https://dx.doi.org/10.1016/j.patrec.2023.07.016

Avoiding Dominance of Speaker Features in Speech-based Depression Detection

Lishi ZUO[a], Man-Wai MAK[a,**]

[a]*Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong SAR, China*

ABSTRACT

The performance of speech-based depression detectors is limited by the scarcity and imbalance in depression data. We found that depression detectors could be strongly biased toward speaker features when the number of training speakers is insufficient. To address this issue, we propose a speaker-invariant depression detector (SIDD) that minimizes speaker information in the latent space. The SIDD consists of an autoencoder, a depression classifier, and a speaker-embedding projector. By incorporating speaker-embedding vectors into the autoencoder's latent vectors, speaker information is effectively eliminated for the depression classifier. Experimental results demonstrate significant improvements achieved by minimizing speaker information, and our proposed method generally outperforms previous approaches for depression detection on the DAIC-WOZ dataset.

Keywords: Depression detection; speaker invariance; feature disentanglement; speaker embedding

1. Introduction

Depression is a prevalent and concerning public health issue, with growing evidence indicating an increase in its prevalence [1]. Traditional depression diagnosis relies on time-consuming and burdensome interviews that are either conducted by doctors or self-administered by patients [2]. The growing penetration of mobile phones has made remote screening and monitoring of depression increasingly promising. As physiological and cognitive changes caused by depression can influence the process of speech production, detecting depression based on speech is feasible [3, 4, 5].

Recent work demonstrates the effectiveness of deep learning-based depression detection. Various frameworks have been proposed to learn a latent representation of depression, including the recurrent neural network (RNN) family such as long short-term memory (LSTM) [6] and gate recurrent units (GRU) [7], convolutional neural networks (CNN) and their variants [8, 9], and transformers [10, 11].

The success of deep learning is driven by the abundance of data. Nevertheless, the availability of depression data is often limited by privacy concerns, resulting in relatively small speech corpora for depression studies. Training a depression detector on limited and imbalanced depression data can lead to the detector relying on non-depression features for detection. Consider the case where the training data contain a few speakers only. In such a scenario, the detector can use speaker features to achieve high performance on the training data because speakers with and without depression are mutually exclusive, i.e., the speaker IDs and the depression states are strongly correlated. In the extreme case of two training speakers only (one has depression and another is healthy), the depression detector becomes a speaker identifier. Because speaker features are more apparent in speech signals and are easier to learn, the detector takes less effort to decrease the training loss via identifying speakers instead of detecting depression. As a result, the detector will perform well on a validation/test set comprising the same speakers from the training set, but it will fail on a validation/test set comprising different speakers. Figure 1 demonstrates such a scenario using the speech of 77 speakers from the Distress Analysis Interview Corpus – Wizard-of-Oz (DAIC-WOZ) [12]. A large performance gap exists between testing the detector on

**Corresponding author.
e-mail: lishi.zuo@connect.polyu.hk (Lishi ZUO), man.wai.mak@polyu.edu.hk (Man-Wai MAK)

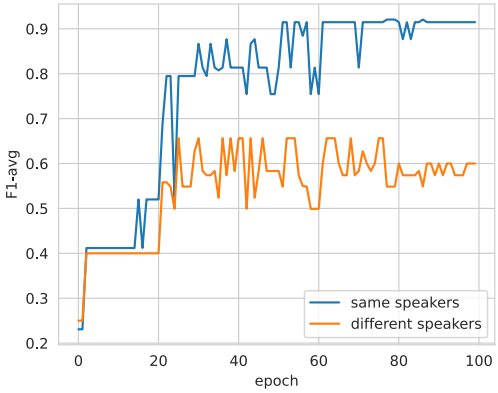

Fig. 1. A depression detector trained on a dataset with a small number of speakers could be biased toward speaker features. The figure shows the average F1 score of depression and non-depression classes (F1-avg) on two validation sets: one containing the same speakers in the training set and another containing speakers different from the training set. Each validation set contains 30 speakers (10 responses per speaker).

the training and non-training speakers, indicating that the model overfits to speaker features [13].

The significant performance difference between evaluating on training and non-training speakers has also been observed by Lopez-Otero *et al.* [14]. Unlike our approach, where the overfitting problem in deep learning is addressed, Lopez-Otero *et al.* [14] approached this issue from a traditional machine learning perspective and demonstrated that speaker information could be used as prior knowledge to narrow down the search space for depression detectors. They emphasized the significance of including speakers in the test set who differ from those in the training set. However, they did not provide a solution to tackle this problem.

