the importance of the feature, and FR amounts to determining the dropout rates of individual input nodes. To formulate the dropout rate of a feature, we adopt an approach similar to dropout feature ranking (DropoutFR) [29]. Specifically, given a dropout rate vector $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k, \ldots, \theta_d)$ and a dropout mask vector $\boldsymbol{z} = (z_1, z_2, \ldots, z_k, \ldots, z_d)$, we denote the distribution of $\boldsymbol{z}$ as $q(\boldsymbol{z}) = \prod_{k=1}^{d} q\left(z_k \mid \theta_k\right) = \prod_{k=1}^{d} \mathrm{Bern}\left(z_k \mid \theta_k\right)$, where θ_k is the dropout rate for the k^{th} feature and $z_k \in \{0, 1\}$ is the corresponding dropout mask. This gives us a fully factorized Bernoulli distribution that focuses on FR. Suppose $\boldsymbol{x} = (x_1, x_2, \ldots, x_k, \ldots, x_d)$ is an input feature vector. During the forward pass, we place the dropout mask vector on the input layer, that is $\boldsymbol{x} \odot \boldsymbol{z}$, where $\odot$ is the element-wise product (Hadamard product).

2.2. Trainable Dropout Rates

In ordinary dropout, the dropout rates are fixed hyper-parameters. Instead of fixing the dropout rates, we treat them as *trainable* parameters. To optimize the dropout rates, we relax the binary dropout masks to *soft* dropout masks as follows:

$$\boldsymbol{z}(\boldsymbol{\theta}) = \mathrm{sigmoid}\left(\frac{1}{t}\left[\log \boldsymbol{\theta} - \log(\mathbf{1} - \boldsymbol{\theta}) + \log \boldsymbol{u} - \log(\mathbf{1} - \boldsymbol{u})\right]\right), \qquad (1)$$

where $\boldsymbol{u} \in \mathbb{R}^d$ follows the Uniform$(\mathbf{0}, \mathbf{1})$ distribution and t is a normalization constant, which is set to 0.1 in our experiments. Note that this relaxation has also been used in Concrete Dropout [30] and DropoutFR [29]. Eq. (1) suggests that $q(\boldsymbol{z})$ places most of the mass to either $z_k = 0$ or $z_k = 1$ to closely resemble the binary dropout mask. With the continuous relaxation in Eq. (1), the dropout rates can be optimized through backpropagation, and we can gradually select the optimal features $\boldsymbol{x} \odot \boldsymbol{z}$ along with the optimization of the dropout rates. The relation between the features' ranks and trainable dropout rates is depicted in Fig. 1.

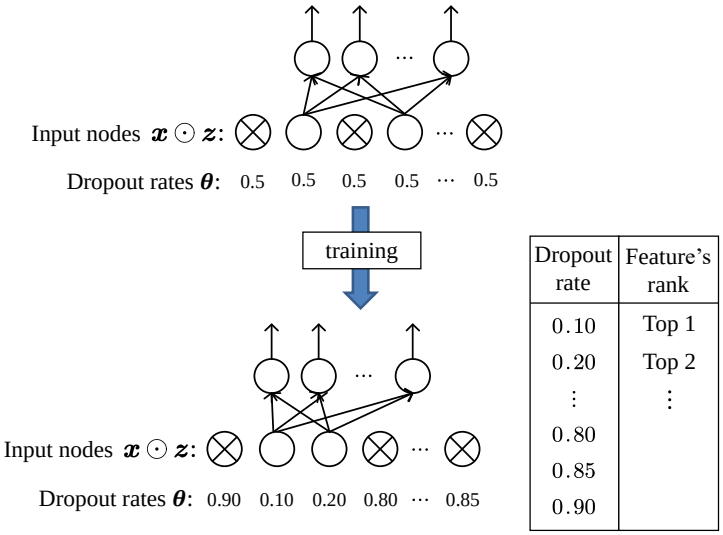

Figure 1: The relationship between the features' ranks and the trainable dropout rates. Before training, each of the input features is assigned the same dropout rate (e.g., 0.5). After training, the features with a lower dropout rate will be assigned a higher rank.

2.3. Learning Algorithm

Suppose $\mathcal{M} = \{\mathcal{X}, \mathcal{Y}\}$ is a mini-batch comprising $|\mathcal{M}|$ pairs of $\boldsymbol{x}$ and $\boldsymbol{y}$, where $\boldsymbol{x} \in \mathcal{X}$ is a feature vector of size d, and $\boldsymbol{y} \in \mathcal{Y}$ is the corresponding target. By sampling the uniform distribution in Eq. (1), we obtain several *soft* dropout mask vectors $\boldsymbol{z} = (z_1, z_2, \ldots, z_k, \ldots, z_d)$ and form a dropout mask subset $\mathcal{Z}$ of size $|\mathcal{Z}|$. The learning objectives of the dual-net for DDR

are defined as:

Operator's objective:[2]

$$\mathcal{L}_O\left(\mathcal{M}, \mathcal{Z}; \psi\right) = \frac{1}{|\mathcal{Z}||\mathcal{M}|} \sum_{\boldsymbol{z}\in\mathcal{Z}} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{M}} l(\boldsymbol{x}\odot\boldsymbol{z}, \boldsymbol{y}; \psi), \tag{2a}$$

Selector's objective:[3]

$$\mathcal{L}_S\left(\mathcal{Z}(\boldsymbol{\theta}); \varphi\right) = \frac{1}{|\mathcal{Z}|} \sum_{\boldsymbol{z}\in\mathcal{Z}} \left\{ \left| f_S(\boldsymbol{z}; \varphi) - \frac{1}{|\mathcal{M}|} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{M}} l(\boldsymbol{x}\odot\boldsymbol{z}, \boldsymbol{y}; \psi) \right| \Big/ \sum_{k=1}^{d} (1 - z_k) \right\}, \tag{2b}$$

where $l(\boldsymbol{x}\odot\boldsymbol{z}, \boldsymbol{y}; \psi)$ is either the cross-entropy loss for binary/multiclass classification or the mean squared error (MSE) loss for regression, ψ is the operator's parameters, $f_S(\boldsymbol{z}, \varphi)$ is the selector's output, and φ is the selector's parameters. The relationship between the operator and the selector in the dual-net architecture is depicted in Fig. 2. During training, the operator and selector are trained alternately. The alternate training procedure is depicted in Appendix A. The advantages of the dual-net architecture are as follows. 1) It can offload the optimization of dropout rates to the selector, which lets the operator focus on the classification or regression tasks. 2) It can shift the FS constraint (the denominator of Eq. (2b)) to the selector, and with the alternate training procedure, it enables *automatic regularization*. 3) It avoids manually setting the regularization coefficients.

2.3.1. Operator

The operator is trained on the features selected by the selector to reduce classification loss. For each iteration, given the dropout mask subset $\mathcal{Z}$ from the selector, the selected features $\{\boldsymbol{x}\odot\boldsymbol{z}\}_{\boldsymbol{x}\in\mathcal{X},\boldsymbol{z}\in\mathcal{Z}}$ are fed to the operator, and the operator's learning performance based on the selected features is obtained. Given the selected features $\boldsymbol{x}\odot\boldsymbol{z}$, $\frac{1}{|\mathcal{M}|}\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{M}} l(\boldsymbol{x}\odot\boldsymbol{z}, \boldsymbol{y}; \psi)$ is the learning performance of the operator on the mini-batch $\mathcal{M}$. By enumerating $\boldsymbol{z}$ in $\mathcal{Z}$, we obtain the average learning performance of the operator on the

[2]During the optimization of the operator, $\boldsymbol{\theta}$ is considered fixed. Therefore, we drop the dependence of $\boldsymbol{z}$ on $\boldsymbol{\theta}$.

[3]For notational simplicity, we omit the dependence of $\boldsymbol{z}$ on $\boldsymbol{\theta}$ on the right side of this equation.

mini-batch. Then, we update the operator's parameters and pass the operator's learning performance to the selector as feedback indicating how well the operator performs on the selected features. Different from the sparsity regularization methods that also incorporate regularization into the network, the operator only focuses on reducing classification loss. Given the selected features, the operator's architecture can be tailored to different learning tasks (classification or regression).

2.3.2. Selector

The selector learns to predict the operator's learning performance using as few selected features as possible. The mean absolute error (MAE) between $f_S(\boldsymbol{z}, \varphi)$ and $\frac{1}{|\mathcal{M}|}\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{M}} l(\boldsymbol{x}\odot\boldsymbol{z}, \boldsymbol{y}; \psi)$ requires that the selector closely predicts the operator's learning performance. The constraint $\sum_{k=1}^{d}(1-z_k)$ on the denominator of Eq. (2b) automatically causes most of the dropout masks in $\boldsymbol{z}$ to become 0; so the selector only selects a small number of features when predicting the operator's learning performance.

After training and updating the selector's parameters and dropout rates, we have the updated dropout rate vector $\boldsymbol{\theta}'$. Through sampling the uniform distribution in Eq. (1), we obtain several new soft dropout mask vectors $\boldsymbol{z}'$ from the updated dropout rate vector $\boldsymbol{\theta}'$ and form a new dropout mask subset $\mathcal{Z}'$ for the next iteration. In practical implementation, the dropout mask vector fed to the selector is $\boldsymbol{z}\odot\boldsymbol{z}'$, where $\boldsymbol{z}\in\mathcal{Z}$ and $\boldsymbol{z}'\in\mathcal{Z}'$.

