IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 2, 2021, accepted April 18, 2021, date of publication April 30, 2021, date of current version May 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3076820

Automatic Personality Traits Perception Using Asymmetric Auto-Encoder

EFFAT JALAEIAN ZAFERANI[1], MOHAMMAD TESHNEHLAB[1], AND MANSOUR VALI[2]
[1]Intelligent Systems Laboratory, Faculty of Electrical and Computer Engineering, K. N. Toosi University of Technology, Tehran 16317-14191, Iran
[2]Speech Processing Laboratory, Faculty of Electrical and Computer Engineering, K. N. Toosi University of Technology, Tehran 16317-14191, Iran

Corresponding author: Mohammad Teshnehlab (teshnehlab@eetd.kntu.ac.ir)

ABSTRACT On account of an increase in the human-computer interface applications, the study of automatic personality perception has become more and more prevalent than speech signal processing in recent years. These studies have shown that personality traits derived from psychology theories mainly affect acoustic features. However, some obstacles remain in the automatic personality perception classification, and the most important one is to extract the features related to each personality trait. Previous studies have shown that the personality effect differs from one acoustic feature to the others. Additionally, there are many features one can extract from speech signals. Curse of dimensionality in features also makes the classification difficult. This paper aimed to introduce and examine a novel and efficient automatic feature extraction method to classify the well-known big five personality traits. In this regard, three data augmentation methods for increasing data samples were examined. Afterwards, 6,373 statistical features were extracted from the nonverbal features of the SSPNet Speaker Personality Corpus. Finally, an innovative stacked asymmetric auto-encoder was utilized to extract useful features automatically to improve classification results. Compared with the conventional stacked auto-encoder and convolutional neural network, the proposed method exhibited an average improvement of 12.40%(10.14%) and 14.36%(1.42%) in terms of the unweighted average recall (accuracy), respectively. In comparison with other published works, classification results also revealed a notable average enhancement (11.78%) for unweighted average recall for all five traits and an average improvement of 5.1% for accuracy in two out of five personality traits.

INDEX TERMS Asymmetric auto-encoders, big five inventory, curse of dimensionality, data augmentation, deep neural networks, feature extraction, semi-supervised learning, personality traits.

I. INTRODUCTION

The importance of Personality Perception (PP) from speech has increased with the advancement of science and technology, since in today's world of digital and cyberspace, analyzing individuals' perceived personality (not true personality) can help to grow these spaces smarter [1]. Regarding the field of the personality-related human-computer interface [2], a wide range of interesting applications, such as in-car voice assistants [3], personalization of user interfaces [4], user-problematic behaviors in cyberspace [5], troubleshooting a cold start problem for new users of virtual service provider systems [6] has been developed over the recent years.

In fact, the PP model describes the relationship between a set of features of the speech signals and five measurable personality traits derived from the psychology theory of the Big Five Inventory (BFI) [7]. These traits are introduced as Openness to experience (Ope.), Conscientiousness (Con.), Extraversion (Ext.), Agreeableness (Agr.), and Neuroticism (Neu.) [8], [9].

Aiming to find a significant relationship between speech characteristics and personality traits, several audio features were extracted manually by the data scientist (hand-crafted features). In this regards, Mairesse *et al.* explored the correlation between acoustic-prosodic, psychological features, and personality traits [10]. In [11] and [12], the authors have classified personality by extracting statistical parameters from speaking styles such as prosodic features, intonation, and voice quality. The frequency-domain linear prediction and Mel Frequency Cepstral Coefficient (MFCC) features were

The associate editor coordinating the review of this manuscript and approving it for publication was Maurizio Tucci.

extracted in [13]. In [14], the acoustic and duration-related features, such as silence ratio, speech duration ratio, and speech rate features, were proposed. Moreover, Guidi *et al.* evaluated the impact of median/mean of the fundamental frequency of speech signal, the frame-to-frame jitter factor, and the glottal flow spectral slope features on personality recognition [15].

Studies were not limited only to the acoustic and nonverbal features extraction. The lexical [8], [16], knowledge-based features [14], and BFI questionnaire scores [17] were added to this variety of features as well. The effort toward achieving a wider variety of features reached 6,373 statistical feature in [18].

Although the above-mentioned studies were an excellent initiation for APP, they faced several challenges. Herein, we described three important challenges in the field of Automatic Personality Perception (APP) as follows and then introduced our solution to overcome them:

1. Studies have indicated that a specific feature set and model could not outperform all the five traits [19]. Therefore, the most critical challenge is to extract the appropriate feature sets for classifying every five traits individually.
2. A high-dimensional feature set poses a curse of dimensionality problem. This phenomenon deteriorates the model's performance through irrelevant features [20].
3. By increasing the number of features, the trainable model parameters increase, and more samples would be needed for the complete training process as a result.

In this regard, some techniques such as a sequential direct search algorithm [21], the correlation-based feature selection method [22], mutual information [18] and feature selection based on classification/statistical dependence [23] were introduced to mitigate the problem of dimensionality curse and reduce the trainable parameters of the model to extract appropriate feature sets. Although the classification results were improved, the performance was still insufficient.

Before the advent of deep learning methods (DLMs), the lack of effective methods to cope with high-dimensional data and automatic feature extraction methods were obvious.

The development of DLMs in PP applications had been growing in popularity afterwards. The Deep Belief Networks (DBNs) and Auto-Encoder (AE) were applied to the low-level features in [24], and the results outperformed compared with some baseline methods. In [25], Stacked Auto-Encoder (S_{AE}) and Long Short-Term Memory (LSTM) on four hand-crafted feature sets were applied. The outcomes exhibited an average improvement of 2% for all of the dataset, except for one of them, for which no improvement was specified.

Classifying prosodic features by Support Vector Machine (SVM) were compared with applying the LSTM to MFCC features in [26].

The authors in [27] employed a two-modality dataset (video and audio) for APP. A VGG-face model together with the Convolutional Neural Networks (CNN) were used for processing the video dataset. In audio processing, however, the deep learning technique was applied on log filter bank features. This trend was the first-place winner of the ChaLearn 2016 APP competition [28].

Thereafter, the CNN method was employed for feature extraction on the raw audio data of ChaLearn 2016 dataset in [27]. Although the outcomes were notable, they were diminished for all the traits in comparison with the first winner of the competition.

Despite the ability of feature extraction of DLMs, considering published works in APP confirmed that the DLMs were applied on the hand-crafted audio features, not on the raw speech signals. The main reason is that DLMs need a large dataset to outperform. However, one major obstacle in APP has been to provide a suitable, available and annotated dataset [19], which has limited the study of various methods [19], [28]. In this paper, recently published data augmentation methods were used to cope with the limited samples of the dataset.

The other mentioned problems are extracting the appropriate feature sets and pose a curse of dimensionality issue. DLMs sift all inputs to find patterns (unsupervised learning) in a way that no one would normally be able to obtain [29]. To the best of our knowledge, there are different types of patterns in speech signals related to emotion detection [30], deception detection [31], and speech recognition [32]. There is no need to extract all patterns but the appropriate ones for our target (supervised learning). On the other hand, fine-tuning of DLMs parameters causes the vanishing gradient problem, which seriously dominates the extraction of target-appropriate patterns [33]. To mitigate the learning problems of deep methods and curse of dimensionality, we proposed a Stacked Asymmetric Auto-Encoder (SA_{AE}) as semi-supervised feature extractor from the hand-crafted audio features.

The rest of this paper is ordered as follows: related works are presented in section II; the dataset and the three data augmentation methods are defined and introduced in section III; section IV outlines the feature extraction based on the proposed strategy; the results of the simulations are given in section V; to evaluate the efficiency of our novel method, comparison table is discussed in this section; finally, the conclusions are debated in section VI.

II. RELATED WORKS

Classifying personality traits in a two-way dialogue was the aim of the article [34]. By applying the coupled Hidden Markov Model (HMM) classification to the correlation between BFI scores and linguistic features, the obtained accuracy lied between 59.5% and 86.8%.

The speaker's automatic trait prediction was studied based on the acoustic and prosodic features extraction from speech, using the neural network classifier [35]. The reported accuracy did not exceed the range of 68.68% to 81.63%.

The SVM classifier was applied to acoustic and lexical features [16]. The Unweighted Average (UA) recall results were 74%–80%. In another study, the spearman correlation was applied to deceptive and non-deceptive speech as a feature selection method, and Sequential-Minimal-Optimization (SMO) classification was considered. Finally, the range of UA was 37%–44% [16]. These authors' advanced paper was [36], in which they used the normalized forms of the same dataset and features. They examined separate models for each gender and test results by Weka's SVM classifier. The results exhibited a significant accuracy improvement.

The purpose of [25] was to predict the speaker's behavior by nonverbal features. The authors considered the relationship between the speech signal and personality traits using spectral properties. The K-Nearest Neighbors (KNN) clustering results were compared to the SVM, and the range of accuracy was between 60% to 92%.

In a series of researches [37]–[39], the authors classified the personality traits by extracting nonverbal features. They improved the accuracy of classification results to the range of 70.1%–88.8% through the SVM classifier and logistic regression. Although the accuracy results were significant, the confusion matrix in the article [12] showed that the recall average of the low and high class was between the 53% and 65% range. They also expressed the effect of the number of the BFI questionnaire evaluator variation on classification and regression results.

The [26] manifests APP from speech signals by a skip-frame LSTM system. the LSTM was applied on a low-level feature set, and the empirical results showed outperforms. Obviously, the LSTM system can learn personality information better than the traditional SVM system via extracting time-series information. In [26], deep learning performance on six sub-traits of BFI was examined and discussed.

In [9], the variance of unknown differences in judgment's perception of modified speech targets with hierarchical clustering reduces. Before the feature extraction step, three filters were applied to the audio clips to avoid the uncertain effects of noise, silence, and pitch. Some features, such as pitch, pause rate, and power roll-off, are classified by three classifiers (SVM, KNN, and logistic regression). The accuracy results are between 65.3% and 76.3%.