Although bias toward unintended features is a serious issue, it has not been treated seriously. A few works have addressed this issue for different purposes. For example, Sardari *et al.* [15] used an autoencoder to learn latent representations from raw data. The method reduces feature dimensionality and removes unimportant information through compression, reducing the chance of overfitting to unintended features. However, for one thing, the compression process is uncontrollable because which feature has been removed is unknown. For another, the apparent and dominant speaker information is likely to be kept due to the compression nature of the autoencoder. Adversarial approaches have been employed to protect participants' privacy by removing personal information from data [16, 17]. For example, Ravi *et al.* [17] introduced an adversarial branch to their network to force the detector to ignore speaker information and reported performance improvements. However, adversarial training has stability issues, making it difficult to control the disentanglement of the speaker and depression features.

Inspired by [18, 19], we propose a speaker-invariant depression detector (SIDD), which uses a simple autoencoder and feature disentanglement to prevent the model from using speaker features for depression detection. The idea is to create a latent space with minimal speaker information so that the depression classifier can only *see* the depression features. By injecting speaker information through the projector (*Proj*) in the lower branch of Figure 2, the latent vector $\boldsymbol{z}$ does not need to contain rich speaker information for the decoder to reproduce the speaker-dependent depression vector $\boldsymbol{x}$. The irrelevant speaker information could be easily filtered out by feeding the speaker identity embedding to the decoder and by controlling the dimension of the bottleneck layer of the autoencoder.

The SIDD combines the advantages of [15] and [17]: 1) the autoencoder structure enables efficient information compression, reduces feature dimensionality, and eliminates irrelevant information to mitigate the risk of overfitting; 2) the SIDD minimizes the influence of speaker information on depression detection via speaker disentanglement, enabling the classifier to focus on depression-related features. More importantly, by using a vanilla autoencoder, the SIDD avoids the hassle of adversarial training in [17], resulting in a more stable training process.

The SIDD generally outperforms previous methods on DAIC-WOZ. We conducted experiments to evaluate the individual contributions of each component in the SIDD framework. Specifically, we show that: 1) minimizing speaker information in the latent vectors remarkably improves the performance of depression detection; and 2) a balanced number of feature vectors across all training speakers leads to a better performance.

2. Methodology

This section explains the architecture of the SIDD and its optimization strategy.

2.1. Architecture

Denote $\mathcal{S}$ as a training set comprising a collection of N tuples $\{(\boldsymbol{x}_i, \ell_i, \boldsymbol{u}_i)\}_{i=1}^{N}$, where $\boldsymbol{x}_i$ is a feature vector of dimension D, $\ell_i \in \{0, 1\}$ is the depression label of $\boldsymbol{x}_i$, and $\boldsymbol{u}_i$ is the one-hot representation of the speaker producing the i-th sample.

Figure 2 shows the architecture of the proposed SIDD. The encoder $Enc(\cdot)$ produces the speaker-invariant latent vector $\boldsymbol{z}_i$ by mapping $\boldsymbol{x}_i$ to a lower dimensional latent space. The projector $Proj(\cdot)$ is a single-layer linear network that maps the one-hot representation $\boldsymbol{u}_i$ to a vector with dimension matching that of $\boldsymbol{z}_i$. Then, $\boldsymbol{u}_i$ and $\boldsymbol{z}_i$ can be added and fed to the decoder. The decoder $Dec(\cdot)$ reconstructs the speaker-dependent depression vector $\hat{\boldsymbol{x}}_i$:

$$\hat{\boldsymbol{x}}_i = Dec\left(Enc\left(\boldsymbol{x}_i\right) + Proj\left(\boldsymbol{u}_i\right)\right). \tag{1}$$

The classifier $Cls(\cdot)$ outputs the posterior probabilities of depression:

$$\boldsymbol{y}_i = Cls\left(\boldsymbol{z}_i\right). \tag{2}$$

Ideally, the latent vector $\boldsymbol{z}_i$ is speaker-independent because speaker information in $\hat{\boldsymbol{x}}_i$ comes from $Proj(\boldsymbol{u}_i)$, i.e., the decoder does not require the latent vector $\boldsymbol{z}_i$ to contain speaker information. The intuition is

$$I(\boldsymbol{x}_i) \approx I(\boldsymbol{u}_i) + I(\boldsymbol{z}_i), \tag{3}$$