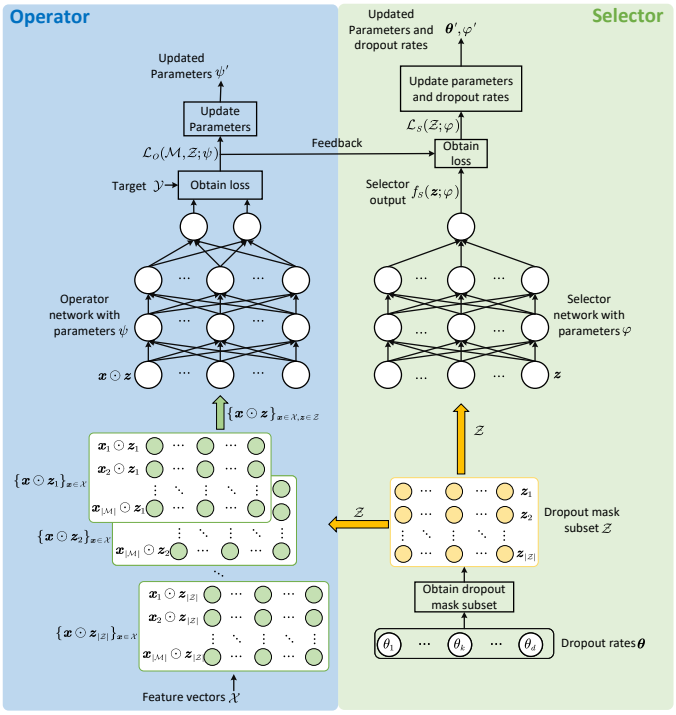

Figure 2: The dual-net architecture of DDR. ψ and φ represent the network parameters of the operator and selector, respectively. $\boldsymbol{\theta}$ comprises the dropout rates at the input layer of the selector. $\mathcal{X}$ contains $|\mathcal{M}|$ feature vectors and $\mathcal{Z}$ contains $|\mathcal{Z}|$ dropout masks.

3. Feature Engineering

We focus on two categories of features: transcription-based and speech-based. The transcription-based features are extracted from either the manual or automatic transcriptions, which capture the semantic, syntactic, and lexical aspects of the speaker's spoken language. The speech-based features contain a variety of acoustic characteristics of the speakers. For the ADReSS dataset [25], the transcription-based features significantly outperform the speech-based features [16, 25] because accurate manual transcriptions are provided; therefore, we focus on the transcription-based features only. For the AD2021 dataset [26], because of erroneous transcriptions, we include various types of speech-based features in addition to the transcription-based features.

3.1. ADReSS Dataset

3.1.1. Linguistic Features

34 linguistic features were extracted from the CHAT annotated transcriptions using the EVAL command in the CLAN program [25]. The features include lengths of utterances, type-token ratios, statistics of part-of-speech (POS), etc. The ADReSS challenge [25] has provided a baseline recognition performance on the linguistic features.

3.1.2. BERT Features

The BERT model [13], which comprises deep bidirectional transformers, has been widely adopted in natural language processing (NLP). A pre-trained BERT model can be fine-tuned to suit a wide range of tasks. In [31], the authors fine-tuned a BERT model at the transcription level for AD recognition and achieved impressive results. In this paper, we use the pre-trained BERT model as a *feature extractor*. More specifically, we fed the subjects' transcriptions to the pre-trained BERT model and extracted the representations from the last layer of the model. For each subject, the model produces a 768-dimensional feature vector (called the BERT features) that abstractly captures the semantic, syntactic, and lexical information of the transcriptions. Li *et al.* [16] extracted BERT features from both manual and automatic transcriptions. Their results demonstrate the effectiveness of the BERT features for dementia detection.

3.1.3. Pause Features

In [32], the authors demonstrated that pauses can function as word-finding, as planning at the word, phrase, and narrative levels, and as pragmatic compensation when other interactional and narrative skills deteriorate. In [31], pause information was incorporated into the feature representations to improve AD recognition performance. Thus, we included the pause features for dementia detection.

We used the pause statistics in Table 1 as the pause features. To obtain these features, we followed the procedure in [31] and used the 'chat2text' command in CLAN to convert the CHAT annotated transcriptions into plain words and tokens. Then, the converted transcriptions were forced aligned with the speech recordings using the Penn Phonetics Lab Forced Aligner [33]. The outputs of the alignments contain the identifications and durations

of the pauses.[4]

We divided the pauses into six duration groups: G_1 (pauses between 0.05s–0.5s), G_2 (pauses between 0.5s–1s), G_3 (pauses between 1s–2s), G_4 (pauses between 2s–3s), G_5 (pauses between 3s–4s), and G_6 (pauses longer than 4s). For each duration group, we extracted the five pause features in Table 1. As a result, we had a total of $5 \times 6 = 30$ pause features per recording.

Table 1: Five pause features extracted from the six duration groups (listed in Section 3.1.3).

Pause feature	**Description**
#p	Number of pauses per minute
%p/word ratio	Pause-to-word ratio
p duration	Total duration of pauses per minute
p mean duration	Mean duration of pauses
%p duration/word duration	Pause-duration-to-word-duration ratio

3.2. AD2021 Dataset

3.2.1. Lexical Features

Because manual transcriptions are not available in AD2021, ASR was used before extracting the lexical features. The Tencent Cloud ASR[5] was adopted to transcribe the Mandarin speech recordings. Based on the transcriptions, the following lexical features were extracted:[6] the number of sentences per minute, the average number of words per sentence, the ratio of unique words to all words, and the average word frequency. Then, the Stanford POS tagger[7] was utilized to parse the transcriptions to extract the following lexical features: POS counts per minute, POS ratio, the ratio of pronoun to noun, the ratio of noun to verb, the maximum parsed tree height, the mean parsed tree height, and the median parsed tree height. These lexical features lead to a 143-dimensional feature vector per recording.

[4]The between-word pauses are indicated by 'sp'.

[5]`https://cloud.tencent.com/product/asr`

[6]The lexical features are extracted using this toolbox: `https://github.com/SPOClab-ca/COVFEFE`.

[7]`https://nlp.stanford.edu/software/tagger.shtml`

3.2.2. BERT Features

Similar to the ADReSS dataset, a pre-trained Chinese BERT model[8] was employed as the feature extractor. The transcriptions were fed to the BERT model and high-level representations were extracted from the last layer of the model, resulting in a 768-dimensional vector per recording.

3.2.3. Acoustic Features

We followed the standard pipelines in the COVFEFE toolbox[9] to extract the acoustic features from the speech recordings, which include formants, loudness, pitch, zero-crossing rate, etc.

3.2.4. COVAREP Features

COVAREP [34] provides comprehensive acoustic features, which include prosodic features (fundamental frequency and voicing), voice quality features, and spectral features. We extracted COVAREP features at 100Hz; for each recording, the mean, maximum, minimum, median, standard deviation, skew, and kurtosis of the features were computed, leading to a 518-dimensional feature vector per recording.

3.2.5. INTERSPEECH 2010 Paralinguistic Challenge Features (IS10)

IS10 [35] is a feature set for emotion recognition and bipolar disorder recognition. In addition to the 32 low-level descriptors (LLDs) in INTERSPEECH 2009 Emotion Challenge (IS09), 44 LLDs were added to IS10, including PCM loudness, eight log Mel-frequency bands, eight line-spectral frequency pairs, F0 envelope, voicing probability, jitter, and shimmer. Twelve statistics (minimum, maximum, mean, range, etc.) of the LLDs were computed, leading to a 1582-dimensional feature vector per recording.

3.2.6. Pause Features

An energy-based voice activity detector (VAD) was utilized to identify the pauses. Similar to the ADReSS dataset, the pauses were divided into six groups, and pause features (Table 1) were determined from individual groups, leading to a 30-dimensional feature vector per recording.

[8] `https://huggingface.co/bert-base-chinese`

[9] `https://github.com/SPOClab-ca/COVFEFE`

4. Feature Selection

4.1. Two-step Feature Selection

The features described in Section 3 amount to 832-dimensional vectors for the ADReSS dataset and 3071-dimensional vectors for the AD2021 dataset. The feature dimensions are much larger than the number of training samples, which will easily cause overfitting in machine learning models. Because of the limited number of training samples, the high-dimensional feature vectors also cause difficulties for FS. In this section, we extend the DDR in Section 2 to a *two-step FS* approach, which aims to deal with the circumstance where the feature dimensions are much larger than the number of training samples. The FS approach is depicted in Fig. 3. FS can be nested inside cross-validation (CV), which means that FS is conducted on the training partitions (TR) of individual folds instead of the entire training set. On the TR of individual folds, a two-step FS approach is applied to select the most discriminative features, as shown in Fig. 3. Filter methods are usually computationally cheap and do not require training. When the feature dimension is very high, filter methods are indispensable for obtaining a reduced set of features for the expensive FS methods. Therefore, in Step 1, filter methods are utilized to pre-screen the original features. Three filter methods were evaluated in the experiments: Fisher's discriminant ratio (FDR) [36], PeaCorr tests, and mutual information (MutInfo). In Step 2, the proposed DDR is applied to rank the remaining features. Before training, each of the remaining features is assigned the same dropout rate (e.g., 0.5). During training, the DDR adjusts the dropout rates to reflect the features' importance. After training, we rank the features according to the dropout rates and select the features with low dropout rates.