The unsupervised cross-modal feature learning algorithm was proposed in [24]. In this model, the low-level features were extracted first. Feature learning from the first set was then employed based on DBNs and AE to recognize the three modalities' personality traits. The accuracy results demonstrated that the proposed method outperforms compared with the baseline methods.

The ChaLearn 2016 personality perception competition was done, whose results are presented in [28]. The goal was APP from a multimodal dataset. The outcomes of the successful nine team's approaches from 42 teams were presented and discussed.

The first-place team proposed two separate models for video and audio. Subsequently, they used a late fusion method. For the video modality, A VGG-face model with two layers of the CNN and a fully connected layer as the output was used for pre-training (Transfer learning method). The video dataset of completion fine-tuned the model. In the audio processing step, log filter bank features were fed to a fully connected layer with sigmoid activation functions [27].

The second-place team extracted the statistical parameters from the spectral features for audio modality. The video clips were preprocessing with face alignment. The output was fed to a Recurrent CNN (RCNN) and trained end-to-end. The audio features were fused with the video features. Therefore, personality traits were predicted by RCNN [40].

The authors of [25] tried to classify personality traits using DLMs. They applied S_{AE} and LSTM. The low-level descriptor features, two lexical dictionaries features and word embedding feature set were extracted. The implementation was done on two datasets. The accuracy results indicated no improvement in one dataset compared with the state-of-the-art works, but in the other dataset, average accuracy increased by about 2%.

The purpose of [41] was to examine the contribution of multi modalities to personality perception. The authors investigated different modalities and different fusion methods to compare the two winners of the ChaLearn 2016 competition. They implemented CNN to raw data of all the modalities for feature engineering. The performance, however, decreased compared with the first-place winner.

III. DATASET

One remaining challenge with speech-based personality perception is the limitation in the dataset, which has been discussed frequently in academic conferences for a decade [42]. Some reasons behind these limitations are: 1) most of them are not public, 2) some of them are not prosodically annotated, 3) labeling is an expensive process, 4) training annotators is a difficult trend, and 5) at least one psychologist must be employed to supervise the annotating process.

A standard dataset that has been used mostly in APP studies is described below.

A. THE PERSONALITY CORPUS

The SSPNet Speaker Personality Corpus (SPC) includes 640 speech clips in the French language. Each clip is recorded in 10 seconds (short clip) because many studies have shown that people form a first impression about a person characteristics within the first 10s of a meeting. The number of subjects was 322. 11 assessors (annotators) evaluated each clip, who did not understand French and were not influenced by linguistic cues. This evaluation was based on the BFI-10 questionnaire. The average of the 11 questionnaire scores was the final scores for each clip [8].

The final scores equal to or over 50 got *high* labels (got 1 in the target vector), and the scores below 50 got *low* tags (got 0 in the target vector). Thus, our algorithm's target was

a vector like [0, 1, 1, 0, 1] for each audio clip representing five personality traits. For instance, it means one can be at a high level of Openness, Extraversion, and Conscientiousness traits and be at a low level for the others simultaneously or any other combination of 0 and 1.

Table 1 indicates the clip's number of the SPC dataset at the high and low levels in each trait.

TABLE 1. Number of short audio clips at high and low levels in each trait at SPC dataset.

Traits	Low	High	Total sample
Neu.	337	303	640
Ext.	370	270	640
Ope.	227	413	640
Agr.	334	306	640
Con.	418	222	640

The SPC dataset is a suitable dataset for comparing our novel method to the others since several published works are based on it. Nevertheless, it contains insufficient samples (640 samples) for Deep Neural Network (DNN) training.

One approach to solving this problem involves the use of data augmentation techniques. Data augmentation is a technique to expand the size of the current training dataset artificially [43].

B. DATA AUGMENTATION

Data augmentation is a popular method in image enhancement, especially when there is not another data resource. Some transformations such as rotation, scaling, cropping, and flipping are applied to copies of original images while the label is preserved.

Regarding the study of PP, the use of data augmentation methods is not as easy as in image processing. Because the prosodic content of speech must be preserved during transformations. So, those transformations must be examined to ensure if the speaker personality differences maintained [42].

Recently, some methods have been proposed for audio data augmentation in emotional classification [44] and speech recognition [37], [45]. These data augmentation types are based on the spectrogram of the audio signals [46].

The spectrogram is a frequency-time visualization of a signal, represented in a 2-D color image in which the intensity of the colors indicates the amplitude of the respective frequencies components (see Fig.1-a) [30].

To illustrate the spectrogram, a speech signal x(n) with N samples is proposed. $\hat{x}(n)$ is the Discrete Fourier Transform (DFT) of a finite duration signal calculated by (1). The inverse of it (IDFT) is presented by (2).

$$\hat{x}(n) = \sum_{n=0}^{N-1} x(n) e^{-i\frac{2\pi k}{N}n} \quad k = 0, \ldots, N-1 \tag{1}$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} \hat{x}(k) e^{-i\frac{2\pi k}{N}n} \quad n = 0, \ldots, N-1 \tag{2}$$

x(n) is divided into consecutive frames with m samples ($m < N$). The frame overlap is m-2. Every single frame contains $\left[x[j], x[j+1], \ldots, x[j+m-1]\right]^T$ samples, where j is the sample number by which the frame starts. Placing the frames together, one can obtain $\mathbf{X}$ as:

$$\mathbf{X} = \begin{bmatrix} x\,[1] & x[2] & \ldots & x[N\text{-}m] \\ x[2] & x[3] & \ldots & x[N\text{-}m+1] \\ \vdots & \vdots & \vdots & \vdots \\ x[m\text{-}1] & x[m] & \cdots & x[N] \end{bmatrix}$$

In this paper, three audio augmentations, including time warping, time masking and frequency masking methods, are utilized as follows.

Time Warping: In this method, a random point along the time axis of the spectrogram is chosen. The spectrogram is warped to the left or right by a distance of w in the range of (w,N − w) [45], [47]. To be specified, the warping method employs per-pixel flow vectors. It is applied to the spectrogram specified by a flow field over time. It defines a pixel value in the output image corresponding to that pixel in the origin spectrogram (image). By this trend, the location of a new point does not necessarily map to an integer index. For instance, the pixel value (t,f) in the original spectrogram is (t-flow(t,f),f) in the new spectrogram. So, the pixel value is captured by calculating the bilinear interpolation around (t-flow (t,f),f). Put simply, this method squeezes and stretches the spectrogram through the time axis by interpolation techniques (Fig. 1-b) [43], [48], [49].

Time Masking: t consecutive time steps are masked, replaced by a minimum intensity, at a random point along the spectrogram's time axis in the range of $[t_0, t_0 + t)$, where t_0 is chosen from $[0, N - t)$ randomly [45], [46]. The vertical black strip with a bandwidth of t in Fig. 1-c is the visual illustration of the time masking method.

Frequency Masking: f consecutive frequency channels are masked, replaced by a minimum intensity, at a random point along the spectrogram's frequency axis in the range of $[f_0, f_0 + f)$, where f_0 is picked from $[0, \nu - f)$ and ν indicates the number of frequency channels [45], [46]. The horizontal black strip with a bandwidth of f in Fig. 1-d displays the frequency masking effect on the spectrogram.

The time masking and frequency masking methods are like the cutout data augmentation technique. According to Fig. 1, the minimum intensity means −80 dB. The sound intensity is calculated by (3).

$$\text{sound intensity (dB)} = 10 \log \left(\frac{P_2}{P_1}\right) \tag{3}$$

where P_1 and P_2 are the powers of the sound, the P_1 is the reference power, and the P_2 is the studied power. In our study, P_1 and P_2 refer to the value in matrix $\hat{\mathbf{X}}$. From Fig. 1, it is evident that the maximum intensity illustrated by 0dB. So, the P_1 is the maximum power (reference power). Therefore it can be concluded from (3) that $P_1 = 10^8 P_2$ for sound intensity −80 dB. If considering P_1 related to value 1, the P_2 is related to value 10^{-8} (near to zero).

To analyze the three data augmentation effects, the reconstructed signals of these three methods are illustrated

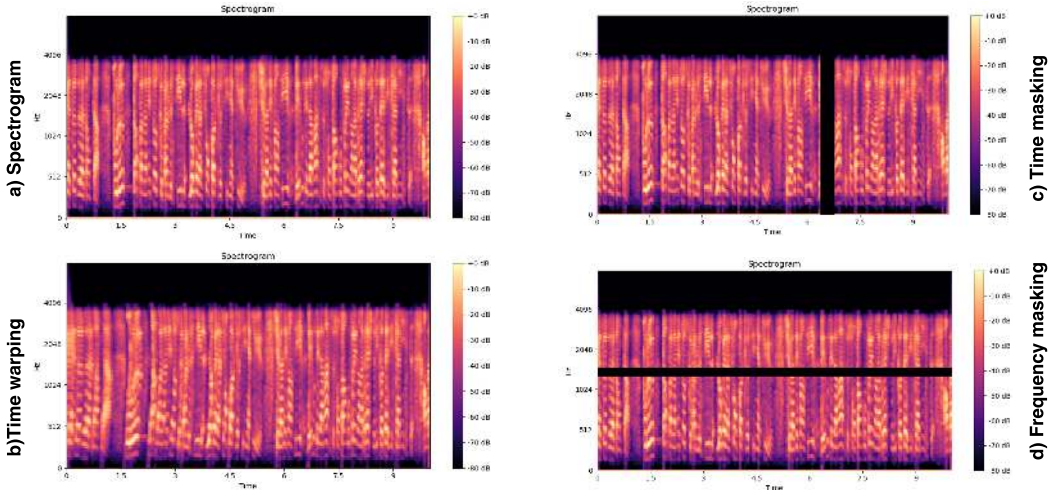

FIGURE 1. Three audio data augmentation methods. a) Displaying the original spectrogram. b) Showing the time warping method applied to the spectrogram. c) Representative time masking method applied to the spectrogram. d) Exhibition of the frequency masking method applied to the spectrogram.