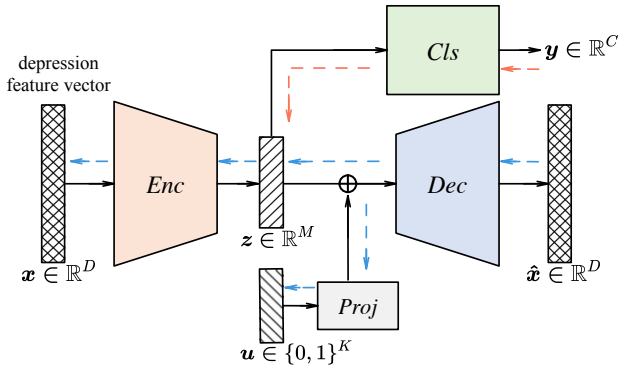

Fig. 2. The proposed speaker-invariant depression detector (SIDD). The architecture comprises four DNNs: an encoder (*Enc*), a depression classifier (*Cls*), a decoder (*Dec*), and a projector (*Proj*). In the figure, D is the dimension of the depression features (wav2vec latent vectors), M is the number of nodes in the bottleneck layer of the autoencoder, C is the number of depression classes, and K is the number of speakers in the training set. The dashed arrows represent backpropagation flows.

where $I(\cdot)$ stands for information.

The mutual information between $\boldsymbol{u}_i$ and $\boldsymbol{z}_i$ is expected to be zero. This could be achieved by tuning the dimension of $\boldsymbol{z}_i$ in the bottleneck layer. The dimension controls the amount of information loss when compressing $\boldsymbol{x}_i$ to $\boldsymbol{z}_i$. To be specific, 1) if the bottleneck layer is too wide, $\boldsymbol{z}_i$ will contain all information to perfectly reconstruct $\boldsymbol{x}_i$, meaning that the removal of speaker features will fail; 2) if the bottleneck layer is too narrow, $\boldsymbol{z}_i$ will not have enough depression information for the prediction; and 3) if the dimension of the bottleneck is properly set, the ideal condition in Eq. 3 could be achieved. This analysis is compatible with the descriptions in [18].

In practice, we introduce a bottleneck factor q ($0 < q \leq 1$) to control the dimension of the bottleneck layer, $M = qD$, where D is the dimension of the input vector. Then, the whole model structure is flexibly adjusted by q. All modules are fully connected neural networks with `tanh` activation function in their hidden nodes. The output layers of *Enc* and *Proj* are linear, and *Cls* has a `LogSoftmax` layer in its output. Detailed network structures are shown in Table 1, where q is set to 1/6.

Table 1. The network structure (Input, hiddens, output) of different modules in the framework in Figure 2. Refer to the caption of Figure 2 for the meaning of C, D, M, and K.

Module	Network Structure	Actual Implementation
Enc	D, $(D-(D-M)/2)$, M	1536, 896, 256
Dec	M, M, $(D-(D-M)/2)$, D	256, 256, 896, 1536
Cls	$(D-(D-M)/2)$, $16C$, C	256, 32, 2
Proj	K, $(D-(D-M)/2)$	107, 256

2.2. Training

The goals stated in Section 2.1 are achieved by optimizing two losses. For the autoencoder, $\boldsymbol{x}_i$ and $\hat{\boldsymbol{x}}_i$ should be as close as possible. We thus use the mean squared error:

$$\mathcal{L}_{rec} = \frac{1}{BD}\sum_{i=1}^{B} \|\boldsymbol{x}_i - \hat{\boldsymbol{x}}_i\|_2^2 , \tag{4}$$

where B is the number of training vectors in a mini-batch.

Let $\boldsymbol{y}_i = [y_{i0}, y_{i1}]^\top$, where y_{i0} and y_{i1} are the probability of non-depression and depression given $\boldsymbol{x}_i$, respectively. For notational convenience, we define

$$p_i = (1-\ell_i)y_{i0} + \ell_i y_{i1}. \tag{5}$$

For depression classification, we utilize the focal loss [20]:

$$\mathcal{L}_{cls} = -\frac{1}{B}\sum_{i=1}^{B} \alpha_{\ell_i}(1-p_i)^\gamma \log(p_i), \tag{6}$$

where α_{ℓ_i} is the class weighting factor for class ℓ_i and γ is the focusing parameter. In our case, $\ell_i = 0$ and $\ell_i = 1$ correspond to the majority class (non-depression) and the minority class (depression), respectively.