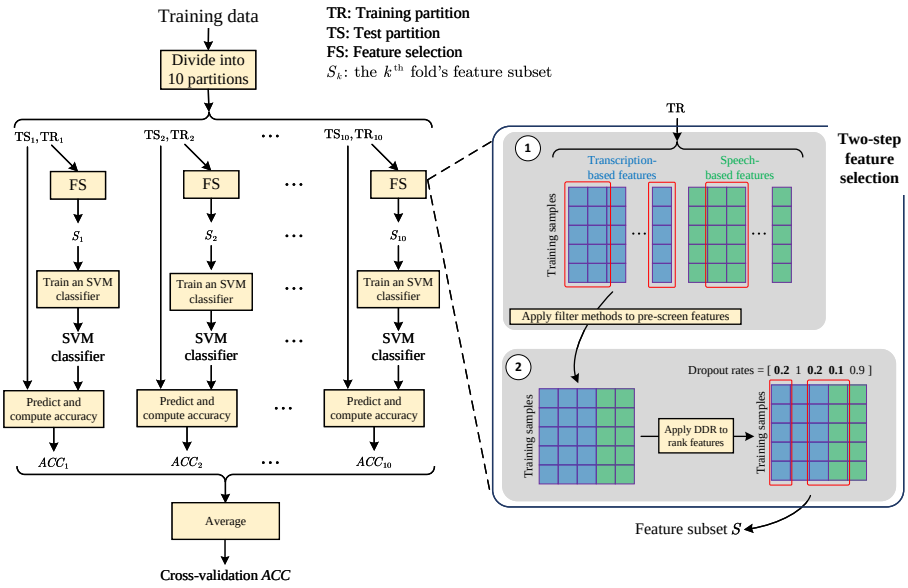

Figure 3: The FS procedure: 10-fold CV was adopted. The training data were divided into 10 TR. In the TR of individual folds, the two-step FS approach was applied to select the most discriminative features. In Step 1, filter methods were utilized to pre-screen the features. In Step 2, DDR was adopted to rank the features selected in Step 1 (the remaining features). Features with low dropout rates were then selected.

4.2. Feature Selection in Cross-validation

As illustrated in Fig. 3, an approach termed *nested* FS is adopted within the 10-fold CV as opposed to the traditional approach to conducting FS outside the CV. Specifically, within each fold of the CV, the TR is used to select features, which are then tested using the testing partition (TS). As the training data differ among the individual folds, different features will be selected in each fold. Conducting FS on the entire training data prior to CV introduces bias, as the TS will also be used for FS. This may ultimately affect the CV results.

5. Experimental Setup

5.1. Datasets

5.1.1. ADReSS

The AD Recognition Through Spontaneous Speech Challenge (ADReSS) [25] provides a benchmark dataset and a platform where the research community can compare their methods for improving AD detection performance.

The dataset comprises recordings of the spoken-language descriptions of the Cookie Theft picture description task in Boston Diagnostic Aphasia Examinations. 156 subjects aged between 50 to 80 participated in the examinations, among whom 78 were AD patients and 78 were healthy controls (HC). Among these participants, 108 were grouped into the training set, and the remaining 48 were grouped into the test set. The dataset is gender-balanced, and the spoken language is English. Table 2 shows the dataset's details.

5.1.2. AD2021

The AD2021 dataset [26] was released through an AD recognition competition organized by Jiangsu Normal University, SATLab of Tsinghua University, and Beijing Haitian Ruisheng Science Technology Ltd. The dataset comprises the speech recordings of "Cookie Theft picture description" sessions, fluency tests, and normal conversations. The training set contains 25 AD patients, 53 older adults suffering from mild cognitive impairment (MCI), and 44 HC. Each subject in the training set has several recording sessions, resulting in 279 training sessions. The test set contains 119 subjects, of which 35 are AD patients, 39 have MCI, and 45 are HC. The spoken language of the dataset is Mandarin Chinese. No manual transcription is provided. Table 2 shows the dataset's details.

Table 2: The characteristics of the ADReSS and AD2021 datasets. *AD*: Alzheimer's Disease, *MCI*: mild cognitive impairment, *HC*: healthy control, *M*: male, and *F*: female.

Dataset		**ADReSS**								**AD2021**											
Training/test data		Training data				Test data				Training data						Test data					
Class		HC		AD		HC		AD		HC		MCI		AD		HC		MCI		AD	
Gender		M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F	M	F
Age	[50, 55)	1	0	1	0	1	0	1	0	18	26	27	27	10	15	22	23	10	29	6	29
	[55, 60)	5	4	5	4	2	2	2	2												
	[60, 65)	3	6	3	6	1	3	1	3												
	[65, 70)	6	10	6	10	3	4	3	4												
	[70, 75)	6	8	6	8	3	3	3	3												
	[75, 80)	3	2	3	2	1	1	1	1												
Number of samples		54		54		24		24		108		93		78		45		39		35	
Spoken language		English								Mandarin Chinese											
Task		Cookie theft picture description								Cookie theft picture description, fluency test, and normal conversation*											
Manual transcriptions provided		Yes								No											

*Our experiments adhered to the official guideline by utilizing all the three tasks.

5.2. Implementation Details of DDR

Both the operator network and selector network in DDR are feedforward neural networks. A batch-normalization layer followed by a dropout layer

with a dropout rate of 0.5 was added after each hidden layer in the two networks. The activation function for the hidden layers is ReLU for both networks, while the activation function for the last layer of the operator network is softmax, and that for the selector network is linear. An Adam optimizer with a learning rate of 0.001 was used to optimize the networks' parameters and the trainable dropout rates, which were initialized to 0.35. The batch size $|\mathcal{M}|$ was set to 32 and the size of the dropout mask subset $|\mathcal{Z}|$ was set to 32. On a Ubuntu 20.4 machine with one RTX3090 GPU, each experiment took about 5 minutes.

5.3. Performance Metrics

The goal is to determine the most discriminative features that can effectively identify individuals who are HC, those with MCI, and others with AD. The two-step FS approach described in Section 4 was utilized to identify the discriminative features. For the ADReSS dataset, the identified features were then used for training linear SVM classifiers[10] with a box constraint of 1 to classify AD and HC. For the AD2021 dataset, the selected features were used for training Gaussian SVM classifiers[11] with a box constraint of 1 to identify AD, MCI, and HC.

The performance metrics for the ADReSS dataset include precision (PRE), recall (REC), and F_1 scores for each class (AD and HC) as well as their unweighted mean and accuracy (ACC). The performance on the training set was obtained by 10-fold CV.

For the AD2021 dataset, except for the ACC, the performance metrics were calculated for each class (AD, MCI, and HC), and their unweighted mean was reported. The 10-fold CV was replaced by a leave-n-subject-out CV in which the training samples of the same speakers were grouped into either the TR or the TS for each fold.

6. Experiments and Results

In this section, we first evaluate DDR on a synthetic dataset and the MNIST hand-written digit dataset and then evaluate the two-step FS approach on the ADReSS and AD2021 datasets.

[10]The classifier's setting was adopted from [16]. The SVMs were forced to produce probabilistic outputs when computing the predicted scores.

[11]The classifier's setting was taken from the AD2021 competition baseline.

6.1. Analysis of Keep Probabilities on a Synthetic Dataset

A synthetic data set was designed to evaluate the capability of classifiers and FS algorithms in solving a multi-dimensional XOR problem [37]. By grouping the eight corners of a 3-dimensional hypercube $(v_0, v_1, v_2) \in \{-1, 1\}^3$ into the tuples (v_0v_2, v_1v_2), we have 4 sets of vectors and their negations $\{\boldsymbol{v}^{(c)}, -\boldsymbol{v}^{(c)}\}_{c=1}^{4}$, where c is the class index. For example, the tuple $(v_0v_2, v_1v_2) = (-1, -1)$ corresponds to $c = 2$, where $\boldsymbol{v}^{(2)} = [1, 1, -1]^\intercal$. The points in class c are generated from the distribution $\frac{1}{2}[\mathcal{N}(\boldsymbol{v}^{(c)}, 0.5\boldsymbol{I}_3) + \mathcal{N}(-\boldsymbol{v}^{(c)}, 0.5\boldsymbol{I}_3)]$, where $\boldsymbol{I}_3$ is a 3×3 identity matrix and $\mathcal{N}(\mu, \sigma)$ is a Gaussian distribution. Each sample is additionally accompanied by 7 Gaussian noise features with zero mean and unit variance, leading to a 10-dimensional feature vector.

We trained a dual network (Fig. 2) on the synthetic data for feature ranking. After training, the keep probabilities $(\mathbf{1} - \boldsymbol{\theta})$ of the features for 20 random seeds are depicted in Fig. 4. It shows that the keep probabilities associated with the valid features (v_0, v_1, v_2) converge to 1, whereas the noise features $(v_3 \sim v_9)$ have keep probabilities close to 0. This result suggests that DDR can effectively identify the valid features.

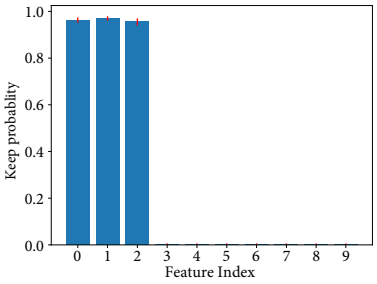

Figure 4: The keep probabilities $(\mathbf{1} - \boldsymbol{\theta})$ of 10 features in the synthetic dataset for 20 random seeds. Indexes 0–2 and 3–9 correspond to the valid and invalid features, respectively. The blue bars and the red error bars denote the means and two times the standard deviations of 20 random seeds, respectively.

6.2. Visualizing the Keep Probabilities

To further demonstrate the explainability of DDR, we employed digits '3' and '8' in the MINST hand-written dataset to train a Gaussian SVM classifier. For the dual networks, we adopted the architectures "784–128–32–2" for the operator net and "784–128–32–1" for the selector net. These

architectures mean that we flattened each image's 28×28 pixels into a 784-dimensional vector as the input to these networks. We utilized an Adam optimizer with a 0.001 learning rate and initialized the trainable dropout rates to 0.35. The batch size $|\mathcal{M}|$ and the size of the dropout mask subset $|\mathcal{Z}|$ were set to 32. We trained the network with 25,000 epochs using a 5-fold CV. We trained a dual-net for each fold and selected 50 features according to the selector net. The selected features were then used for training a Gaussian SVM with a box constraint of 1.0 to classify digits '3' and '8'.[12] We achieved an accuracy of 0.981 ± 0.003 based on the selected features. The feature importance map is shown in Fig. 5. It shows that DDR can identify the relevant features despite the flattening process destroying the images' spatial information.