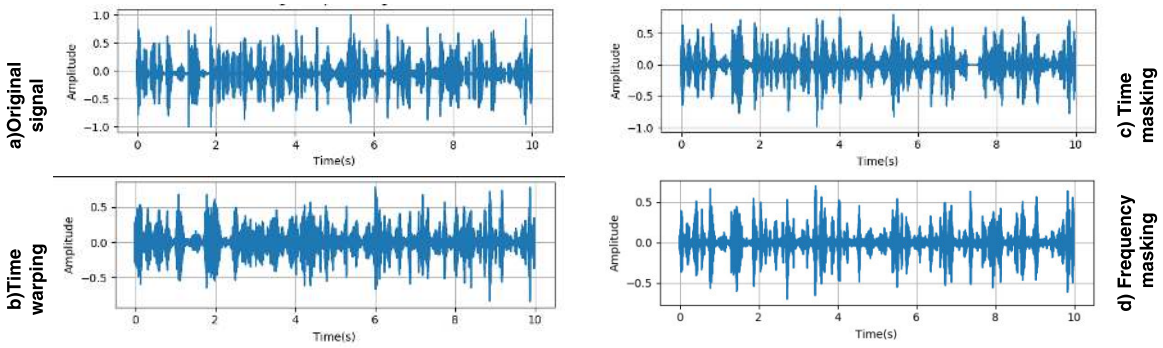

FIGURE 2. Original speech signal (a) in comparison with reconstructed speech signal affected by (b) time warping (c) time masking methods, and (d) frequency masking.

in Fig. 2. As noted, the spectrogram is calculated by applying the DFT to $\mathbf{X}$, which makes the $\hat{\mathbf{X}}$ matrix as:

$$\hat{\mathbf{X}} = \begin{bmatrix} \hat{x[1]} & \hat{x[2]} & \dots & \hat{x[N-m]} \\ \hat{x[2]} & \hat{x[3]} & \dots & \hat{x[N-m+1]} \\ \vdots & \vdots & \vdots & \vdots \\ \hat{x[m-1]} & \hat{x[m]} & \cdots & \hat{x[N]} \end{bmatrix}$$

For reconstructing a speech signal from the spectrogram, the IDFT is applied on the $\hat{\mathbf{X}}$ by (2). Note that DFT and IDFT are applied to the columns of $\mathbf{X}$ and $\hat{\mathbf{X}}$, respectively.

Fig. 2 shows an original signal compared with the reconstructed signals affected by time warping (Fig. 2-b), time masking (Fig. 2-c), and frequency masking (Fig. 2-d) methods.

Comparing Fig. 2-b with Fig. 2-a, it can be concluded that the time warping method shifts the DFT values of the X matrix in the time axis, and because of using the interpolating technique, the values around the baseline are increased.

Time Masking Analysis: By employing the time masking method, some consecutive columns of $\hat{\mathbf{X}}$ are replaced with zero. For signal reconstruction, the IDFT output of these columns is zero columns too. Therefore, it is expected that the time masking method sets the reconstructed signal's amplitude to zero in the specified time range. Fig. 2-c indicates such analysis.

Frequency Masking Analysis: Using the frequency masking method, some rows of $\hat{\mathbf{X}}$ are replaced with zero according to the channels that must be masked. Applying (2) to $\hat{\mathbf{X}}$ with some zero values, the matrix $\mathbf{Z}$ is provided as:

$$\mathbf{Z} = \begin{bmatrix} z[1] & z[2] & \dots & z[N-m] \\ z[2] & z[3] & \dots & z[N-m+1] \\ \vdots & \vdots & \vdots & \vdots \\ z[m-1] & z[m] & \cdots & z[N] \end{bmatrix}$$

where $z(n) < x(n)$ for $n = 1, \dots, N$ due to the summation with zero values. It is expected that the reconstructed signal with this method has a lower amplitude compared with the original signal (Fig. 2-d).

We also touch on some statistical analysis in the following. A comparison table of a descriptive statistic, the histogram

TABLE 2. A descriptive statistic of reconstructed signals compared with the original signal.

	Original	Time warping	Time masking	Frequency masking
Lenght	159744	159744	159744	159744
Mean	-0.036	0.018	-7.915e -5	2.506e -5
Std. Deviation	0.131	0.149	0.113	0.122
Skewness	0.068	0.019	-0.002	-0.068
Kurtosis	5.458	3.053	5.974	5.715
Minimum	-1.000	-1.000	-1.000	-1.000
Maximum	1.000	1.000	1.000	1.000
P_value		0.038	< .001	< .001

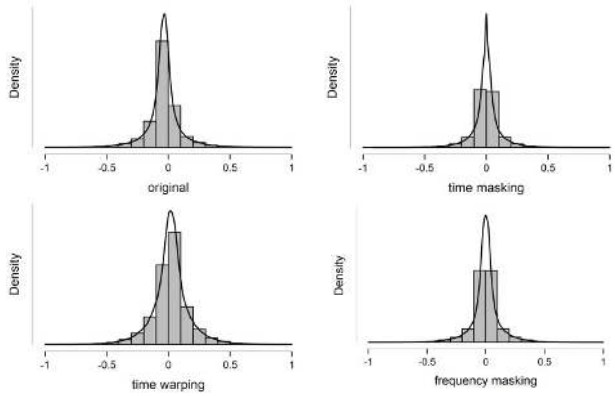

FIGURE 3. Histogram and density plot of three reconstructed signals and the original speech signal.

and density plot of those reconstructed signals compared with the original signal are reported in Table 2 and Fig. 3, respectively.

The description and analysis of Table 2 are as below.

1. Length: The length of the four signals is the same. It means the mentioned manipulations do not change the length of the signal.
2. Mean: The mean of time masking and frequency masking is near zero. It is because of the amplitude decreasing process in these two methods. For the time warping method, the mean is higher than the original. It is due to the interpolated technique that replaces the original value with the average of some other values.
3. Std. Deviation (a measure of the amount of variation in data): For the time masking and frequency masking, this measure is lower than the original and for the time warping is higher than the original. It can be seen in Fig. 3.
4. Skewness (a measure of symmetry): Although this measure for all signals is near zero, the skewness value for the original signal and time warping is positive, and the two others are negative. Due to the low value of this measure, it is not easy to show our description from Fig. 3.
5. Kurtosis (a measure of heavy-tailed/light-tailed relative to a normal distribution): As can be concluded from Table 2, all four signals have heavy-tailed than the normal distribution. The time masking is tailed than others. Also, It can be concluded from Fig. 3.
6. P_value (probability of rejecting the null hypothesis): The null hypothesis is true for the three augmented methods. In this study, the null hypothesis is that there is no statistically significant difference between the mean of the studied reconstructed signal than the original signal. The significance level is 5%.

The narrower the black band in the time masking and the frequency masking, the more similar the reconstructed signal to the original signal.

It should be noted that changes in speech signal lonely do not represent specific information about variation in personality. In other words, just statistical analysis cannot comment on the personality trait manipulation extracted from the reconstructed speech signal. In fact, accurate analysis is a complicated task to do. We need to choose transformations that maintain speaker's personalities. So, we have to be confident that such manipulations in the spectrogram do not interfere with the extracted features related to personality traits. In this regard, the best way is to classify some features extracted from the augmented signals and discuss the results afterwards to confirm our analysis in the experimental result section.

Hereafter, the datasets based on time warping, time masking, and frequency masking are called *Twarp, Tmask*, and *Fmask*.

IV. SUGGESTED STRATEGY

The flowchart of the human-computer interface is manifested in Fig. 4. The stages within the dashed line box display our work steps.

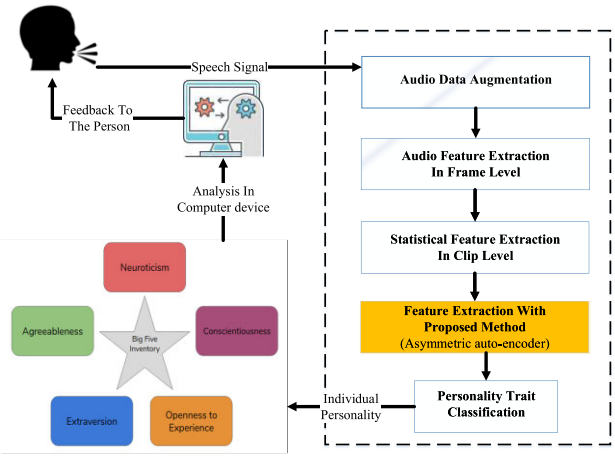

FIGURE 4. Human-computer interaction flowchart containing the framework of our work.

According to Fig. 4, the mentioned audio data augmentation methods were performed on the speech dataset first. The frame-level features were then extracted from the augmented dataset. Nevertheless, personality perception is more complex than being taken at the frame level. Therefore, statistical

properties were extracted from frame-level features of the entire a clip to utilize long-term variations. Our proposed method was applied to statistical features to extract new ones. Finally, the five personality traits were classified using the features obtained from our novel method.

In the following, each stage is described in detail.

A. FEATURE EXTRACTION IN FRAME LEVEL

Low-Level Description (LLD) features have been extracted from the Opensmile2.3 toolkit (a 60ms frame with a 20ms overlap in the time domain and a 20ms frame with a 10ms overlap in the frequency domain in every 10-second utterance).

The IS12_speaker_trait configuration file extracts 130 LLD features. These features include 65 LLD and 65 first derivatives of LLD (ΔLLD), which are described in Table 3. All the extracted features are eventually named LLD. These LLDs are a set of characteristics consisting of prosodic, cepstral, spectral, and voice quality features. As a result, 130 frame-level features were extracted [18].

B. FEATURE EXTRACTION IN CLIP LEVEL

At the clip level (audio level), 6,373 statistical parameters were extracted separately from the augmented dataset. Reference [21] describes the details of these features.

Although the clip-level and frame-level feature sets provide useful information about the speech signal, previous studies have indicated no direct relationships between speech features and personality traits. Thus, we proposed a novel feature extraction method based on deep learning to accomplish a nonlinear relationship between personality traits and speech characteristics.