The total loss for training is

$$\begin{aligned}\mathcal{L}_{total}(\theta_{enc}, \theta_{dec}, \theta_{cls}, \theta_{proj}) =& \mathcal{L}_{rec}(\theta_{enc}, \theta_{dec}, \theta_{proj}) \\ &+ \mathcal{L}_{cls}(\theta_{cls}),\end{aligned} \tag{7}$$

where θ_{enc}, θ_{dec}, θ_{cls}, θ_{proj} denote the parameters of the encoder, decoder, depression classifier, and projector, respectively.

Note that the gradients will not be backpropagated from *Cls* to *Enc*, as illustrated by the red-dashed arrows in Figure 2. Given the premise that the model will be biased toward speaker features when training speakers are insufficient, we should prevent *Cls* from being exposed to speaker information. By not propagating the error gradient from *Cls* to *Enc*, we can prevent *Enc* from extracting speaker features to decrease $\mathcal{L}_{cls}$ when the number of speakers is small. Instead, the error gradient that propagates through *Enc* comes from *Dec* only. As a result, *Enc* attempts to extract depression features for *Dec* to reconstruct the feature vector $\boldsymbol{x}$. Evidence supporting this argument will be shown in Section 4.3.

3. Experiments

3.1. Dataset

DAIC-WOZ is a depression dataset containing clinical interviews of 189 participants [12]. The participants responded to a series of questions asked by a virtual interviewer controlled by a researcher sitting in another room. Each participant has one recording with a duration of up to 33 minutes. PHQ-8 [21] was used as the measurement of depression, and participants with PHQ-8 Scores $\geq$ 10 will be diagnosed as depressed. The dataset contains three subsets: a training set, a development set, and a test set. The details of DAIC-WOZ can be found in Table 3.

Table 3. Summary of the number of speakers in DAIC-WOZ.

	Depressed	Non-depressed	Total
Training	30	77	107
Development	12	23	35
Test	14	33	47

Table 2. Performance of the proposed SIDD on the development set and the test set.

	Development			Test		
	F1 (ND)	**F1 (D)**	**F1-avg**	**F1 (ND)**	**F1 (D)**	**F1-avg**
Mean±STD	0.866±0.018	0.745±0.023	0.805±0.020	0.721±0.039	0.481±0.037	0.601±0.035
Best	0.894	0.783	0.838	0.781	0.533	0.657
Avg.Top-5	0.869	0.751	0.810	0.773	0.524	0.648

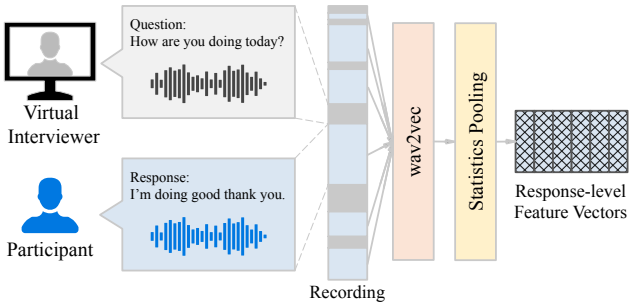

Fig. 3. Our strategy to segment conversations into responses (speaker turns). The raw speech within a speaker turn is processed by wav2vec and then statistically pooled to obtain a response-level vector. Each vector contains information of a response.

3.2. Data Preprocessing

As shown in Figure 3, each speech recording was segmented into a number of responses (denote as "response-level"), and the segments corresponding to the virtual interviewer were discarded. Because wav2vec [22] has shown remarkable performance under data sparsity scenarios, we used a wav2vec model pretrained on LibriSpeech [23] to extract features in all experiments. After feature extraction, the frame-based features were statistically pooled into vectors. The mean across time, the standard deviation across time, and the mean first-order difference between the successive feature frames were calculated. The three 512-dimensional vectors were then concatenated to form a 1536-dimensional vector, which is $\boldsymbol{x}$ in Figure 2.

3.3. Evaluation

Following prior works [8, 17, 24, 25], we used the F1 score as the evaluation metric, which is the harmonic mean of precision and recall. The F1 score considers class imbalance, ensuring unbiased model performance. The F1 scores of the depression class and the non-depression class are denoted as F1(D) and F1(ND), respectively. In addition, the mean of the F1 scores of the two classes (F1-avg) was also calculated. The final prediction of a participant is obtained by majority votes. To improve the reliability of the F1 scores, we randomly generated 20 seeds to initialize the networks and kept the seeds the same for all experiments. The mean performance of 20 runs of each experiment is reported.