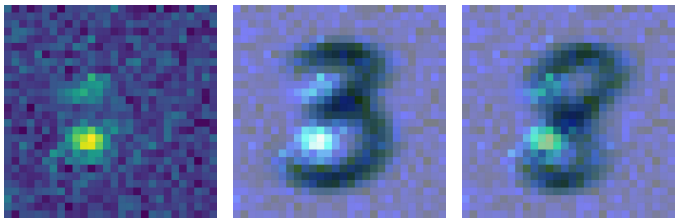

Figure 5: A feature importance map produced by a selector trained on MNIST data. The left picture is the normalized feature importance map. The middle and the right pictures are the feature importance map superimposed on the mean images of digit '3' and digit '8', respectively.

6.3. Performance of Different Feature Types

We first evaluated the recognition performance of all the feature sets *before* FS. We ran 100 repetitions of 10-fold CV and averaged the performance values. The corresponding results are reported in Table 3. The results show that on the ADReSS training data, the linguistic features achieve the best performance before FS. On the AD2021 training data, the IS10 feature set achieves the best performance among all the feature sets. The transcription-based features (lexical and BERT) perform worse than the speech-based features. This may be due to word errors in the automatic transcriptions.

[12]The source codes are available at `https://github.com/kexquan/dual-dropout-ranking`.

Table 3: Classification performance on the ADReSS and the AD2021 training data before FS. The numbers in the brackets are the sizes of the feature sets.

Dataset	**Feature set**	**10-fold CV on training data**			
		ACC	PRE	REC	F_1
ADReSS	Linguistic (34)	***0.802***	***0.806***	***0.799***	***0.783***
	BERT (768)	0.748	0.737	0.776	0.735
	Pause (30)	0.523	0.534	0.446	0.454
AD2021	Lexical (143)	0.553	0.479	0.511	0.450
	BERT (768)	0.575	0.514	0.530	0.482
	Acoustic (30)	0.613	0.575	0.565	0.519
	COVAREP (518)	***0.678***	0.636	0.628	0.578
	IS10 (1582)	0.666	***0.638***	***0.642***	***0.587***
	Pause (30)	0.351	0.308	0.324	0.281

6.4. Performance of Filter Methods

For the ADReSS dataset, we combined all the features to form 832-dimensional vectors. The dimensionality of the combined features for the AD2021 training data is 3071. When conducting 10-fold CV on the combined features, large differences in recognition performance across CV were observed, as illustrated in Fig. 6. This is because during the CV, applying random splitting on a limited number of training samples will induce great differences across TR in different folds. These large differences suggest recognition performance on unseen data is likely to be brittle. To mitigate this brittleness, we propose the following ensemble procedure to stabilize the classification performance during CV. We ran I repetitions of CV based on different data splittings. We then produced the predicted scores $p(i, j)$ for subject j in CV i. Finally, we averaged the predicted scores $p(j) = (1/I) \sum_{i=1}^{I} p\,(i, j)$ over all the CV for each of the J subjects, as shown in Fig. 7.

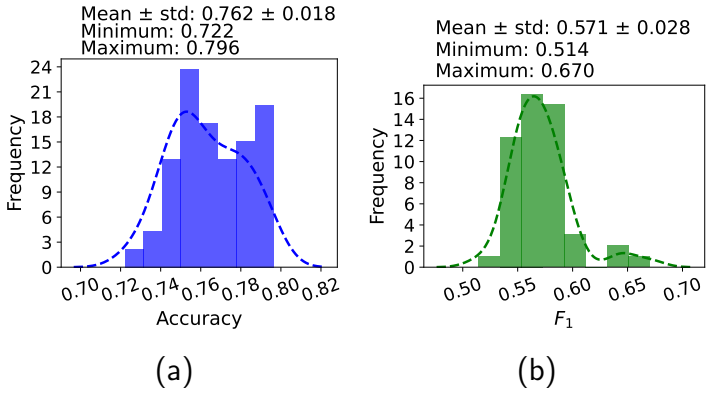

Figure 6: When conducting 10-fold CV based on different data splittings, large variations in recognition performance across CV were observed on (a) the ADReSS and (b) the AD2021 training data.

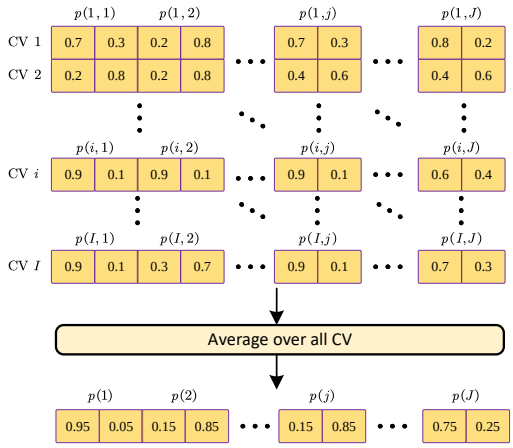

Figure 7: The ensemble procedure to stabilize the classification performance during CV. We ran I repetitions of CV based on different data splittings and averaged the predicted scores $p(i, j)$ over all the CV for each of the J subjects.

To test our proposed ensemble procedure, we ran 50 repetitions of CV based on different data splittings. From the 50 CV, we selected five CV ($m = 5$) and averaged the predicted scores over the five CV. The results in the first row of Table 4 summarize 100 draws of the five CV. The second row is similar, except $m = 10$. Comparing the legend of Fig. 6 and Table 4, we can see that the ensemble procedure increases the mean ACC and F_1 and reduces

variances on both datasets. On the ADReSS training data, when $m = 25$, the ensemble procedure achieves the highest mean ACC and boosts the minimum ACC from 0.722 (Fig. 6(a)) to 0.750. On the AD2021 training data, the ensemble procedure achieves the highest mean F_1 and boosts the minimum F_1 from 0.514 (Fig. 6(b)) to 0.555 when $m = 10$. Therefore, subsequent experiments repeated the CV 25 times and averaged the predicted scores on the ADReSS dataset. On the AD2021 dataset, we conducted 10 repetitions of CV and averaged the predicted scores.

Table 4: The proposed ensemble procedure improves mean classification performance and reduce variances. m is the ensemble size.

	ADReSS (ACC)		AD2021 (F_1)	
m	Mean ± std	Min – Max	Mean ± std	Min – Max
5	0.768 ± 0.013	0.741 – 0.796	0.571 ± 0.010	0.552 – 0.603
10	0.769 ± 0.010	0.741 – 0.787	***0.573 ± 0.007***	0.555 – 0.588
15	0.771 ± 0.009	0.750 – 0.787	0.570 ± 0.007	0.552 – 0.588
20	0.773 ± 0.009	0.750 – 0.787	0.571 ± 0.006	0.561 – 0.582
25	***0.773 ± 0.008***	0.750 – 0.787	0.570 ± 0.005	0.555 – 0.582
30	0.771 ± 0.008	0.759 – 0.787	0.571 ± 0.006	0.555 – 0.585
35	0.771 ± 0.008	0.759 – 0.787	0.572 ± 0.005	0.561 – 0.585
40	0.771 ± 0.008	0.759 – 0.787	0.571 ± 0.005	0.558 – 0.582
45	0.771 ± 0.006	0.759 – 0.787	0.571 ± 0.004	0.564 – 0.582

We followed the procedure described in Section 4 to evaluate the classification performance of the filter methods (FDR, PeaCorr, and MutInfo) on the combined feature vectors. Note that FS was performed *inside* the CV, and each fold may select different features because the TR in Fig. 3 were different for different folds. On the TR of individual folds, we applied the filter methods to reduce the feature dimension to $n = \{25, 50, 100, 150, \ldots, 600\}$, as shown in Table 5. It shows that using the filter methods to pre-screen the combined features can improve classification performance on both datasets. On the ADReSS training data, MutInfo achieves the highest ACC (0.796) when the feature dimension was reduced to 50. On the AD2021 training data, MutInfo achieves the highest F_1 (0.641) when the feature dimension was reduced to 100. Therefore, in the two-step FS, subsequent experiments utilized MutInfo to pre-screen the combined feature vectors to 50 and 100 for ADReSS and AD2021, respectively.

Table 5: Classification performance of the filter methods on the ADReSS and the AD2021 training data. n: the number of selected features.

	ADReSS (ACC)			AD2021 (F_1)		
n	FDR	PeaCorr	MutInfo	FDR	PeaCorr	MutInfo
25	0.741	0.741	0.778	0.559	0.596	0.623
50	0.759	0.769	***0.796***	0.567	0.592	0.640
100	0.769	0.769	0.759	0.586	0.588	***0.641***
150	0.731	0.741	0.787	0.568	0.588	0.623
200	0.741	0.741	0.787	0.568	0.604	0.601
250	0.778	0.778	0.778	0.585	0.601	0.597
300	0.778	0.778	0.787	0.586	0.592	0.602
350	0.787	0.787	0.778	0.582	0.595	0.594
400	0.778	0.778	0.787	0.594	0.595	0.583
450	0.778	0.778	0.787	0.591	0.595	0.586
500	0.787	0.787	0.769	0.585	0.592	0.585
550	0.787	0.787	0.769	0.582	0.598	0.588
600	0.796	0.796	0.759	0.577	0.588	0.579

6.5. Performance of Two-step FS on Training Data

This subsection reports the performance of DDR and some strong supervised FS methods on the ADReSS and the AD2021 training data. These strong supervised FS methods include deep feature selection (DFS) [38], DropoutFR [29], and feature importance ranking (FIR) [27]. On the TR of individual folds, after using MutInfo to pre-screen the combined features, we applied DDR and these strong supervised FS methods on the remaining 50 features for ADReSS and 100 features for AD2021 to further select relevant features. We adopted the same network architectures ("50–128–32–2" for ADReSS and "100–128–128–32–3" for AD2021) with softmax outputs and default hyper-parameters in the source codes for these strong supervised FS methods and DDR. During the CV, we selected the same number of features n in each fold for each of the FS methods. The results on the ADReSS training data are shown in Table 6, and results on the AD2021 training data are shown in Table 7. The results show that applying DDR and these strong supervised FS methods on the pre-screened features can further improve recognition performance. The two-step FS significantly reduces feature dimensionality while identifying small feature subsets that achieve comparable or superior performance compared with the combined feature sets. The results also show that DDR performs the best on both datasets, that is, it achieves the best mean recognition performance among these FS methods.