C. FEATURE EXTRACTION WITH ASYMMETRIC AUTO-ENCODER

AEs are unsupervised learning algorithms to reconstruct their input as output [50]. The weight matrix of the decoder layer transposes the encoder layer weight matrix [51]. This property of the auto-encoder makes the decoder and encoder layers to be symmetric. A conventional S_{AE} with high depth depended on properties and dimensions input data encounters the problem of vanishing gradient. This problem comes up when gradient values back-propagates to the beginning of the network, in such a small extent that the network's parameter changes are negligible or completely stopped [33]. Therefore, the deep network's first layers' parameters would not be tuned well and degrade the classification results [52].

Although fine-tuning improves parameter training, it restricts the use of classifiers. In other words, a gradient-based classifier must be employed [53].

In our approach, one neuron is added to the decoder section of the Conventional AE (Con_{AE}), whose value is the personality traits label. Cleverly adding this neuron to the decoder layer causes the label to be involved in the weight training process but is removed in the testing process.

Fig. 5 is shown our proposed Asy_{AE}.

TABLE 3. The 130 LLD features, including 65 LLD and 65 Δ LLD features [18].

4 Energy Related LLD	Group
Sum of Auditory Spectrum (Loudness)	Prosodic
Sum of RASTA-Style Filtered Auditory Spectrum	Prosodic
RMS Energy, Zero-Crossing Rate	Prosodic
55 Spectral LLD	**Group**
RASTA-Style Auditory Spectrum, Bands 1–26 (0–8 kHz)	Spectral
MFCC 1–14	Cepstral
Spectral Energy 250–650 Hz, 1 k–4 kHz	Spectral
Spectral Roll Off Point 0.25, 0.50, 0.75, 0.90	Spectral
Spectral Flux, Centroid, Entropy, Slope, Harmonicity	Spectral
Spectral Psychoacoustic Sharpness	Spectral
Spectral Variance, Skewness, Kurtosis	Spectral
6 Voicing Related LLD	**Group**
F_0 (SHS & Viterbi Smoothing)	Prosodic
Probability of Voicing	Sound Quality
Log. HNR, Jitter (Local, Delta), Shimmer (Local)	Sound Quality

Mean Values
Arithmetic Mean $A^{\triangleq}, B$, Arithmetic Mean of Positive Values$^{A^{\Delta},B}$,
Root-Quadratic Mean, Flatness
Moments: Standard Deviation, Skewness, Kurtosis
Temporal Centroid $A^{\triangleq}, B$
Percentiles
Quartiles 1–3, Inter-Quartile Ranges 1–2, 2–3, 1–3,
1%-tile, 99%-tile, Range 1–99%
Extrema
Relative Position of Maximum and Minimum, Full Range (Maximum–Minimum)
Peaks and Valleys^{A}
Mean of Peak Amplitudes,
Difference of Mean of Peak Amplitudes to Arithmetic Mean,
Mean of Peak Amplitudes Relative to Arithmetic Mean,
Peak to Peak Distances: Mean and Standard Deviation,
Peak Range Relative to Arithmetic Mean
Range of Peak Amplitude Values,
Range of Valley Amplitude Values Relative to Arithmetic Mean,
Valley-Peak (Rising) Slopes: Mean and Standard Deviation,
Peak-Valley (Falling) Slopes: Mean and Standard Deviation
Up-Level Times: 25%, 50%, 75%, 90%
Rise and Curvature Time
Relative Time in which Signal is Rising,
Relative Time in which Signal has Left Curvative
Segment Lengths^{A}
Mean, Standard Deviation, Minimum, Maximum
Regression $A^{\triangleq}, B$
Linear Regression: Slope, Offset, Quadratic Error,
Quadratic Regression: Coefficients a and b, Offset c, Quadratic Error
Linear Prediction
LP Analysis Gain (Amplitude Error), LP Coefficients 1–5

^{A}Functionals applied only to energy related and spectral LLDs (group A)
^{B}Functionals applied only to voicing related LLDs (group B)
$^{\Delta}$Functionals applied only to ΔLLDs
$^{\triangleq}$Functionals **not** applied to ΔLLDs

Adding a neuron into the Con_{AE} can turn unsupervised learning into a semi-supervised one. This single neuron produces $1 \times n_1 + 1$ weights (blue lines in Fig. 5). Here, n_1 is the number of input layer neurons. The error is obtained by subtracting this neuron's output and its desired value, which back-propagates to the encoder and decoder layers.

The desired value of this single neuron is important. If 1 (high level) or -1 (low level) is chosen, the output neuron would be saturated. If 1 (high level) and 0 (low level) is selected, the zero value will deactivate the neuron. Hence, 0.5 and -0.5 are selected as the desired values for the high and low levels, respectively.

The feed-forward equations for Asy_{AE} are described as follows.

For the encoder layer:

$$\mathbf{net}^{(1)} = \mathbf{W}^{(1)}\mathbf{X} \tag{4}$$

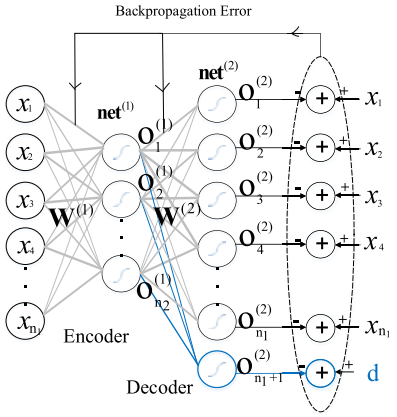

FIGURE 5. Our proposed asymmetric auto-encoder.

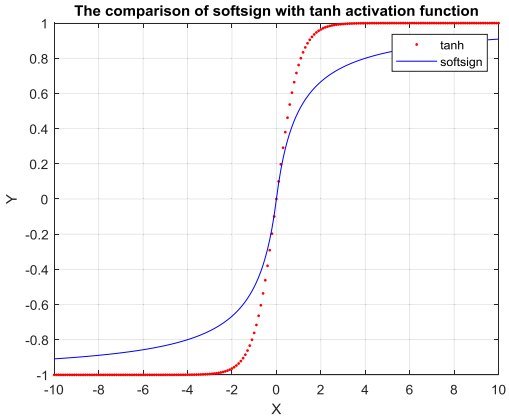

FIGURE 6. Comparison of softsign and tanh activation function diagram in terms of saturation speed, linear and nonlinear region.

$$\mathbf{O}^{(1)} = \mathrm{f}\left(\mathbf{net}^{(1)}\right) \tag{5}$$

$\mathbf{W}^{(1)}$ is the weight matrix of the encoder layer. $\mathbf{X}$ indicates the input matrix in each $\mathrm{Asy_{AE}}$, which is the feature matrix introduced in section IV-B for the first $\mathrm{Asy_{AE}}$ and the previous $\mathrm{Asy_{AE}}$ encoder output matrix for the remaining $\mathrm{Asy_{AE}}$.

In all equations, superscript 1 represents the encoder layer, and superscript 2 represents the decoder layer.

Coming to the decoder layer:

$$\mathbf{net}^{(2)} = \mathbf{W}^{(2)}\mathbf{O}^{(1)} \tag{6}$$

$$\mathbf{O}^{(2)} = \mathrm{f}\left(\mathbf{net}^{(2)}\right) \tag{7}$$

$\mathbf{O}^{(1)}$ and $\mathbf{O}^{(2)}$ are the encoder and decoder layer's output matrixes, respectively. $\mathbf{W}^{(2)}$ is the weight matrix of the decoder layer.

Since the dimensions of the weight matrix of the encoder and decoder layers are not equal ($\mathbf{W}^{(1)} \neq \left(\mathbf{W}^{(2)}\right)^{\mathrm{T}}$), the proposed auto-encoder would be considered as asymmetric.

The weight matrix of the decoder layer is as follows:

$$\mathbf{W}^{(2)} = \begin{bmatrix} w^{(2)}_{11} & w^{(2)}_{12} & \cdots & w^{(2)}_{1n_2} \\ w^{(2)}_{21} & w^{(2)}_{22} & \cdots & w^{(2)}_{2n_2} \\ \vdots & \vdots & \vdots & \vdots \\ w^{(2)}_{(n_1+1)1} & w^{(2)}_{(n_1+1)2} & \cdots & w^{(2)}_{(n_1+1)n_2} \end{bmatrix}$$

In this matrix, $\mathrm{n_1}$ and $\mathrm{n_2}$ are the number of input layer neurons and encoder layer neurons, respectively.

In (7), f demonstrates the activation function. Here, we have two points to choose neural network activation functions. 1) Preventing the activation function saturation, and 2) Including both linear and nonlinear ranges [33]. Thus, the softsign function was chosen. This function has linear, nonlinear, positive, and negative ranges larger than the tanh function, which causes later saturation than tanh [50]. Exploring more nonlinear space for feature extracting, we opted for the activation function, which provides a rather larger nonlinear range.

Fig. 6 compares the softsign (the blue curve) and tanh (the red dotted curve) functions.

The softsign function is defined by (8).

$$\mathrm{f(x)} = \frac{\mathrm{x}}{1 + |\mathrm{x}|} \tag{8}$$

The error back-propagation equation is:

$$\mathbf{e}_\mathrm{t} = \mathbf{d}_\mathrm{t} - \mathbf{o}^{(2)}_\mathrm{t} \tag{9}$$

$\mathbf{e}_\mathrm{t}$ is the $\mathrm{Asy_{AE}}$ error vector and $\mathbf{d}_\mathrm{t}$ is the desired output vector at time t.

