Snore-GANs: Improving Automatic Snore Sound Classification With Synthesized Data

Zixing Zhang, *Member, IEEE*, Jing Han, *Student Member, IEEE*, Kun Qian, *Student Member, IEEE*, Christoph Janott, *Student Member, IEEE*, Yanan Guo, and Björn Schuller, *Fellow, IEEE*

***Abstract*—One of the frontier issues that severely hamper the development of automatic snore sound classification (ASSC) associates to the lack of sufficient supervised training data. To cope with this problem, we propose a novel data augmentation approach based on semi-supervised conditional generative adversarial networks (scGANs), which aims to automatically learn a mapping strategy from a random noise space to original data distribution. The proposed approach has the capability of well synthesizing "realistic" high-dimensional data, while requiring no additional annotation process. To handle the mode collapse problem of GANs, we further introduce an ensemble strategy to enhance the diversity of the generated data. The systematic experiments conducted on a widely used Munich–Passau snore sound corpus demonstrate that the scGANs-based systems can remarkably outperform other classic data augmentation systems, and are also competitive to other recently reported systems for ASSC.**

***Index Terms*—Snore sound classification, obstructive sleep apnea, data augmentation, data synthesis.**

Manuscript received October 5, 2018; revised January 18, 2019 and February 19, 2019; accepted March 19, 2019. Date of publication April 1, 2019; date of current version January 6, 2020. This work was supported in part by the U.K.'s Economic and Social Research Council through the Research Grant HJ-253479 (ACLEW) and in part by the EU's Horizon 2020/EFPIA Innovative Medicines Initiative through Grant 115902 (RADAR-CNS). *(Corresponding author: Zixing Zhang.)*

Z. Zhang is with the Group on Language, Audio and Music, Imperial College London, London SW7 2AZ, U.K. (e-mail: zixing.zhang@imperial.ac.uk).

J. Han is with the ZD.B Chair of Embedded Intelligence for Health Care and Wellbeing, University of Augsburg, Augsburg 86159, Germany (e-mail: jing.han@informatik.uni-augsburg.de).

K. Qian was with the Machine Intelligence and Signal Processing Group, MMK, Technical University of Munich, Munich 80333, Germany, and also with the ZD.B Chair of Embedded Intelligence for Health Care and Wellbeing, University of Augsburg, Augsburg 86159, Germany. He is now with the Educational Physiology Laboratory, Graduate School of Education, The University of Tokyo, Tokyo 113-8654, Japan (e-mail: qian@p.u-tokyo.ac.jp).

C. Janott is with the Institute for Medical Engineering, Technical University of Munich, Garching 85748, Germany, and also with the audEERING GmbH, Gilching 82205, Germany (e-mail: cjanott@audeering.com).

Y. Guo is with the ZD.B Chair of Embedded Intelligence for Health Care and Wellbeing, University of Augsburg, Augsburg 86159, Germany, and also with Lanzhou University, Lanzhou 730000, China (e-mail: yanan.guo@informatik.uni-augsburg.de).

B. Schuller is with the ZD.B Chair of Embedded Intelligence for Health Care and Wellbeing, University of Augsburg, Augsburg 86159, Germany, with the Group on Language, Audio and Music, Imperial College London, London SW7 2AZ, U.K., and also with the audEERING GmbH, Gilching 82205, Germany (e-mail: bjoern.schuller@imperial.ac.uk).