3.4. Network Optimization

We optimized the model using an Adam optimizer and early stopping to ensure the networks converge without overfitting to the training data. The hyperparameters, such as the number of epochs, learning rate, and batch size, were empirically set. Specifically, the learning rate was set to 10^{-4}. We first gradually warmed up the learning rate in the first 50 epochs and then adjusted the learning rate using a cosine annealing scheduler in the following epochs. The batch size was set to 256. To avoid overfitting to the training set, we ran 500 epochs with early stopping based on the F1(D) score on the development set. Class weights (α_{ℓ_i} in Eq. 6) were set to the inverse class frequencies to alleviate the class-imbalance problem. The focusing parameter γ was set to 2, consistent with its use in [20].

4. Results

4.1. Performance of SIDD

Table 2 summarizes the results of the proposed SIDD. Since it is not unusual to observe large performance variations in small datasets, we report the mean and standard deviation (STD) of 20 repeated experiments. We also show the best and the average of the top five (Avg.Top-5) results.

4.2. Comparing with State-of-the-Art

Generally, as shown in Table 4, our proposed SIDD outperforms all other methods on the development set and the test set. The methods in [24] and [25] perform poorly on the test set. It may be because they are pure encoders without any design to avoid overfitting to unintended features. The methods that remove speaker information [17] perform better than pure depression detectors [24, 25]. The method in [8] achieves the second-best F1-avg score. One possible reason is that it employs domain adaptation by pretraining the models on another depression dataset, implicitly increasing the number of training speakers, and thus easing the curse of overfitting. By combining the advantages of [15] and [17], our proposed SIDD outperforms the second-best method [8] by 4% on the development set. Note that we did not compare with [15] because it did not apply majority votes or average the probabilities of all segments of the same speakers for final prediction.

We also observed a big difference in performance between the development set and the test set, which is consistent with the results in [24] and [25]. We conjecture that it is due to the serious mismatch between the development set and the test set in DAIC-WOZ.

4.3. Ablation Study

To evaluate the contributions of individual modules in the proposed framework, we reconfigured the structure in Figure 2 into two networks: plain classifier and plain autoencoder. Figure 4 shows the structures of the ablated networks. In the plain

Table 4. Comparing with state-of-the-art. The results in the test set of the competing methods were obtained from [25].

Method	Development			Test
	F1(ND)	F1(D)	F1-avg	F1-avg
DepAudioNet [24]	0.700	0.520	0.610	0.380
FRAUG [25]	-	-	0.656	0.479
Disentanglement [17]	0.808	0.576	0.692	-
Domain Adaptation [8]	0.860	0.670	0.765	-
SIDD (Ours)	**0.866**	**0.745**	**0.805**	**0.601**

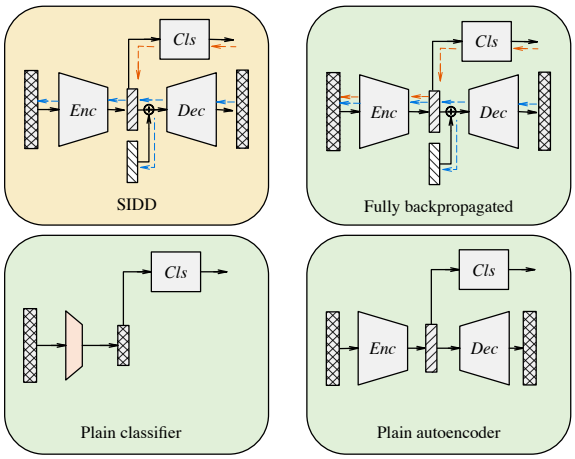

Fig. 4. Illustration of the structures of the ablated networks in the ablation study.

classifier, *Enc* and *Dec* were removed, meaning that *Cls* was trained using the given feature vectors $\boldsymbol{x}_i$'s. A linear layer with 1536 inputs and 256 outputs was prepended to *Cls* to match the dimension of $\boldsymbol{x}_i$ and the input of *Cls*. In the plain autoencoder, *Proj* and $\boldsymbol{u}_i$ were removed, i.e., the speaker information $\boldsymbol{u}_i$ was not provided to *Dec*. In addition, we allowed the gradients to be backpropagated from *Cls* to *Enc* (denote as "Fully backpropagated"). Table 5 shows the ablation results on the development set and the test set.