The vector $\mathbf{d}_\mathrm{t}$ belongs to the $\mathbf{D}$ matrix.

$$\mathbf{D} = \begin{bmatrix} \mathrm{x}_{11} & \mathrm{x}_{12} & \cdots & \mathrm{x}_{1n_0}\ell \\ \mathrm{x}_{21} & \mathrm{x}_{22} & \cdots & \mathrm{x}_{2n_0}\ell \\ \vdots & \vdots & \vdots & \vdots \\ \mathrm{x}_{m1} & \mathrm{x}_{m2} & \cdots & \mathrm{x}_{mn_0}\ell \end{bmatrix}$$

The $\mathbf{D}$ is the desired output matrix of $\mathrm{Asy_{AE}}$, which is a combination of labels and $\mathrm{Asy_{AE}}$ input. The components of the matrix $\mathbf{D}$ are $\mathrm{x_{ij}}$ and ℓ. For the first $\mathrm{Asy_{AE}}$, $\mathrm{x_{ij}}$ is the feature matrix elements, and for the other $\mathrm{Asy_{AE}}$, are the previous $\mathrm{Asy_{AE}}$ encoder layer's output. ℓ is the label of personality trait, in which the values are -0.5 (low level) or 0.5 (high level).

For the first $\mathrm{Asy_{AE}}$ in the stacked auto-encoder, the dimensions of the matrix $\mathbf{D}$ would be $\mathrm{m} \times (\mathrm{n_0} + 1)$. Here, m is the number of samples and $\mathrm{n_0}$ is the number of features, which is 6373.

For the second $\mathrm{Asy_{AE}}$, and the rest of them, the dimensions of the matrix $\mathbf{D}$ are equal to $\mathrm{m}\times(\mathrm{n_i} + 1)$. $\mathrm{n_i}$ is the number of neurons in the encoder layer in the $\mathrm{Asy_{AEi-1}}$.

Assuming the five personality traits are independent, five separate neural networks are trained to classify five personality traits.

Moreover, the model error loss function is calculated using the logcosh function [54]. The logcosh works like the mean squared error (MSE) but not affected by the incorrect prediction. The formula is described in (10) as

follows:

$$E = \frac{1}{k}\sum_{i=1}^{k} \log\left(\cosh\left(\mathbf{e}_t\right)\right) \tag{10}$$

where k is the number of neurons in the decoder's output layer.

Fig. 7 illustrates the complete diagram of our feature extraction method and classification stage. SA_{AE}, in which each weight is pre-trained by Asy_{AE}, extracts new features from clip-level features. The features obtained by the proposed method are then classified by SVM to determine if each personality trait level is low or high.

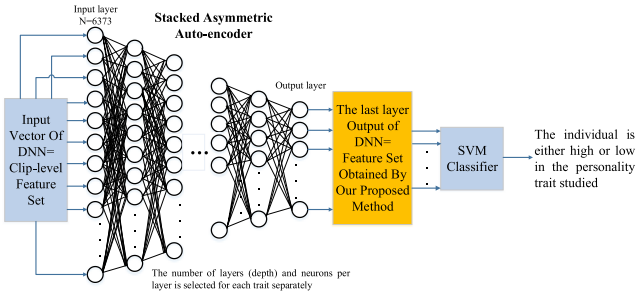

FIGURE 7. Schematic illustration of SA_{AE} for automatic feature extraction. The depth and neurons per layer are selected during the training process. The SVM classifier was utilized to recognize the high or low level of the personality trait studied.

Using semi-supervised training in every Asy_{AE} can manage the vanishing gradient problem, which occurs inevitably for high-depth DNNs. So the depth of DNN can increase as much as needed. For more emphasis on this ability, we used the SVM classifier to indicate that DNN parameters are appropriately adjusted, and the fine-tuning process is not needed by gradient back-propagation. It is important to note that our method did not solve the vanishing gradient problem, but the novel semi-supervised training method eliminated the need for fine-tuning. Hence, the vanishing gradient problem will not occur, and the network depth can increase as much as needed.

The parameter training process is accomplished as soon as the epoch (the number of times the parameters are updated) reaches its maximum [40], or the error rate stopped improving [55]. As discussed in the introduction, the purpose of Asy_{AE} is to extract those features that provide adequate separation between low and high levels of the studied personality trait, but adding a neuron alone does not meet this goal. Thus, a further stop criterion is needed to find the epoch where the trained weights are the maximum separation between the two levels. In other words, although personality traits influence the features extracted by the Asy_{AE}, this alone does not guarantee that the obtained features will be well separable. Therefore, J variation is also examined.

J is a ratio of between-class scattering to within-class scattering, which is a scalar value. The higher the J value, the greater the separation.

The value of J is calculated as follows:

$$J = \frac{\det(\mathbf{S_B})}{\det(\mathbf{S_W})} \tag{11}$$

$$\mathbf{S_w} = \sum_{i=1}^{c}\sum_{x\in c_i}(\mathbf{X}-\boldsymbol{\mu_i})(\mathbf{X}-\boldsymbol{\mu_i})^T \tag{12}$$

$$\mathbf{S_B} = \sum_{i=1}^{c} n_i(\boldsymbol{\mu_i}-\boldsymbol{\mu})(\boldsymbol{\mu_i}-\boldsymbol{\mu})^T \tag{13}$$

where S_W is a within-class scattering matrix, calculated by (12) that should be minimized, whereas S_B is a between-class scattering matrix, calculated by (13) that should be maximized [56]. $\mathbf{X}$ is the encoder output matrix and c is the number of classes.

It should be noted that both the S_W and S_B matrixes are defined for the encoder layer.

In (11), **det** represents the determinant of the matrix. There are binary classes (low and high). n_i and μ_i in (12) and (13) are the numbers of samples in each class and each class's mean matrix, respectively, for the high class, $i = 2$ and for the low class, $i = 1$. μ is the mean matrix for all samples.

Relying on the fact that different personality traits have different effects on speech characteristics [57], [12], [54], using the same DNN structure for all traits to extract features is not recommended.

Hence, the network's depth was determined by classifying the output of each Asy_{AE} encoder layer by the SVM with radial basis function kernel. The Asy_{AE} with higher classification results is considered as the output layer of the S_{AE}.

V. EXPERIMENTAL RESULTS

In this section, the results of the two simulations are discussed.

Firstly, we evaluated the augmented SPC dataset to prove the effectiveness of augmented data in the APP. Then, the stacked AsyAE was applied to the augmented dataset. Finally, the proficiency of our feature extraction method was compared to that of other methods and published works.

A. DATA AUGMENTATION RESULTS

As mentioned in section III-B, the three audio augmentation methods must be evaluated in the APP field.

In this context, the method proposed in [12], which is the first usage of SPC dataset in APP, was implemented with these three augmented datasets.

We implemented the article [12] once with the original SPC dataset and compared the accuracy and UA recall results to those of the published values. The process of [12] is implemented on *Twarp*, *Tmask*, and *Fmask* datasets.

Four statistical features, calculated from some acoustic features, were extracted from the original and augmented SPC datasets. Afterwards, the features were classified by logistic regression. These features are the maximum, minimum, mean, and entropy of pitch, first and second formant, energy and the length of voiced, and unvoiced segments.

TABLE 4. Classification results of the SPC dataset to illustrate the impact of three audio augmentation methods compared with the original dataset in terms of UA recall% (accuracy%).

Traits	Dataset				
	Original		*Twarp*	*Tmask*	*Fmask*
	[12]	Our			
Neu.	N/A(66.1)	55.9(61.7)	54.6(51.3)	52.7(53.4)	56.8(57.5)
Ext.	N/A(71.4)	62.2 (70.2)	53.3(71.3)	60.1(66.2)	61.3(62.6)
Ope.	N/A(58.6)	58.8(55.3)	53.1(56.5)	44.8(57.9)	51.1(68.5)
Agr.	N/A(58.8)	60.5(53.8)	50.3(62.6)	48.7(78.6)	50.1(73.1)
Con.	N/A(72.5)	64.4(69.6)	60.1(67.6)	56.8(62.1)	51.1(60.7)

Table 4 indicates the regression results of the augmented dataset for each trait at high and low levels. The grey column specifies the published results in [12].

The *Original-Our* column shows our implementation results based on the original SPC dataset. N/A represents the considered value was not available. Obviously, the results we obtained were close to those announced in [12]. As a result, we analyzed the results of the other three datasets confidently.

Comparing the outcomes of *Twarp*, *Tmask*, and *Fmask* with *Original-our*, it was observed that although the UA results were lower than those in the *Original-our*, the three traits (Openness, Agreeableness, and Conscientiousness) had a higher UA recall in the *Twarp* dataset. In contrast, Neuroticism and Extraversion had the upper UA in *Fmask* dataset. Henceforth, in this article, our proposed method was examined based on *Fmask* for Neuroticism and Extraversion traits and based on *Twarp* for the three others.

It should be emphasized that with the augmentation method, the number of audio signals can increase to the desired number as many times as needed. However, in the first simulation, the number of augmented signals in each dataset was kept as the number of the *Original* dataset for a more accurate comparison. For the second simulation, the number of clips increased up to 64,000 in each dataset.

B. ASSESSMENT OF THE FEATURE EXTRACTED BY Asy_{AE}

For comparing the proposed method to the DLMs, two simulations were done with five different models for five traits. There are two major reasons for separating model for each trait as follows:

1. As discussed in the introduction, there is no evidence claiming that a specific feature set could work well for all five personality traits.
2. As Table 4 indicates, the augmented dataset was different for the five traits, which means we did not have one dataset for the five traits.

Hence we implemented five independent classification models for conventional stacked auto-encoder (S_{AE}), CNN, and our proposed method (SA_{AE}). In this way, keeping networks' basic conditions the same, a fairer comparison shall be made.

In the first simulation, the same structure (number of hidden layers and the number of neurons in each layer) was implemented for S_{AE} and SA_{AE}. The input matrix contained 6,373 statistical features for both methods. Asy_{AE} was used to train weights of $SA_{AE,}$ and Con_{AE} was utilized for the S_{AE}. Although the two networks had the same structures, the same learning rate could not be used because of their different training weights process. Then, the learning rate of the two networks was adjusted separately. It was observed that if the learning rates were considered the same, one of the networks would be overfitted.

A batch normalization technique was used to normalize the input layer in both networks [58], and a dropout method was used to prevent overfitting [59].