I. Introduction

AUTOMATIC snore sound classification (ASSC) targets at developing an automated and non-invasive method for the classification of Obstructive Sleep Apnea (OSA) based on the snore sound [1]–[5]. OSA is characterized by repetitive episodes of decreased (hypopnea) or completely halted (apnea) airflow during sleep, despite the effort to breathe. According to the statistic investigation in [6], approximately 3∼7% adult men and 2∼5% adult women in the general population around the world suffer from OSA. This leads to a serious deterioration of health conditions, such as daytime sleepiness, excessive fatigue, morning headache, and even high blood pressure and depression mood in a long-term case [6]–[9]. To treat OSA, doctors need to determine the obstructive position of the respiratory tract in the very beginning. A standard determination approach often associates with a Drug-Induced Sleep Endoscopy (DISE) procedure, in which a flexible nasopharyngoscope is introduced into the upper airway while the patient is in a state of artificial sleep [9], [10]. Vibration mechanisms and locations can be observed while video and audio signals are recorded. However, this diagnosis has many disadvantages, such as the exhaustive time-consumption and the high strain of patients [10]. All these disadvantages underline the necessity of ASSC.

However, the lack of sufficient amounts of labelled data has become one of the major barriers to its progress. The rationales behind this problem can be summarized into four points. i) *Data privacy*: Due to the sensitivity of health-associated data, patients are often reluctant to publicly share their data. In addition, data privacy regulations restrict the legal usage possibilities of health data [11]. ii) *Time exhaustion when collecting data*. For example, to collect less than one thousand labelled samples for the ASSC sub-challenge in the INTERSPEECH 2017 Computational Paralinguistics challenges, about ten years were taken across three hospitals [10], [12]. iii) *Imbalanced nature of classes*: In practice, the patients who suffer from a tongue base snoring or an epiglottis snoring are much fewer than the ones from other types of snoring [10]. iv) *A High requirement of qualified experts for data annotation*: To label these data, highly experienced experts are demanded to analyze the recorded data and determine the obstruction location based on their prior knowledge.

The data sparsity problem becomes even worse with the recent rise of high capacity deep neural networks, which are more hungry for data to avoid underestimated parameters and poorly

generalized networks [13]. *Data augmentation* is an appealing approach to alleviate the data sparsity problem because it is theoretically able to produce infinite amounts of labelled data at minimum expense. In the context of machine learning, a plethora of data augmentation approaches have been investigated [14]–[16], which generally fall into two groups based on either transformation or synthesis. The *transformation*-based approaches conduct a certain number of transformation operations on existing samples to generate additional samples while retaining the annotations. These transformation operations include, for example, random cropping, rotation, flips for image samples [14], or the addition of diverse noises for audio samples [15]. Nevertheless, such data augmentation does not improve data distribution which is determined by higher-level features. In contrast, the synthesis-based approaches manage to generate artificial samples given specific labels via a synthesizer. The *Synthetic Minority Oversampling Technique (SMOTE)* [17] is a typical synthesizer-based data augmentation approach, which has been widely used in the domain of machine learning. The underlying idea is the creation of a new set of artificial samples by means of the nearest neighbours belonging to the minority class. The problem of the synthesizer-based approaches lies in the realistic gap between the synthetic and real samples, leading the models to learn the wrong information from the synthetic samples. Therefore, improving the synthesizer is considered to be vital to close the gap.

Over the past few years, a promising generative model, namely *Generative Adversarial Networks* (GANs), has attracted extremely widespread research interests in machine learning [18]–[22]. It consists of two neural networks – a generator and a discriminator, which contest with each other in a two-player zero-sum game [18]. Since its inception, GANs have been consistently demonstrated to be powerful in generating impressively realistic images and natural languages [18], [23], [24]. In this light, GANs emerge as a potential tool for data augmentation. In the literature of machine learning, a handful of related studies have been reported for some applications. For instance, for gaze estimation, traditional synthesized images were further decorated by an adversarial network, improving the previous data augmentation models [25]. For object classification, images were straightforwardly generated by GANs to increase the size of the training set [26], [27], leading to remarkable performance improvement. For emotion recognition, several class-specific GANs were used to efficiently transfer data across different domains [28].