Table 6: Recognition performance of the two-step FS on the ADReSS training data. Features were pre-screened by MutInfo. n: the number of selected features in each fold.

	CV on training data (ACC)			
n	DFS [38]	DropoutFR [29]	FIR [27]	DDR (Ours)
5	0.778	0.769	0.778	0.815
10	0.787	0.796	0.778	0.815
15	0.787	0.787	0.778	0.787
20	0.787	0.787	0.769	0.796
25	0.787	0.806	0.769	0.787
Mean	0.785	0.789	0.774	***0.800***

Table 7: Recognition performance of the two-step FS on the AD2021 training data. Features were pre-screened by MutInfo. n: the number of selected features in each fold.

	CV on training data (F_1)			
n	DFS [38]	DropoutFR [29]	FIR [27]	DDR (Ours)
5	0.705	0.663	0.652	0.744
10	0.738	0.714	0.772	0.734
15	0.774	0.736	0.763	0.752
20	0.763	0.744	0.777	0.751
25	0.726	0.760	0.767	0.757
30	0.731	0.760	0.760	0.773
35	0.718	0.729	0.748	0.742
40	0.691	0.701	0.719	0.727
45	0.686	0.692	0.698	0.699
50	0.664	0.679	0.670	0.689
Mean	0.720	0.718	0.733	***0.737***

6.6. Performance of Two-step FS on Test Data

This subsection reports the performance of the identified feature subsets on the ADReSS and AD2021 test data. During CV, each fold may select different feature subsets because the TR in Fig. 3 are different for different folds. When evaluating the selected feature subsets on test data, we utilized the following *soft voting* procedure to incorporate these different feature subsets. We utilized SVM classifiers to produce the predicted scores $p(k)$ for the k-th feature subset. We then averaged the predicted scores $p = (1/K) \sum_{k=1}^{K} p\,(k)$ over all the K feature subsets for the final classification. We computed the results of different sizes of feature subsets and averaged the results in Table 8. We also compared our methods with some recent results in Table 8. On the AD2021 test data, "MutInfo + DDR" achieves the highest recognition performance among all the methods. On the ADReSS test data, the proposed two-step FS significantly performs better than the official baseline. "MutInfo

+ DDR" also outperforms the best reported results in the ADReSS challenge [31]. Additionally, Table 8 supports the following key findings:

1) Our method performs FS on the combined feature vectors (official baseline features [25] + pause features + BERT features [16]). On this basis, our method not only reduces feature dimension but also boosts the accuracy of the official baseline [25] from 75% to 90%.
2) Compared to using the BERT features [16] only, our method can select features that increase the accuracy from 87.5% to 90.4%, while the features selected by"MutInfo + DFS" [38] reduce the accuracy from 87.5% to 86.3%.
3) Our method yields superior performance to "MutInfo + DropoutFR" [29]. Specifically, while the features selected by the latter increase the accuracy from 87.5% to 89.6%, the accuracy achieved by our method is even higher (90.4%).
4) While "MutInfo + FIR" [27] improves the accuracy from 87.5% to 90.0%, it reduces the recall for the AD class from 83.3% to 80.0%. As a result, "MutInfo + FIR" diagnoses fewer AD patients than using the BERT features alone. In contrast, our method not only improves the accuracy to 90.4% but also maintains the recall for the AD class.

Table 8: Recognition performance of the two-step FS on the ADReSS and AD2021 test data. *PRE*: precision; *REC*: recall; *ACC*: accuracy.

Dataset	**Method**	Class/mean	**Performance on test data**			
			PRE	REC	F_1	ACC
ADReSS	Official baseline (Linguistic) [25]	HC	0.700	0.870	0.780	
		AD	0.830	0.620	0.710	0.750
		Mean	0.765	0.745	0.745	
	Pause	HC	0.680	0.708	0.694	
		AD	0.696	0.667	0.681	0.688
		Mean	0.688	0.688	0.687	
	BERT [16]	HC	0.846	0.917	0.880	
		AD	0.909	0.833	0.870	0.875
		Mean	0.878	0.875	0.875	
	Text modality + label fusion [39]	Mean	–	–	–	0.854
	ERNIE3p [31]	Mean	–	–	–	0.896
	BERT + vision transformer [40]	Mean	0.871	0.892	0.880	0.879
	MutInfo + DFS [38]	HC	0.796	0.975	0.876	
		AD	0.968	0.750	0.845	0.863
		Mean	0.882	0.863	0.861	
	MutInfo + DropoutFR [29]	HC	0.852	0.958	0.902	
		AD	0.952	0.833	0.889	0.896
		Mean	0.902	0.896	0.895	
	MutInfo + FIR [27]	HC	0.833	1.000	0.909	
		AD	1.000	0.800	0.889	0.900
		Mean	***0.917***	0.900	0.899	
	MutInfo + DDR (Ours)	HC	0.855	0.975	0.911	
		AD	0.972	0.833	0.897	***0.904***
		Mean	0.913	***0.904***	***0.904***	
AD2021	Official baseline (IS10)[13]	Mean	0.799	0.785	0.786	0.798
	Lexical[14]	Mean	0.738	0.602	0.578	0.630
	Pause	Mean	0.422	0.425	0.421	0.437
	Acoustic[15]	Mean	0.651	0.648	0.647	0.655
	COVAREP [34]	Mean	0.717	0.703	0.704	0.706
	BERT[16]	Mean	0.674	0.620	0.615	0.639
	Wav2vec 2.0 [26]	Mean	0.830	0.828	0.828	0.832
	Adversarial self-supervised model [41]	Mean	0.838	0.837	0.837	–
	MutInfo + DFS [38]	Mean	0.858	0.852	0.851	0.852
	MutInfo + DropoutFR [29]	Mean	0.864	0.861	0.860	0.862
	MutInfo + FIR [27]	Mean	0.862	0.855	0.854	0.857
	MutInfo + DDR (Ours)	Mean	***0.875***	0***.869***	***0.867***	***0.871***

6.7. Analysis of Selected Features

Fig. 8 depicts the t-SNE plots of the ADReSS and AD2021 training data. Fig. 8(b) shows that the selected features distinguish the two groups with a

[16]`https://github.com/THUsatlab/AD2021`

[16]The lexical features are extracted using this toolbox: `https://github.com/SPOClab-ca/COVFEFE`.

[16]The acoustic features are extracted using this toolbox: `https://github.com/SPOClab-ca/COVFEFE`.

[16]`https://huggingface.co/bert-base-chinese`

bigger gap. Fig. 8(d) shows that the selected features reduce the intra-group distance, although there is still some overlap between the groups.

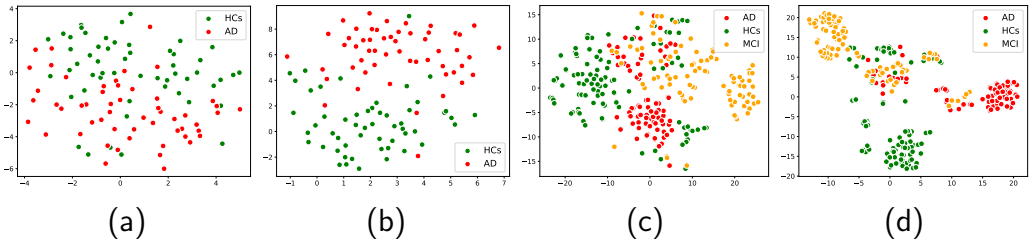

Figure 8: 2D t-SNE plots of the ADReSS training data based on (a) all feature sets and (b) 30 features selected by "MutInfo + DDR" with the highest selection frequency. The selected features distinguish the two groups with a bigger gap. 2D t-SNE plots of the AD2021 training data based on (c) all feature sets and (d) 30 features selected by "MutInfo + DDR" with the highest selection frequency. The selected features reduce the intra-group distance.

We then depict 100 features selected by "MutInfo + DDR" with the highest selection frequency in Fig. 9. Fig. 9(a) shows that although none of the features were selected in all folds, among the 1250 folds, 1083 folds selected the most common feature. The most commonly selected features are BERT features. Additionally, two of the pause features and some of the linguistic features were selected, as shown in Table 9. Fig. 9(b) shows that in the AD2021 dataset, still no feature was selected in all 1000 folds, but the most common one appears in 988 folds. COVAREP and IS10 features were the most commonly selected features. This is reasonable because COVAREP and IS10 features perform well on the training data. Only a few transcription-based features were selected. This may be due to the transcription errors. Compared with the ADReSS dataset, the performance of transcription-based features in AD2021 is unsatisfactory. None of the pause features rank above the top 100.