The same training/development/test set was employed for both networks as well. The initial weights of S_{AE} and SA_{AE} were considered the same.

Fig. 8 indicates the trend of loss error of the Asy_{AE} compared with Con_{AE}. *Tr_Loss Asy* and *Tr_Loss* represent the training dataset's loss error in the Asy_{AE} and the Con_{AE} method, respectively. Also, *Val_Loss Asy* and *Val_Loss* represent the loss error of the development dataset of the Asy_{AE} and the Con_{AE} method, respectively.

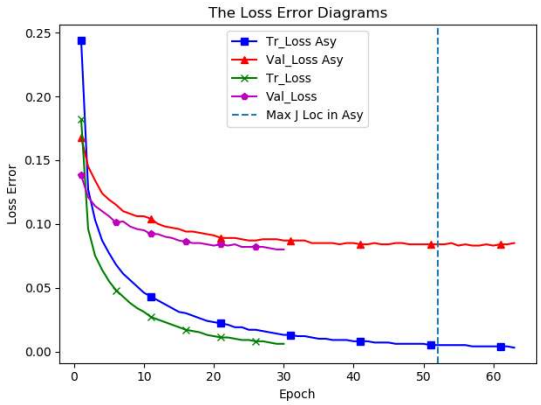

FIGURE 8. Training and development loss error diagrams of Asy_{AE} vs. Con_{AE} for the first encoder layer in the Conscientiousness trait.

As the ConAE was symmetric, which reduced the computational complexity, the maximum epoch was 30, but for the AsyAE, it was set to 500. Adding a neuron was caused by adding the labeling error to the reconstruction error increased the total error. Consequently, more epochs were considered for the training parameters of AsyAE. This was the reason why stopping the loss error diagrams of ConAE was earlier than that of AsyAE.

The convergence speeds of the two methods' error diagrams were not comparable because their learning rates were different.

For note, the objective was to achieve greater resolution between the low and high levels to classify five personality traits. Due to the offline training, the convergence speed did not affect choosing a better method.

The vertical dashed line in Fig. 8, which is called *Max J loc in Asy*, indicates the epoch at which the maximum J occurred. This criterion was considered after the first ten epochs to ensure the network is partially trained. This was because the initial weights may be such that they maximize the J value first, but this value is not valid if the network is not trained.

The comparison of J variation in Con_{AE} and Asy_{AE} is illustrated in Fig. 9. It shows that Asy_{AE} was substantially better than Con_{AE} to create binary class separately because the *J in Asy_AE* was higher than *J in AE*.

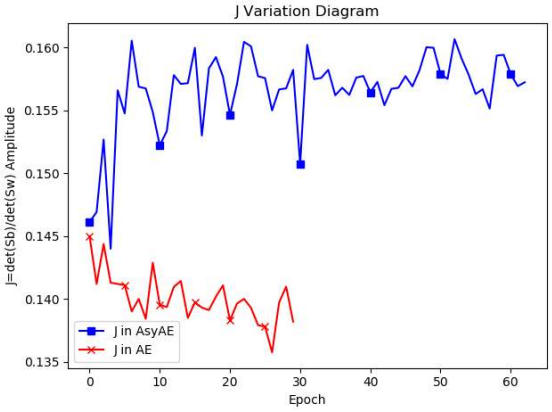

FIGURE 9. J variation diagrams of Asy_{AE} vs Con_{AE} for the first encoder layer in the Conscientiousness trait.

The diagrams in Fig. 8 and 9 are related to these two networks' first hidden layers. The description of the other layers is the same.

In the second simulation, we employed 6,373 statistical features as an input of the one dimensional CNN for feature extraction with a fully connected layer at the end [60]. This approach is similar to the other works using DLMs, described in the literature. The batch normalization and dropout techniques were used for each convolution layer. The same training/development/test set as the first simulation was used. We considered a different kernel size and stride size for the convolution layers of five CNN models.

The loss error diagrams of CNN are shown in Fig. 10. In this figure, *Tr_Loss CNN* and *Val_Loss CNN* represent the training and development dataset's loss error in the CNN, respectively.

Due to the different structure of CNN and our proposed method, it was not possible to compare the loss error values.

The classification results of the two simulations are reported in Table 5.

Table 5 compares SA_{AE} with S_{AE} and CNN in terms of UA recall and accuracy. Column N denotes the number of neurons in the hidden layer by which the best classification results in that layer occurred. Its value indicates the depth of the neural network designed for each personality trait and the degree of nonlinearity of the extracted features.

In the Agreeableness trait, the depth of the S_{AE} and SA_{AE} networks were the same. This means that for this personality trait, both networks achieved features with the same degree of

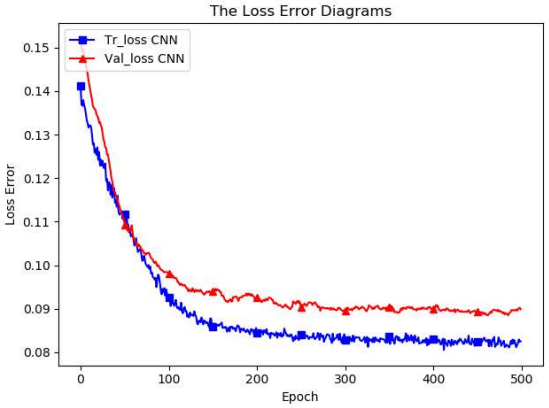

FIGURE 10. Training and development loss error diagrams of CNN for the first convolution layer in the Conscientiousness trait.

TABLE 5. Comparison results of the SA_{AE} with the S_{AE} and CNN in terms of UA recall% (Accuracy %).

Traits	Methods				
	S_{AE}	N	SA_{AE}	N	CNN
Neu.	69.2 (67.3)	250	77.1 (76.9)	100	67.5 (67.7)
Ext.	65.1 (58.3)	30	76.6 (72.9)	10	63.4 (60.1)
Ope.	69.8 (66.6)	100	81.2 (70.4)	30	60.6 (74.8)
Agr.	60.2 (56.2)	80	80.7 (68.7)	80	65.7 (79.5)
Con.	67.8 (59.3)	250	78.5 (69.5)	30	65.1 (69.2)

nonlinearity, but due to the semi-supervised training of SA_{AE}, the classification results improved compared with S_{AE}. For the other four traits, the depth of SA_{AE} was more than that of S_{AE}, which means SA_{AE} explored more than S_{AE} in feature space. On the other hand, this indicates that Neuroticism, Extraversion, Openness, and Conscientiousness need higher-orders of nonlinear properties to be well classified.

As mentioned, the criteria for evaluating APP methods based on speech signals in various articles were UA recall and accuracy. From this perspective, all of our results are presented based on these two criteria, and 10-fold CV evaluated outcomes. For two reasons, we chose ten folds. In most of the personality perception studies on the SPC dataset, ten folds were employed, which meets the number of the samples. Therefore, we also went for ten folds herein accordingly.

From Table 5, it can be concluded that the proposed method could simultaneously enhance both accuracy and UA recall in comparison with the conventional stacked auto-encoder.

As demonstrated in Table 5, our novel method's UA results were more efficient than those of the CNN in all the five traits, although the accuracies of Openness and Agreeableness traits in CNN were more than SA_{AE}. One of the major reasons is that CNN uses the pooling layer for dimensionality reduction and downsampling features [61]. The pooling layer depends on its size and type and ignores some beneficial information. Meanwhile, the SA_{AE} method reduces dimensionality by controlling features quality through a smart stop criterion and

considers classification results in each layer. The other reason is that the CNN feature extraction process (in convolution layers) is unsupervised [62]. The inefficiency of unsupervised features was described in detail in the introduction section. Also, from a practical point of view, the vanishing gradient problem affects CNN parameters trained by the fine-tuning process. As a result, the deep method's first layers cannot be tuned well. The first layer tuning is important because it is the feature extractor of hand-crafted features and the only layer related to it. If this layer's fine-tuning is not satisfied, the other layers are affected by the classification results. Meanwhile, the SA_{AE} uses the semi-supervised method to find appropriate features related to personality. On the other hand, according to the kernel size, feature extraction in the convolutional layer is local. For example, the first convolution layer ignores the relationship between the first hand-crafted feature and the last one, while in SA_{AE} and S_{AE} methods, it is globally through the fully connected layer in each layer. The last consideration of CNN is the stride size which determines how many features the sliding window skips.

The above issues are the properties of convolutional neural networks, yet in the subject of the current study, are kinds of weakness. Because of the limitations in the dataset, the outstanding authors used hand-crafted features as the input of deep methods described in the introduction and related works sections.

It should be noted that it is not possible to compare the depth of CNN to that of SA_{AE} because the feature extraction process is different.

Now we want to have a comparison with the methods suggested in other articles.

Table 6 compares the UA and accuracy results obtained from the proposed method and previous studies applied to the SPC dataset. As demonstrated, our proposed method was simple, but accomplished UA recalls comparable to other works in all the five traits. It can also be deduced from Table 6 that previous methods were not effective in classifying all the five personality traits simultaneously. For example, for the method of [14], the UA and accuracy outcomes for Extraversion and Conscientiousness are significant. However, only Conscientiousness UA enhanced compared with previous studies.

During a decade of intensive studies on APP, the classification UA (accuracy) results has not exceeded 70.8% (69.2%) for Neuroticism, 75.5% (76.3%) for the Extraversion, 73.4% (74.7%) for Openness, 64.9% (65.3%) for Agreeableness, and 75.7% (75.6%) for Conscientiousness. These results indicate the complexity of the feature extraction process for APP is acceptable despite the improvements in machine learning algorithms. It is worth mentioning that the above percentages of accuracy and UA results of a trait were not achieved by one method but in different studies and different years.

As stated in the introduction, all the five personality traits cannot be identified by one model only. In other words, if classification results obtained from one specific set of the feature were noteworthy, this set of feature would not necessarily pay off for the other traits. The advantage of our method, which results in such great success, is the fact that it automatically extracts each personality trait's appropriate feature set considering different network depths and semi-supervised training.