Overall, the performance improved every time we added one more component. Compared to the plain classifier, the F1-avg scores of the plain autoencoder on the development set and the test set were increased by 7.7% and 6.0%, respectively. This is because the autoencoder compresses the features to a low dimensional space, suppressing some unimportant information, thus reducing the risk of overfitting to non-depression features and easing the extraction of depression features. By filtering out speaker information from input vectors, the performance further improves by 1.8% and 4.1% on the development and the test sets, respectively.

Table 5 shows that the performance of "Fully backpropagated" is even worse than the plain classifier. Given the access to the input vectors rich in speaker information, the classification loss takes advantage of *Enc* to overfit to speaker features when the number of speakers is limited, making the depression detector become a speaker classifier. Figure 5 shows a significant increase in speaker separability when allowing the gradients to be backpropagated from *Cls* to *Enc*.

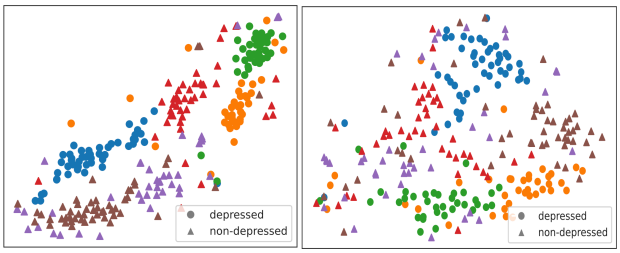

Fig. 5. T-SNE plots of z of 6 speakers using the proposed method with (left) and without (right) backpropagating $\mathcal{L}_{cls}$ from *Cls* to *Enc*. The colors represent speakers.

Due to the small size of the DAIC-WOZ dataset, significance tests were conducted to verify whether the SIDD truly outperforms the ablated models. As shown in Figure 6, the experimental results demonstrate that the proposed SIDD is significantly better (p-*value* < 0.01) than the ablated structures.

A noticeable result in Table 5 is that our plain classifier has reached a high F1-avg at 0.71 on the development set. Surprisingly, it defeats many complex structures in [17, 24, 25]. The recipe that enables the good performance of the plain classifier is that the numbers of response-level vectors in our setting are relatively balanced across speakers because all speakers have a similar number of responses (around 44 responses per speaker). The importance of balancing the numbers of training vectors across speakers will be discussed in Section 4.4.

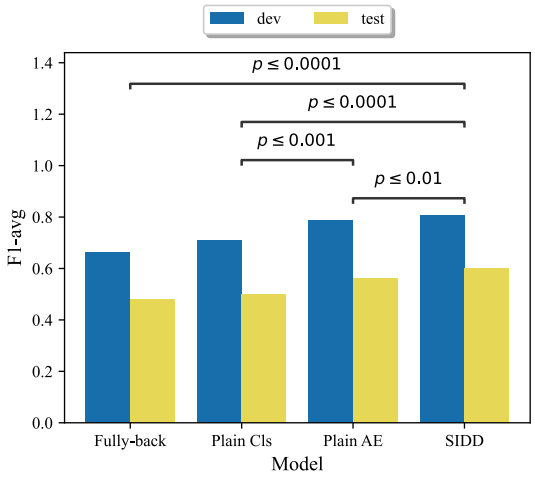

Fig. 6. *t-Test* comparison of SIDD, plain autoencoder (Plain AE), plain classifier, and fully backpropagated model (Fully-back) in the ablation study. The corresponding p-*value* ranges shown in the figure indicate different levels of statistical significance.

4.4. Number of Training Vectors per Speaker

We conducted experiments using the plain classifier to show the importance of a balanced number of training vectors per speaker across all speakers in the training set. For simplicity, the "number of training vectors per speaker" is referred to as "vectors-per-speaker (VPS)". Table 6 shows the F1-avg scores on the development set under different VPS. Table 6 also shows the coefficient of variation (the ratio between the

Table 5. Ablation study on the development set and the test set of DAIC-WOZ.

Model	Development			Test		
	F1 (ND)	F1 (D)	F1-avg	F1 (ND)	F1 (D)	F1-avg
Fully backpropagated	0.726	0.598	0.662	0.594	0.367	0.480
Plain classifier	0.790	0.630	0.710	0.642	0.359	0.500
Plain autoencoder	0.847	0.727	0.787	0.672	0.448	0.560
Proposed SIDD	0.866	0.745	0.805	0.721	0.481	0.601