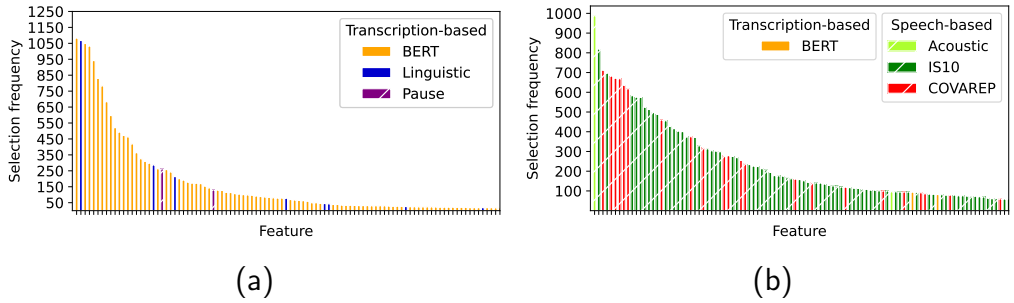

Figure 9: 100 features selected by "MutInfo + DDR" with the highest selection frequency on (a) the ADReSS and (b) the AD2021 training data.

Table 9: The linguistic features and pause features discovered by "MutInfo + DDR" on the ADReSS training data. The parenthesized values are the frequency of the features being selected during the CV. *AD*: Alzheimer's Disease, *HC*: healthy control.

Feature	**Known specificity**
% pro: Percentage of pronouns (1068)	Ahmed *et al.* [42] reported changes in *the number of pronouns*, and Jarrold *et al.* [43] reported an increase in *the proportion of pronouns* in AD patients.
% Nouns: Percentage of nouns (287)	Jarrold *et al.* [43] reported a decrease in *the proportion of nouns* in AD patients.
%p/word ratio: (Pauses between 0.05s–0.5s)-to-word ratio (262)	–
Words/min: Words per minute (214)	AD could be detected through the analysis of voice activity detection and *speech rate* tracking [44].
%p duration/word duration: (pauses between 2s–3s)-duration-to-word-duration ratio (130)	–
noun/verb ratio: Total no. of nouns / total no. of verbs (78)	AD patients may have more difficulty *naming verbs than nouns* [45], and Robinson *et al.* [46] found that AD patients performed worse on a picture-naming task for *verbs than nouns*.

We finally depict the box plots of top 10 selected features in Fig. 10. Fig. 10(a) shows that on the ADReSS training data, all the top 10 selected features have significant differences (P-value < 0.01) between the AD and HC. Fig. 10(b) shows a similar result on the AD2021 training data, except for the 1^{th} and 9^{th} features where no significant difference between the MCI and HC was found.

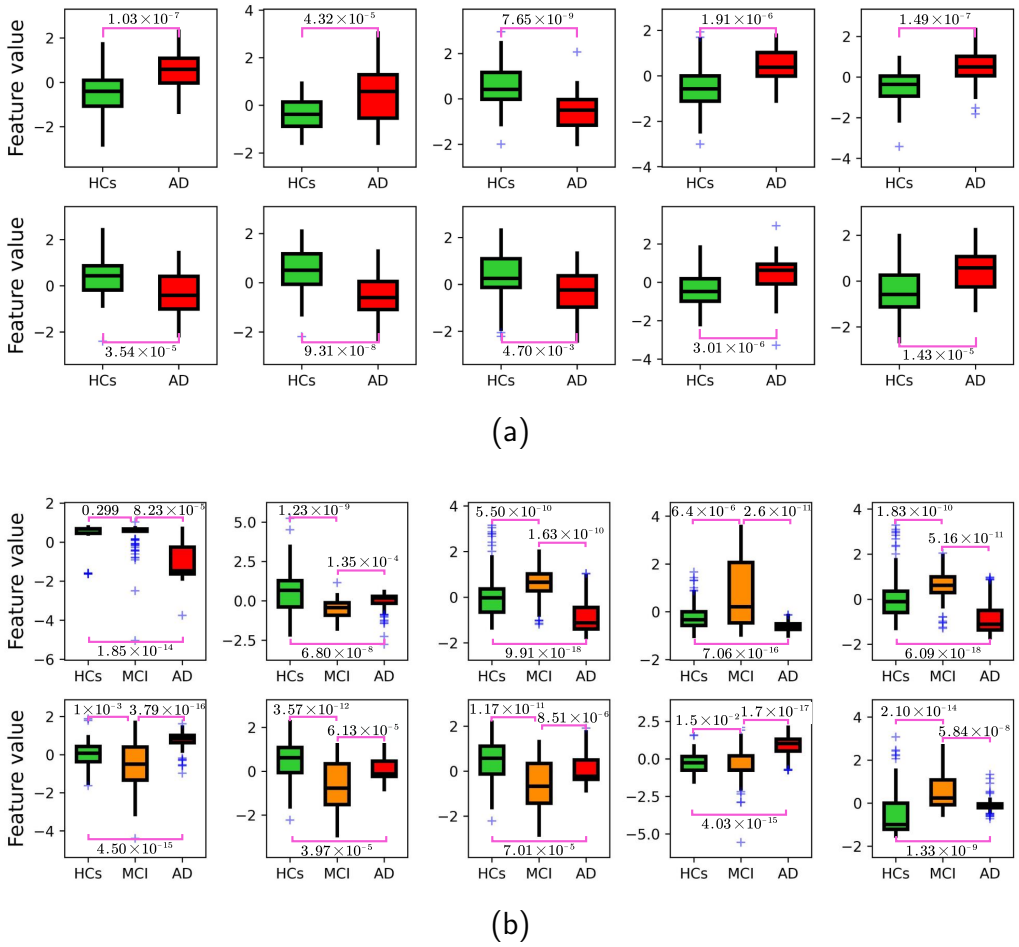

Figure 10: Box plots of the top 10 features selected by "MutInfo + DDR" on (a) the ADReSS and (b) the AD2021 training data. *AD*: Alzheimer's Disease, *MCI*: mild cognitive impairment, and *HC*: healthy control. In each box, the central line represents the median, and the bottom and top edges of the box represent the 25^{th} and 75^{th} percentiles, respectively. Outliers are shown as blue '+'. The *P*-values (two-tailed Wilcoxon rank-sum test) between AD, MCI, and HC for each selected feature are given.

6.8. Error Analysis

To better comprehend the limitations of our proposed approach, we analyzed the subjects who were correctly or incorrectly predicted by the classifier using the features selected by our FS method. Fig. 11 illustrates the numbers of correctly and incorrectly predicted subjects based on the test data in ADReSS and AD2021, respectively.

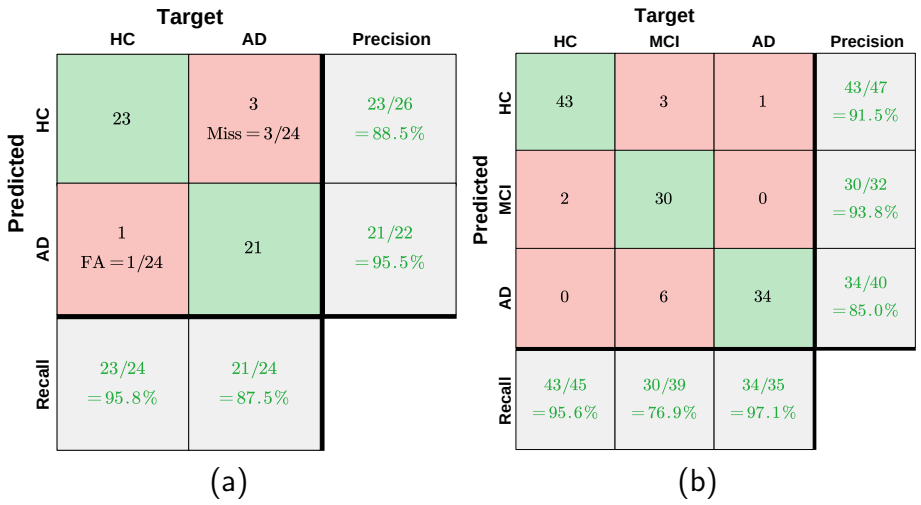

Figure 11: The subjects who were correctly or incorrectly predicted by the classifier using the features selected by our FS method based on the test data in (a) ADReSS and (b) AD2021. *AD*: Alzheimer's Disease, *MCI*: mild cognitive impairment, *HC*: healthy control, *FA*: false alarm.

As shown in Fig. 11(a), four subjects were incorrectly predicted (the pink boxes). In particular, a healthy subject was considered to have AD (a false alarm). Upon analyzing the transcription of this subject, we discovered that it is fairly short. Because a short transcription does not provide sufficient information for classification, it causes a false alarm. Three AD patients were considered healthy (misses). Unlike other AD patients, these patients happen to have long utterances, confusing the classifier because some linguistic features implicitly contain duration information, such as the number of words per minute (Table 9).

As shown in Fig. 11(b), twelve subjects were incorrectly predicted (the pink boxes). Among the three categories (HC, MCI, and AD), subjects having MCI were the most likely to be incorrectly predicted, with nine of them being incorrectly predicted. Since MCI serves as an intermediate stage between HC and AD, the differences between HC and MCI, as well as between MCI and AD, are less evident compared to those between HC and AD. More specifically, six subjects having MCI were considered to have AD, while three subjects having MCI were considered healthy. However, these two types of incorrect predictions have different consequences in medical practices. The former misinterprets the disease progression severity, while the latter may

fail to detect the onset of the disease, thereby preventing interventions to mitigate its progression at the early stage of the disease. To counteract this, we may apply a weighted loss to our FS training procedure by assigning greater weight to losses when the subjects having MCI are considered healthy. Additionally, one of the AD patients was considered healthy. A close analysis of the subject's audio revealed that while the subject was able to smoothly name several animals during the fluency test, the subject repeated some animals like "swallow" and "goat" twice. Adding repetition features to the feature set could help predict this kind of subjects correctly.