TABLE 6. Comparison Results Of our proposed method with other works in the SPC dataset in terms of UA recall% (Accuracy %).

Methods	Traits				
	Neu.	Ext.	Ope.	Agr.	Con.
Mohammadi *et al.* 2010 [38]	N/A (63)	N/A (76.3)	N/A (57.9)	N/A (63)	N/A (72)
Mohammadi *et al.* 2012 [13]	N/A (65.9)	N/A (73.5)	N/A (60.1)	N/A (63.1)	N/A (71.3)
Chastagnol *et al.* 2012 [21]	58 (N/A)	75.5 (N/A)	73.4 (N/A)	65 (N/A)	62.2 (N/A)
Mohammadi *et al.* 2015 [12]	N/A (66.1)	N/A (71.4)	N/A (58.6)	N/A (58.8)	N/A (72.5)
Solera-Urena *et al.* 2017 [14]	65.1 (64.7)	75 (75.1)	59.1 (58.2)	60.3 (60.2)	75.7 (75.6)
Carbonneau *et al.* 2017 [63]	70.8 (N/A)	75.2 (N/A)	56.3 (N/A)	64.9 (N/A)	63.8 (N/A)
Zhen-Tao Liu *et al.* 2020 [9]	N/A (69.2)	N/A (76.3)	N/A (74.7)	N/A (65.3)	N/A (73.3)
Our proposed method	77.1 (76.9)	76.6 (72.9)	81.2 (70.4)	80.7 (68.7)	78.5 (69.5)

Although using the Asy_{AE} method increased UA results for all the traits, the accuracy of Extraversion and Conscientiousness decreased by 3.4% and 4.3%, respectively, compared with [9]. Therefore, it should be noted that besides their advantages, DNNs may have some drawbacks.

The neural network results, including DNNs, depend on network structure and hyper-parameters tuning. Although various methods have been proposed to obtain the optimal neural network structure, no operational methods have been proposed hitherto. Therefore, we used the grid search method, which could be the reason behind the lower accuracy of the two traits.

VI. CONCLUSION

In this paper, an important challenge in the APP was inspected. A novel asymmetric auto-encoder method was then proposed to solve this challenge. The novel asymmetric auto-encoder method, which trains each hidden layer parameters in a semi-supervised manner, was proposed to solve this challenge.

The results indicated that adding one neuron to a conventional auto-encoder has several advantages. The main contribution would be the improved classification UA results of all the five personality traits simultaneously (and an improvement in three accuracy results). Using semi-supervised training in every DNN layer, the depth of DNN could increase as much as needed. Particularly for the Extraversion trait, it allowed DNN to acquire high levels of nonlinear features, which improves classification results, in turn.

The other advantages are: 1) extracting appropriate feature set automatically for each personality traits individually in order to train five DNNs with different structures (different

depth and neuron per layer), 2) improving the training process of DNN parameters in order to do semi-supervised training per layer (the AsyAE ability), 3) reducing dimensionality, 4) finding saddle point of compressing and extracting a high distinction feature set by using a smart stop criterion to classify the low and high levels in all the five traits, 5) ability to use every machine learning classifiers except gradient base in order to the precise weight adjustment within each Asy_{AE}, 6) offering a novel and efficient automatic feature extraction method to classify the well-known big five personality traits.

Finally, comparisons and analyses indicated the promising improvement of the study in terms of UA recall compared with the unsatisfactory results of other works.

As discussed, the neural network structure has a direct effect on classification results. This structure is usually obtained by trial and error, which is time-consuming and not necessarily the best structure. As a result, an automatic method was suggested to find the best structure in future works to improve the accuracy of Neuroticism and Extraversion.

REFERENCES

[1] Y. Mehta, N. Majumder, A. Gelbukh, and E. Cambria, "Recent trends in deep learning based personality detection," *Artif. Intell. Rev.*, vol. 7, pp. 1–27, Oct. 2019.

[2] M. P. Aylett, Y. Vazquez-Alvarez, and S. Butkute, "Creating robot personality: Effects of mixing speech and semantic free utterances," in *Proc. Companion ACM/IEEE Int. Conf. Hum.-Robot Interact.*, Mar. 2020, pp. 110–112.

[3] M. Braun and F. Alt, "Identifying personality dimensions for characters of digital agents," in *Character Computing*. Cham, Switzerland: Springer, 2020, pp. 123–137.

[4] R. Zhou, Y. Ou, W. Tang, Q. Wang, and B. Yu, "An emergency evacuation behavior simulation method combines personality traits and emotion contagion," *IEEE Access*, vol. 8, pp. 66693–66706, 2020.

[5] H. Zhao and X. Zhang, "A mobile security-related behavior prevention model based on speech personality traits," in *Proc. Trustcom/BigDataSE/ISPA*, Feb. 2016, pp. 1803–1810.

[6] X. Zhang and H. Zhao, "Cold-start recommendation based on speech personality traits," *J. Comput. Theor. Nanosci.*, vol. 14, no. 3, pp. 1314–1323, Mar. 2017.

[7] E. Diener and R. E. Lucas, "Personality traits," in *General Psychology: Required Reading* (Noba Textbook Series: Psychology). Champaign, IL, USA, 2019, p. 278.

[8] G. An and R. Levitan, "Lexical and acoustic deep learning model for personality recognition," in *Proc. Interspeech*, 2018, pp. 1761–1765.

[9] Z.-T. Liu, A. Rehman, M. Wu, W. Cao, and M. Hao, "Speech personality recognition based on annotation classification using log-likelihood distance and extraction of essential audio features," *IEEE Trans. Multimedia*, early access, Sep. 18, 2020, doi: 10.1109/TMM.2020.3025108.

[10] F. Mairesse and M. Walker, "Words mark the nerds: Computational models of personality recognition through language," in *Proc. Annu. Meeting Cognit. Sci. Soc.*, 2006, vol. 28, no. 28, pp. 1–10.

[11] F. Valente, S. Kim, and P. Motlicek, "Annotation and recognition of personality traits in spoken conversations from the AMI meetings corpus," in *Proc. INTERSPEECH*, 2012, pp. 1183–1186, 2012.

[12] G. Mohammadi and A. Vinciarelli, "Automatic personality perception: Prediction of trait attribution based on prosodic features extended abstract," in *Proc. Int. Conf. Affect. Comput. Intell. Interact. (ACII)*, 2015, pp. 484–490.

[13] G. Mohammadi and A. Vinciarelli, "Automatic personality perception: Prediction of trait attribution based on prosodic features," *IEEE Trans. Affect. Comput.*, vol. 3, no. 3, pp. 273–284, Jul. 2012.

[14] R. Solera-Ureña, H. Moniz, F. Batista, V. Cabarrão, A. Pompili, R. F. Astudillo, J. Campos, A. Paiva, and I. Trancoso, "A semi-supervised learning approach for acoustic-prosodic personality perception in under-resourced domains," in *Proc. Interspeech*, Aug. 2017, pp. 929–933.

[15] A. Guidi, C. Gentili, E. P. Scilingo, and N. Vanello, "Analysis of speech features and personality traits," *Biomed. Signal Process. Control*, vol. 51, pp. 1–7, May 2019.

[16] G. An, S. I. Levitan, R. Levitan, A. Rosenberg, M. Levine, and J. Hirschberg, "Automatically classifying self-rated personality scores from speech," in *Proc. Interspeech* 2016, pp. 1412–1416.

[17] C. Fayet, A. Delhay, D. Lolive, and P.-F. Marteau, "Big five vs. prosodic features as cues to detect abnormality in SSPNET-personality corpus," in *Proc. Interspeech*, 2017, pp. 3281–3285.

[18] B. Schuller, F. Weninger, Y. Zhang, F. Ringeval, A. Batliner, S. Steidl, F. Eyben, E. Marchi, A. Vinciarelli, K. Scherer, M. Chetouani, and M. Mortillaro, "Affective and behavioural computing: Lessons learnt from the first computational paralinguistics challenge," *Comput. Speech Lang.*, vol. 53, pp. 156–180, Jan. 2019.

[19] J. C. Silveira Jacques Junior, Y. Gucluturk, M. Perez, U. Guclu, C. Andujar, X. Baro, H. J. Escalante, I. Guyon, M. A. J. Van Gerven, R. Van Lier, and S. Escalera, "First impressions: A survey on vision-based apparent personality trait analysis," *IEEE Trans. Affect. Comput.*, early access, Jul. 23, 2019, doi: 10.1109/TAFFC.2019.2930058.

[20] A. Salimi, M. Ziaii, A. Amiri, M. Hosseinjani Zadeh, S. Karimpouli, and M. Moradkhani, "Using a feature subset selection method and support vector machine to address curse of dimensionality and redundancy in hyperion hyperspectral data classification," *Egyptian J. Remote Sens. Space Sci.*, vol. 21, no. 1, pp. 27–36, Apr. 2018.

[21] C. Chastagnol and L. Devillers, "Personality traits detection using a parallelized modified SFFS algorithm," *Computer*, vol. 15, p. 16, Sep. 2012.

[22] J. Pohjalainen, O. Räsänen, and S. Kadioglu, "Feature selection methods and their combinations in high-dimensional classification of speaker likability, intelligibility and personality traits," *Comput. Speech Lang.*, vol. 29, no. 1, pp. 145–171, Jan. 2015.

[23] J. Pohjalainen, S. Kadioglu, and O. Räsänen, "Feature selection for speaker traits," in *Proc. Interspeech*, 2012, pp. 270–273, 2012.

[24] H. Xianyu, M. Xu, Z. Wu, and L. Cai, "Heterogeneity-entropy based unsupervised feature learning for personality prediction with cross-media data," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2016, pp. 1–6.

[25] S. Jothilakshmi, J. Sangeetha, and R. Brindha, "Speech based automatic personality perception using spectral features," *Int. J. Speech Technol.*, vol. 20, no. 1, pp. 43–50, Mar. 2017.