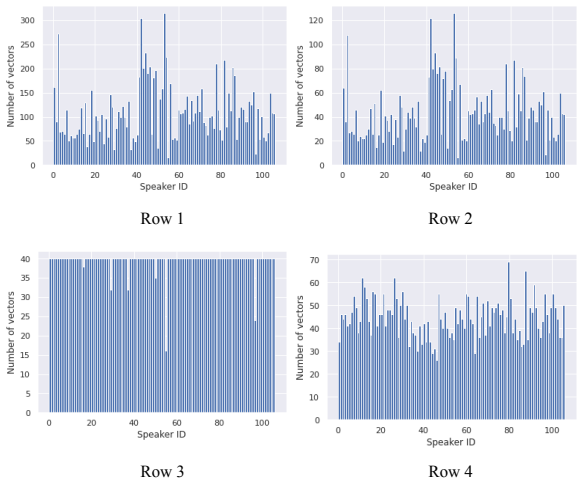

Fig. 7. Illustration of the dispersion of four different vectors-per-speaker (VPS) in the experiments. Refer to Table 6 for the detailed information of Rows 1, 2, 3, and 4.

standard deviation of the numbers of vectors per speaker and their mean). Through this measure, we can fairly compare the dispersion of different VPS. Figure 7 shows the VPS of all speakers corresponding to the four conditions in Table 5.

In Rows 1, 2, and 3 of Table 6, the speech recording of each speaker was segmented to the same length (3.84s[1]), denoted as "segment-level". In detail, Row 1 used all segment-level vectors for training. The VPS in Row 1 is fairly imbalanced because it fluctuates by 53.3%. Row 2 used around 40% of the segment-level training vectors in Row 1, keeping the dispersion of VPS as close to that of Row 1 as possible. Row 3 sampled around 40 segment-level training vectors per speaker, keeping the VPS more or less balanced. For better comparison, we kept the average number of training vectors per speaker to around 40 in Rows 2 and 3.

Comparing Row 3 to Row 2, we found that simply sampling the training set to balance the VPS could increase the F1-avg score by 11.2%. The reason might be that imbalanced VPS makes some speakers with more training data dominate, causing the plain classifier to be biased toward speaker features of those dominant speakers when the number of speakers is small. In contrast, under an imbalanced VPS, adding more segments of the existing speakers would not help improve the performance (see Rows 1 and 2). Row 4 shows the F1-avg score of the response-level training vectors in our setting. It performs the best partly because it uses all data while keeping the VPS relatively balanced.

[1]Follows the setting in [24].

Table 6. Impact of different vectors-per-speaker (VPS) on the development set.

Row	Vector type	# of vectors	Balanced	VPS		F1-avg
				Mean	$\frac{\text{STD}}{\text{Mean}}$	
1	Segment-level	11753	N	109.84	0.533	0.521
2	Segment-level	4657	N	43.52	0.539	0.537
3	Segment-level	4217	Y	39.41	0.076	0.649
4	Response-level	4758	Y	44.00	0.188	0.710

5. Conclusions

In this study, we show that the depression model trained on a dataset with a limited number of speakers is likely to be biased toward speaker features, and we demonstrate a significant performance improvement by minimizing speaker information using the proposed autoencoder-based SIDD. Furthermore, we show that a balanced number of training vectors across speakers is fairly important for depression detection.

However, it is important to acknowledge the limitations of the SIDD, particularly its reliance on pretrained vector-based features. Future research should explore the potential of speaker disentanglement models based on frame-based features, which may extract more specific and discriminative features related to depression.

We demonstrated that the availability of large and balanced datasets is crucial to advance depression detection. It is necessary to consider the negative impact of the scarcity and imbalance in depression data, and develop methods to lessen it. Additionally, exploring which features the depression detectors will be biased toward is an important issue for future research because these features could decide what knowledge the models could learn.

One promising direction for future research is to incorporate other prior domain knowledge, such as gender information, into the design of depression detectors, reducing the risk of bias and overfitting to unintended features. We recommend exploring the insights from [13] to better understand how models learn and to guide the development of more robust depression detection methods.

Acknowledgements

This work was in part supported by the National Natural Science Foundation of China (NSFC), Grant No. 61971371.

References

[1] Jonathan Rottenberg. The prevalence of depression. *Depression*, pages 29–30, 2021.

[2] Katie M Smith, Perry F. Renshaw, and John A. Bilello. The diagnosis of depression: Current and emerging methods. *Comprehensive Psychiatry*, 54 1:1–6, 2013.

[3] Nicholas Cummins, Stefan Scherer, Jarek Krajewski, Sebastian Schnieder, Julien Epps, and Thomas F Quatieri. A review of depression and suicide risk assessment using speech analysis. *Speech Communication*, 71:10–49, 2015.