We further analyze the performance of BERT features and pause features on the ADReSS and AD2021 datasets. Table 8 shows that the performance of these two feature sets on the two datasets is different. Specifically,

1) The BERT features and pause features perform well on the ADReSS dataset, thanks to the accurate manual transcriptions and precise time alignments between the transcriptions and speech recordings. Some of the pause features were selected with high selection frequency (Fig. 9(a)).
2) In contrast, the AD2021 dataset renders the performance of these feature sets unsatisfactory due to the erroneous automatic transcriptions. Additionally, the timestamps detected by VAD are not sufficiently accurate for extracting the pause features. Consequently, none of the pause features is among the top 100 (Fig. 9(b)). Future work may develop a more efficient ASR system to improve the reliability of the transcriptions and investigate robust methods to mark the timestamps for speech activities.

7. Discussions and Conclusions

Our discussions commence with an examination of various studies on FS and its relevance to dementia detection. To identify AD patients, Haider *et al.* [3] combined various paralinguistic acoustic features – including eGeMAPS [4], ComParE 2013 [5], Emobase [5], and MRCG [6] – and applied PeaCorr tests to select the relevant features. The authors utilized PeaCorr tests to reduce the feature dimensionality of the combined feature vectors. However, the authors performed FS on the entire dataset without considering the selection frequency of individual features. In addition, they also identified the discriminative acoustic features for emotion recognition using the combined Emobase and eGeMAPS feature sets [47]. They introduced a new FS method called active feature selection (AFS) and compared its performance

with other FS methods. Nevertheless, because AFS evaluates feature subsets only, it cannot measure the significance of individual features. Weiner *et al.* [21] extracted various speech-based and transcription-based features from biographic interviews to predict AD after five years. The authors utilized forward FS to reduce the size of the initial feature set. A nested leave-one-subject-out CV was performed to determine the selection frequency of individual features. However, forward FS alone cannot determine the relative importance of individual features. Additionally, nested leave-one-subject-out CV is computationally expensive for large datasets or deep-learning-based FS methods. Alhanai *et al.* [23] identified discriminative features from demographic, audio, and text information for cognitive impairment detection. They employed a binomial logistic regression model regularized by an elastic-net for FS. Feature importance was determined using the coefficients of the regularized logistic regression model. Nevertheless, the use of nested leave-one-subject-out CV may be impractical for large datasets.

Our study introduces enhancements to FS for dementia detection based on the above research. For Step 1 of the two-step FS, we utilized the filter methods to pre-screen the original features. We conducted FS inside the CV instead of outside the CV, making the FS *nested* inside the learning process instead of being used as a pre-processing step. This makes individual folds select different features because the TR of individual folds are different. It is rational to nest FS inside the CV. If we conduct FS outside the CV, we will utilize both the TR and TS to select features and test the selected features on the TS, which will bias the performance. We adopted 10-fold CV instead of leave-one-subject-out CV for FS to avoid selection bias, as suggested by Ambroise *et al.* [48]. In the future, we will evaluate nested CV and bootstrap to see if these methods can further improve selection performance.

Our FS method has several limitations when compared with the filter methods that do not require training. For example, in the FDR, the selection variances of individual features depend on how we split the training data in the CV process. On the other hand, our FS method uses two neural networks to select features. The parameters of the trained networks depend on the initial weights and the random seed setting, causing an extra source of variation in addition to the random splits in the CV. Consequently, our method exhibits a higher selection variance.

In addition, during the CV, applying random splitting on a limited number of training samples will induce great differences across the TR. To mitigate the effect of random splittings, we propose an ensemble procedure to

repeat the 10-fold CV and average the predicted scores over all the CV. For the AD2021 dataset, we divided the training samples of the same speakers into either the TR or TS to avoid selecting the features that facilitate speaker recognition instead of dementia detection. For the ADReSS dataset, because accurate manual transcriptions are provided, we prefer using transcription-based features, whereas, for the AD2021 dataset, we include more speech-based features in addition to the transcription-based features because of the erroneous transcriptions.

To the best of our knowledge, this study is the first to exploit deep-learning-based FS methods to select spoken language biomarkers for dementia detection under limited training data scenarios. When the feature dimensionality is very large in relation to the number of training samples, the two-step FS approach can significantly reduce the feature dimensions and identify spoken language biomarkers that can achieve superior performance. Future work may investigate the biological aspects of the spoken language biomarkers. Readers interested in knowing the selected biomarkers can contact the corresponding author.

Acknowledgments

This work was in part supported by the Research Grants Council of Hong Kong, Theme-based Research Scheme (Ref.: T45-407/19-N). All the authors declare that they have no known competing interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Alternate Learning Algorithm of DDR

Require: Operator network with parameters ψ and selector network with parameters φ

Require: The size of dropout mask subset $|\mathcal{Z}|$, size of mini-batch $|\mathcal{M}|$, and number of training iterations n

Output: Dropout rates $\boldsymbol{\theta}_n$

Initialize dropout rates as $\boldsymbol{\theta}_0$
for $i \leftarrow 1$ to n **do**
    Obtain a dropout mask subset $\mathcal{Z}$ with size $|\mathcal{Z}|$ using Eq. (1)
    for $j \leftarrow 1$ to $|\mathcal{Z}|$ **do**

Compute the operator loss given $\boldsymbol{z}_i^{(j)}$:

$$\ell_{O,i}^{(j)} = \frac{1}{|\mathcal{M}|} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{M}} l(\boldsymbol{x} \odot \boldsymbol{z}_i^{(j)}, \boldsymbol{y}; \psi_i)$$

end for

Compute the average operator loss on $\mathcal{Z}$:

$$\mathcal{L}_O\left(\mathcal{M}, \mathcal{Z}; \psi_i\right) = \frac{1}{|\mathcal{Z}|} \sum_{j=1}^{|\mathcal{Z}|} \ell_{O,i}^{(j)}$$

Update operator network's parameters:

$$\psi_i \leftarrow \psi_i - \eta \nabla_\psi \mathcal{L}_O\left(\mathcal{M}, \mathcal{Z}; \psi_i\right)\big|_{\psi=\psi_i}$$

for $j \leftarrow 1$ to $|\mathcal{Z}|$ **do**

Compute the selector loss given $\boldsymbol{z}_i^{(j)}$:

$$\ell_{S,i}^{(j)} = \left| f_S(\boldsymbol{z}_i^{(j)}; \varphi_i) - \ell_{O,i}^{(j)} \right| \Big/ \sum_{k=1}^{d} (1 - z_{i,k}^{(j)})$$

end for

Compute the average selector loss on $\mathcal{Z}$:

$$\mathcal{L}_S\left(\mathcal{Z}(\boldsymbol{\theta}); \varphi_i\right) = \frac{1}{|\mathcal{Z}|} \sum_{j=1}^{|\mathcal{Z}|} \ell_{S,i}^{(j)};$$

Update selector network's parameters:

$$\varphi_i \leftarrow \varphi_i - \eta \nabla_\varphi \mathcal{L}_S\left(\mathcal{Z}(\boldsymbol{\theta}); \varphi_i\right)\big|_{\varphi=\varphi_i}$$

Update dropout rates:[17]

$$\boldsymbol{\theta}_i \leftarrow \boldsymbol{\theta}_i - \eta \sum_{j=1}^{|\mathcal{Z}|} \nabla_{\boldsymbol{z}(\boldsymbol{\theta})} \mathcal{L}_S\left(\mathcal{Z}(\boldsymbol{\theta}); \varphi_i\right) \nabla_{\boldsymbol{\theta}} \boldsymbol{z}(\boldsymbol{\theta})\big|_{\boldsymbol{\theta}=\boldsymbol{\theta_i}, \boldsymbol{z}=\boldsymbol{z}_i^{(j)}}$$

end for

[17]The gradient is based on the chain rule: $\frac{\partial \mathcal{L}_S}{\partial \boldsymbol{\theta}} = \frac{\partial \mathcal{L}_S}{\partial \boldsymbol{z}} \cdot \frac{\partial \boldsymbol{z}}{\partial \boldsymbol{\theta}}$.

References

[1] L. Mickes, J. T. Wixted, C. Fennema-Notestine, D. Galasko, M. W. Bondi, L. J. Thal, and D. P. Salmon, "Progressive impairment on neuropsychological tasks in a longitudinal study of preclinical Alzheimer's disease." *Neuropsychology*, vol. 21, no. 6, pp. 696–705, Nov. 2007.

[2] D. Beltrami, L. Calzà, G. Gagliardi, E. Ghidoni, N. Marcello, R. R. Favretti, and F. Tamburini, "Automatic identification of mild cognitive impairment through the analysis of Italian spontaneous speech productions," in *Proc. Int. Conf. Lang. Resourc. and Eval. (LREC)*, May 2016, pp. 2086–2093.

[3] F. Haider, S. de la Fuente, and S. Luz, "An assessment of paralinguistic acoustic features for detection of Alzheimer's dementia in spontaneous speech," *IEEE J. Sel. Top. Signal Process.*, vol. 14, no. 2, pp. 272–281, Feb. 2020.

[4] F. Eyben, K. R. Scherer, B. W. Schuller, J. Sundberg, E. Andre, C. Busso, L. Y. Devillers, J. Epps, P. Laukka, S. S. Narayanan, and K. P. Truong, "The Geneva minimalistic acoustic parameter set (GeMAPS) for voice research and affective computing," *IEEE Trans. Affect. Comput.*, vol. 7, no. 2, pp. 190–202, Apr. 2016.

[5] F. Eyben, M. Wöllmer, and B. Schuller, "openSMILE - the Munich versatile and fast open-source audio feature extractor," in *Proc. ACM Multimedia Int. Conf.*, Oct. 2010, pp. 1459–1462.

[6] F. Haider and S. Luz, "Attitude recognition using multi-resolution Cochleagram features," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2019, pp. 3737–3741.