[26] M. Zhu, X. Xie, L. Zhang, and J. Wang, "Automatic personality perception from speech in mandarin," in *Proc. 11th Int. Symp. Chin. Spoken Lang. Process. (ISCSLP)*, Nov. 2018, pp. 309–313.

[27] C.-L. Zhang, H. Zhang, X.-S. Wei, and J. Wu, "Deep bimodal regression for apparent personality analysis," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 311–324.

[28] V. Ponce-López, "Chalearn lap 2016: First round challenge on first impressions-dataset and results," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 400–418.

[29] H. Sadr, M. M. Pedram, and M. Teshnehlab, "Multi-view deep network: A deep model based on learning features from heterogeneous neural networks for sentiment analysis," *IEEE Access*, vol. 8, pp. 86984–86997, 2020.

[30] S. Prasomphan, "Improvement of speech emotion recognition with neural network classifier by using speech spectrogram," in *Proc. Int. Conf. Syst., Signals Image Process. (IWSSIP)*, 2015, pp. 73–76.

[31] H. Fu, P. Lei, H. Tao, L. Zhao, and J. Yang, "Improved semi-supervised autoencoder for deception detection," *PLoS ONE*, vol. 14, no. 10, Oct. 2019, Art. no. e0223361.

[32] V. Vestman, D. Gowda, M. Sahidullah, P. Alku, and T. Kinnunen, "Speaker recognition from whispered speech: A tutorial survey and an application of time-varying linear prediction," *Speech Commun.*, vol. 99, pp. 62–79, May 2018.

[33] H. H. Tan and K. H. Lim, "Vanishing gradient mitigation with deep learning neural network optimization," in *Proc. 7th Int. Conf. Smart Comput. Commun. (ICSCC)*, Jun. 2019, pp. 1–4.

[34] M.-H. Su, Y.-T. Zheng, and C.-H. Wu, "Interlocutor personality perception based on BFI profiles and coupled HMMs in a dyadic conversation," in *Proc. 9th Int. Symp. Chin. Spoken Lang. Process.*, Sep. 2014, pp. 178–182.

[35] C.-J. Liu, C.-H. Wu, and Y.-H. Chiu, "BFI-based speaker personality perception using acoustic-prosodic features," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf.*, Oct. 2013, pp. 1–6.

[36] G. An and R. Levitan, "Comparing approaches for mitigating intergroup variability in personality recognition," 2018, *arXiv:1802.01405*. [Online]. Available: http://arxiv.org/abs/1802.01405

[37] G. Mohammadi, A. Origlia, M. Filippone, and A. Vinciarelli, "From speech to personality: Mapping voice quality and intonation into personality differences," in *Proc. 20th ACM Int. Conf. Multimedia*, 2012, pp. 789–792.
[38] G. Mohammadi, A. Vinciarelli, and M. Mortillaro, "The voice of personality: Mapping nonverbal vocal behavior into trait attributions," in *Proc. 2nd Int. workshop Social signal Process.*, 2010, pp. 17–20.
[39] G. Mohammadi and A. Vinciarelli, "Humans as feature extractors: Combining prosody and personality perception for improved speaking style recognition," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 2011, pp. 363–366.
[40] U. Gáçlä, M. A. van Gerven, and R. van Lier, "Deep impression: Audiovisual deep residual networks for multimodal apparent personality trait recognition," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 349–358.
[41] O. Kampman, E. J. Barezi, D. Bertero, and P. Fung, "Investigating audio, visual, and text fusion methods for end-to-end automatic personality prediction," 2018, *arXiv:1805.00705*. [Online]. Available: http://arxiv.org/abs/1805.00705
[42] A. Rosenberg, "Speech, prosody, and machines: Nine challenges for prosody research," in *Proc. Speech Prosody*, 2018, pp. 784–793.
[43] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, p. 60, Dec. 2019.
[44] D. Mungra, A. Agrawal, P. Sharma, S. Tanwar, and M. S. Obaidat, "PRATIT: A CNN-based emotion recognition system using histogram equalization and data augmentation," *Multimedia Tools Appl.*, vol. 79, nos. 3–4, pp. 2285–2307, Jan. 2020.
[45] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A simple data augmentation method for automatic speech recognition," 2019, *arXiv:1904.08779*. [Online]. Available: http://arxiv.org/abs/1904.08779
[46] L. Nanni, G. Maguolo, and M. Paci, "Data augmentation approaches for improving animal audio classification," *Ecol. Informat.*, vol. 57, May 2020, Art. no. 101084.
[47] G. Maguolo, M. Paci, L. Nanni, and L. Bonan, "Audiogmenter: A MATLAB toolbox for audio data augmentation," 2019, *arXiv:1912.05472*. [Online]. Available: http://arxiv.org/abs/1912.05472
[48] T. Leimkähler, "Artificial intelligence for efficient image-based view synthesis," Ph.D. dissertation, Max Planck Inst. Inform., Berlin, Germany, 2019, doi: 10.22028/D291-28379.
[49] R. Vergne, P. Barla, G.-P. Bonneau, and R. W. Fleming, "Flow-guided warping for image-based shape manipulation," *ACM Trans. Graph.*, vol. 35, no. 4, pp. 1–12, Jul. 2016.
[50] Q. V. Le, "A tutorial on deep learning Part 2: Autoencoders, convolutional neural networks and recurrent neural networks," *Google Brain*, vol. 20, pp. 1–20, Oct. 2015.
[51] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Netw.*, vol. 61, pp. 85–117, Jan. 2015.
[52] X. Zhuang, L. C. Nguyen, H. Nguyen-Xuan, N. Alajlan, and T. Rabczuk, "Efficient deep learning for gradient-enhanced stress dependent damage model," *Appl. Sci.*, vol. 10, no. 7, p. 2556, Apr. 2020.
[53] P. Sibi, S. A. Jones, and P. Siddarth, "Analysis of different activation functions using back propagation neural networks," *J. Theor. Appl. Inf. Technol.*, vol. 47, no. 3, pp. 1264–1268, 2013.
[54] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Quantized neural networks: Training neural networks with low precision weights and activations," *The J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6869–6898, 2017.
[55] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
[56] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, "Linear discriminant analysis: A detailed tutorial," *AI Commun.*, vol. 30, no. 2, pp. 169–190, May 2017.
[57] F. Alam and G. Riccardi, "Comparative study of speaker personality traits recognition in conversational and broadcast news speech," in *Proc. INTERSPEECH*, 2013, pp. 2851–2855.
[58] 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.
[59] T. Salimans and D. P. Kingma, "Weight normalization: A simple reparameterization to accelerate training of deep neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 901–909.
[60] D. Poáap, "An adaptive genetic algorithm as a supporting mechanism for microscopy image analysis in a cascade of convolution neural networks," *Appl. Soft Comput.*, vol. 97, Dec. 2020, Art. no. 106824.
[61] H. Sadr, M. M. Pedram, and M. Teshnehlab, "A robust sentiment analysis method based on sequential combination of convolutional and recursive neural networks," *Neural Process. Lett.*, vol. 50, no. 3, pp. 2745–2761, Dec. 2019.
[62] H. Sadr, M. M. Pedram, and M. Teshnelab, "Improving the performance of text sentiment analysis using deep convolutional neural network integrated with hierarchical attention layer," *Int. J. Inf. Commun. Technol. Res.*, vol. 11, no. 3, pp. 57–67, 2019.
[63] M.-A. Carbonneau, E. Granger, Y. Attabi, and G. Gagnon, "Feature learning from spectrograms for assessment of personality traits," *IEEE Trans. Affect. Comput.*, vol. 11, no. 1, pp. 25–31, Jan. 2020.

EFFAT JALAEIAN ZAFERANI was born in Mashhad, Iran, in 1987. She received the basic education in Tehran. She received the B.Sc. degree in biomedical engineering majoring in bioelectric from Shahed University, after entry exam of Iran Universities, and the M.Sc. degree in biomedical engineering, from the K. N. Toosi University of Technology, Tehran, Iran, where she is currently pursuing the Ph.D. degree in biomedical engineering. Her research interests include optimization algorithms, fuzzy algorithms, signal processing, system identification, and deep neural networks.

MOHAMMAD TESHNEHLAB was born in Borujerd, Iran, in 1957. He received the B.Sc. degree in electrical engineering from Stony Brook University, NY, USA, in 1981, the M.Sc. degree in electrical engineering from Oita University, Japan, in 1991, and the Ph.D. degree from Saga University, Japan, in 1993. He is currently a Faculty Member of the Department of Electrical Engineering, K. N. Toosi University of Technology. His main research interest includes intelligent systems and control. He is a member of the Industrial Control Center of Excellence and the Founder of the Intelligent Systems Laboratory (ISLab.). His research interests include artificial rough and deep neural networks, fuzzy systems and neural nets, optimization, and applications in the identification, prediction, classification, and control. He is also the Head and the Co-Founder of the Intelligent Systems Scientific Society of Iran (ISSSI) and a member of the International Journal of Information and Communication Technology Research (IJICTR) Editorial Board.

MANSOUR VALI was born in Esfahan, Iran, in 1973. He received the B.Sc. degree in electrical engineering from the Isfahan University of Technology, in 1998, the M.Sc. degree in bioelectric engineering from the Sharif University of Technology, in 2000, and the Ph.D. degree in biomedical engineering from the Amirkabir University of Technology, Tehran, Iran, in 2006. Then, he worked for one year at the Biomedical Engineering Group, University of Isfahan, as a Faculty Member. From 2007 to 2012, he was a Faculty Member of the Biomedical Engineering Group, Shahed University. In February 2013, he joined the K. N. Toosi University of Technology, where he is currently an Assistant Professor with the Biomedical Engineering Group. His main research interests include sound and speech processing in medical and psychological assessments whose results are presented as a new course at the Department of Electrical Engineering, K. N. Toosi University of Technology, for supplementary students by him. He is also working on big data processing in medical applications recently, and he is trying to progress its advantages among physicians and hospital managers.