[4] Klaus R Scherer. Vocal affect expression: A review and a model for future research. *Psychological Bulletin*, 99(2):143, 1986.

[5] Gary Christopher and John MacDonald. The impact of clinical depression on working memory. *Cognitive Neuropsychiatry*, 10(5):379–399, 2005.

[6] Tuka Al Hanai, Mohammad M Ghassemi, and James R Glass. Detecting depression with audio/text sequence modeling of interviews. In *Proc. Interspeech*, pages 1716–1720, 2018.

[7] Ying Shen, Huiyu Yang, and Lin Lin. Automatic depression detection: An emotional audio-textual corpus and a GRU/BiLSTM-based model. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6247–6251, 2022.

[8] Zhaocheng Huang, Julien Epps, Dale Joachim, Brian Stasak, James R. Williamson, and Thomas F. Quatieri. Domain adaptation for enhancing speech-based depression detection in natural environmental conditions using dilated cnns. In *Proc. Interspeech*, pages 4561–4565, 2020.

[9] Qian Chen, Iti Chaturvedi, Shaoxiong Ji, and Erik Cambria. Sequential fusion of facial appearance and dynamics for depression recognition. *Pattern Recognition Letters*, 150:115–121, 2021.

[10] Weidong Chen, Xiaofen Xing, Xiangmin Xu, Jianxin Pang, and Lan Du. Speechformer: A hierarchical efficient framework incorporating the characteristics of speech. In *Proc. Interspeech*, pages 346–350, 2022.

[11] Weidong Chen, Xiaofen Xing, Xiangmin Xu, Jianxin Pang, and Lan Du. Speechformer++: A hierarchical efficient framework for paralinguistic speech processing. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023.

[12] Jonathan Gratch, Ron Artstein, Gale Lucas, Giota Stratou, Stefan Scherer, Angela Nazarian, Rachel Wood, Jill Boberg, David DeVault, Stacy Marsella, David Traum, Skip Rizzo, and Louis-Philippe Morency. The distress analysis interview corpus of human and computer interviews. In *Proc. the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3123–3128, 2014.

[13] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[14] Paula Lopez-Otero, Laura Docio-Fernandez, and Carmen Garcia-Mateo. Assessing speaker independence on a speech-based depression level estimation system. *Pattern Recognition Letters*, 68:343–350, 2015.

[15] Sara Sardari, Bahareh Nakisa, Mohammad Naim Rastgoo, and Peter W. Eklund. Audio based depression detection using convolutional autoencoder. *Expert Systems with Applications*, 189:116076, 2022.

[16] Paula Lopez-Otero and Laura Docio-Fernandez. Analysis of gender and identity issues in depression detection on de-identified speech. *Computer Speech and Language*, 65:101118, 2021.

[17] Vijay Ravi, Jinhan Wang, Jonathan Flint, and Abeer Alwan. A step towards preserving speakers' identity while detecting depression via speaker disentanglement. In *Proc. Interspeech*, pages 3338–3342, 2022.

[18] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson. Autovc: Zero-shot voice style transfer with only autoencoder loss. In *Proc. the 36th International Conference on Machine Learning (ICML)*, volume 97, pages 5210–5219, 2019.

[19] Ju-Chieh Chou, Cheng-chieh Yeh, Hung-yi Lee, and Lin-Shan Lee. Multi-target voice conversion without parallel data by adversarially learning disentangled audio representations. In *Proc. Interspeech*, pages 501–505, 2018.

[20] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007, 2017.

[21] Kurt Kroenke, Tara W Strine, Robert L Spitzer, Janet BW Williams, Joyce T Berry, and Ali H Mokdad. The PHQ-8 as a measure of current depression in the general population. *Journal of Affective Disorders*, 114(1-3):163–173, 2009.

[22] Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. wav2vec: Unsupervised pre-training for speech recognition. In *Proc. Interspeech*, pages 3465–3469, 2019.

[23] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: An ASR corpus based on public domain audio books. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210, 2015.

[24] Xingchen Ma, Hongyu Yang, Qiang Chen, Di Huang, and Yunhong Wang. Depaudionet: An efficient deep model for audio based depression classification. In *Proc. the 6th International Workshop on Audio/Visual Emotion Challenge*, pages 35–42, 2016.

[25] Vijay Ravi, Jinhan Wang, Jonathan Flint, and Abeer Alwan. Fraug: A frame rate based data augmentation method for depression detection from speech signals. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6267–6271, 2022.