[7] S. Nasreen, J. Hough, and M. Purver, "Detecting Alzheimer's disease using interactional and acoustic features from spontaneous speech," in *Proc. Interspeech*, Aug. 2021, pp. 1962–1966.

[8] L. Gauder, L. Pepino, L. Ferrer, and P. Riera, "Alzheimer disease recognition using speech-based embeddings from pre-trained models," in *Proc. Interspeech*, Aug. 2021, pp. 3795–3799.

[9] J. Shor, A. Jansen, R. Maor, O. Lang, O. Tuval, F. de Chaumont Quitry, M. Tagliasacchi, I. Shavitt, D. Emanuel, and Y. Haviv, "Towards learning a universal non-semantic representation of speech," in *Proc. Interspeech*, Oct. 2020, pp. 140–144.

[10] X. Li, S. Dalmia, J. Li, M. Lee, P. Littell, J. Yao, A. Anastasopoulos, D. R. Mortensen, G. Neubig, A. W. Black, and F. Metze, "Universal phone recognition with a multilingual allophone system," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2020, pp. 8249–8253.

[11] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "Wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. Adv. neural inf. proces. syst. (NIPS)*, Dec. 2020, pp. 12 449–12 460.

[12] M. Ströbel, E. Kerz, and D. Wiechmann, "The relationship between first and second language writing: Investigating the effects of first language complexity on second language complexity in advanced stages of learning," *Lang. Learn.*, vol. 70, no. 3, pp. 732–767, Apr. 2020.

[13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*. [Online]. Available: https://arxiv.org/abs/1810.04805

[14] Y. Sun, S. Wang, Y. Li, S. Feng, H. Tian, H. Wu, and H. Wang, "ERNIE 2.0: A continual pre-training framework for language understanding," in *Proc. AAAI Conf. Artif. Intell.*, Feb. 2020, pp. 8968–8975.

[15] Z. S. Syed, M. S. S. Syed, M. Lech, and E. Pirogova, "Tackling the ADRESSO challenge 2021: The MUET-RMIT system for Alzheimer's dementia recognition from spontaneous speech," in *Proc. Interspeech*, Aug. 2021, pp. 3815–3819.

[16] J. Li, J. Yu, Z. Ye, S. Wong, M. W. Mak, B. Mak, X. Liu, and H. Meng, "A comparative study of acoustic and linguistic features classification for Alzheimer's disease detection," in *Proc. IEEE Int. Conf. Acoust, Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 6423–6427.

[17] Y. Qiao, X. Yin, D. Wiechmann, and E. Kerz, "Alzheimer's disease detection from spontaneous speech through combining linguistic complexity and (dis)fluency features with pretrained language models," in *Proc. Interspeech*, Aug. 2021, pp. 3805–3809.

[18] S. Luz, F. Haider, S. de la Fuente, D. Fromm, and B. MacWhinney, "Detecting cognitive decline using speech only: The ADReSSo challenge," in *Proc. Interspeech*, Aug. 2021, pp. 4211–4215.

[19] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," 2018, *arXiv:1910.01108*. [Online]. Available: https://arxiv.org/abs/1910.01108

[20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*. [Online]. Available: https://arxiv.org/abs/1907.11692

[21] J. Weiner, C. Frankenberg, J. Schröder, and T. Schultz, "Speech reveals future risk of developing dementia: Predictive dementia screening from biographic interviews," in *Proc. IEEE Autom. Speech Recognit. Underst. Workshop (ASRU)*, Dec. 2019, pp. 674–681.

[22] R. B. Ammar and Y. B. Ayed, "Speech processing for early Alzheimer disease diagnosis: Machine learning based approach," in *Proc. IEEE/ACS Int. Conf. Comput. Syst. Appl. (AICCSA)*, Oct. 2018, pp. 1–8.

[23] T. Alhanai, R. Au, and J. Glass, "Spoken language biomarkers for detecting cognitive impairment," in *Proc. IEEE Autom. Speech Recognit. Underst. Workshop (ASRU)*, Dec. 2017, pp. 409–416.

[24] X. Ke, M. W. Mak, J. Li, and H. M. Meng, "Dual dropout ranking of linguistic features for Alzheimer's disease recognition," in *Proc. Asia-Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Dec. 2021, pp. 743–749.

[25] S. Luz, F. Haider, S. de la Fuente, D. Fromm, and B. MacWhinney, "Alzheimer's dementia recognition through spontaneous speech: The

ADReSS challenge," 2020, *arXiv:2004.06833*. [Online]. Available: https://arxiv.org/abs/2004.06833

[26] Y. Qin, W. Liu, Z. Peng, S.-I. Ng, J. Li, H. Hu, and T. Lee, "Exploiting pre-trained ASR models for Alzheimer's disease recognition through spontaneous speech," 2021, *arXiv:2110.01493*. [Online]. Available: https://arxiv.org/abs/2110.01493

[27] M. Wojtas and K. Chen, "Feature importance ranking for deep learning," in *Proc. Adv. neural inf. proces. syst. (NIPS)*, Oct. 2020, pp. 5105–5114.

[28] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[29] C. H. Chang, L. Rampasek, and A. Goldenberg, "Dropout feature ranking for deep learning models," 2017, *arXiv:1712.08645*. [Online]. Available: https://arxiv.org/abs/1712.08645

[30] Y. Gal, J. Hron, and A. Kendall, "Concrete dropout," in *Proc. Adv. neural inf. proces. syst. (NIPS)*, May 2017, pp. 3584–3593.

[31] J. Yuan, Y. Bian, X. Cai, J. Huang, Z. Ye, and K. Church, "Disfluencies and fine-tuning pre-trained language models for detection of Alzheimer's disease." in *Proc. Interspeech*, Oct. 2020, pp. 2162–2166.

[32] B. H. Davis and M. Maclagan, "Examining pauses in Alzheimer's discourse," *Am. J. Alzheimers Dis. Other Dement.*, vol. 24, no. 2, pp. 141–154, Apr. 2009.

[33] J. Yuan, M. Liberman *et al.*, "Speaker identification on the scotus corpus," *J. Acoust. Soc. Am.*, vol. 123, no. 5, p. 3878, May 2008.

[34] G. Degottex, J. Kane, T. Drugman, T. Raitio, and S. Scherer, "COVAREP — a collaborative voice analysis repository for speech technologies," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, May 2014, pp. 960–964.

[35] B. Schuller, S. Steidl, A. Batliner, F. Burkhardt, L. Devillers, C. Müller, and S. Narayanan, "The INTERSPEECH 2010 paralinguistic challenge," in *Proc. Interspeech*, Sep. 2010, pp. 2794–2797.

[36] S. Wang, D. Li, X. Song, Y. Wei, and H. Li, "A feature selection method based on improved Fisher's discriminant ratio for text sentiment classification," *Expert Syst. Appl.*, vol. 38, no. 7, pp. 8696–8702, Jul. 2011.

[37] J. Chen, M. Stern, M. J. Wainwright, and M. I. Jordan, "Kernel feature selection via conditional covariance minimization," 2017, *arXiv:1707.01164*. [Online]. Available: https://arxiv.org/abs/1707.01164

[38] Y. Li, C. Y. Chen, and W. W. Wasserman, "Deep feature selection: Theory and application to identify enhancers and promoters," *J. Comput. Biol.*, vol. 23, no. 5, pp. 322–336, May 2016.

[39] M. S. S. Syed, Z. S. Syed, M. Lech, and E. Pirogova, "Automated screening for Alzheimer's dementia through spontaneous speech," in *Proc. Interspeech*, Oct. 2020, pp. 2222–2226.

[40] L. Ilias, D. Askounis, and J. Psarras, "Detecting dementia from speech and transcripts using transformers," *Comput. Speech Lang.*, vol. 79, p. 101485, Apr. 2023.

[41] L. Yang, W. Wei, S. Li, J. Li, and T. Shinozaki, "Augmented adversarial self-supervised learning for early-stage Alzheimer's speech detection," in *Proc. Interspeech*, Sep. 2022, pp. 541–545.

[42] S. Ahmed, A.-M. F. Haigh, C. A. de Jager, and P. Garrard, "Connected speech as a marker of disease progression in autopsy-proven Alzheimer's disease," *Brain*, vol. 136, no. 12, pp. 3727–3737, Oct. 2013.

[43] W. Jarrold, B. Peintner, D. Wilkins, D. Vergryi, C. Richey, M. L. Gorno-Tempini, and J. Ogar, "Aided diagnosis of dementia type through computer-based analysis of spontaneous speech," in *Proc. Annu. Meet. Assoc. Comput Linguist.*, Jun. 2014.

[44] S. Luz, "Longitudinal monitoring and detection of Alzheimer's type dementia from spontaneous speech data," in *Proc. IEEE Symp. Comput.-Based Med. Syst.*, Jun. 2017.

[45] K. C. Fraser, J. A. Meltzer, and F. Rudzicz, "Linguistic features identify Alzheimer's disease in narrative speech," *J. Alzheimer's Dis.*, vol. 49, pp. 407–422, Nov. 2015.

[46] K. M. Robinson, M. Grossman, T. White-Devine, and M. D'Esposito, "Category-specific difficulty naming with verbs in Alzheimer's disease," *Neurology*, vol. 47, no. 1, pp. 178–182, Jul. 1996.

[47] F. Haider, S. Pollak, P. Albert, and S. Luz, "Emotion recognition in low-resource settings: An evaluation of automatic feature selection methods," *Comput. Speech Lang.*, vol. 65, p. 101119, Jan. 2021.

[48] C. Ambroise and G. J. McLachlan, "Selection bias in gene extraction on the basis of microarray gene-expression data," in *Proc. Natl. Acad. Sci. USA*, Apr. 2002, pp. 6562–6566.