However, no relevant studies have been reported to use GANs to increase the quantity of annotated training data for intelligent health care, especially for the ASSC, to the best of our knowledge. Besides, despite the fact that some previous work focuses on synthesizing standalone samples, for example, images, its performance remains unclear in the case of sequential samples, such as audio data, which significantly differs from the standalone samples. The generation of sequential samples, however, heavily relies on the context information [24]. Albeit a handful of related studies reported in the audio processing domain, they either focus on speech enhancement [29], [30] or music creation [31].

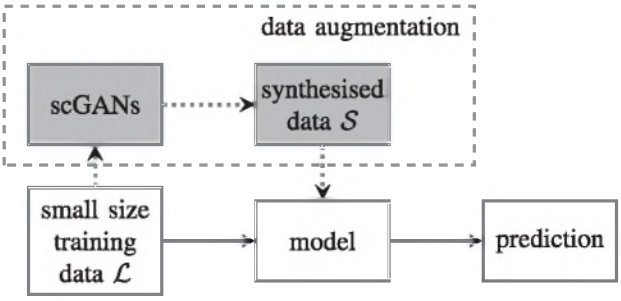


Fig. 1. Data augmentation framework based on semi-supervised conditional Generative Adversarial Networks (scGANs) for model training.

Motivated by the aforementioned analysis, we made the following contributions in the present article. i) We, for the first time, propose *semi-supervised conditional GANs* (scGANs) to generate high-dimensional representations for the ASSC. Compared with classic GANs, the generation process of scGANs is controlled by a condition, and thus there is no need for an additionally exhausting annotation process. Furthermore, in contrast to conditional GANs [28], [32], the proposed scGANs require only one model to synthesize different categorical data by the integration of semi-supervised GANs. Besides, when designing the scGANs, we choose the vanilla GANs, rather than other advanced GANs such as Wasserstein GANs [33], following the principle of the worst-case scenario and for the sake of easy performance comparison. ii) We try to synthesize not only the static acoustic data, but also the sequential acoustic data. For the sequential data, we innovate a recurrent sequence generator with recurrent neural networks, instead of a static data generator. iii) We introduce an ensemble of GANs to deal with the mode collapse problem. iv) We comprehensively investigate three widely used benchmark systems to evaluate the robustness of the proposed methods.

The remainder of this article is organized as follows. In Section II, we elaborately describe the proposed data augmentation framework with semi-supervised conditional generative adversarial neural networks. Then, in Section III, we introduce the database and the experimental setups, followed by the description, analysis, and discussion of the experimental results and findings in Section IV. Finally, we draw conclusions and suggest future research directions in Section V.

II. Methods

In this section, we first outline the proposed data augmentation framework based on scGANs. Then, we comprehensively describe the principle of GANs and semi-supervised conditional GANs, followed by dynamic alternation and ensemble GANs strategies that are introduced to overcome the training instability and mode collapse problems of scGANs. We finally report the approach to generate acoustic sequences.

A. The Framework of GANs-Based Data Augmentation

The framework of scGAN-based data augmentation is illustrated in Fig. 1. In this framework, synthesized data $\mathcal{S}$ is

artificially generated through scGANs (see Section II-B for more details), and then combined with the original data from a small-sized training set $\mathcal{L}$. The expanded data set, i. e., $\mathcal{S} \cup \mathcal{L}$, is further employed to train a model. In this work, we aim to generate high-dimensional representations (features) rather than the raw samples mainly because of the difficulty of learning massive variables in a continuum. Under the assumption that the model trained with augmented data shows better performance than the one merely trained with the small-size data set, the simulated data are expected to be able to well reflect the distribution of real data.

Albeit the availability of some other promising generative models in machine learning, GANs usually empirically outperform them, such as variational autoencoders [34] on the quality of images, and PixelRNN/PixelCNN on the processing speed [35], [36].

B. Semi-Supervised Conditional GANs

The vanilla GANs were first introduced in 2014 by Goodfellow [18]. They comprise two basic components: a *generator* (denoted as G) and a *discriminator* (denoted as D). The G aims to capture the potential distribution of real samples and generates new samples to 'cheat' the D as far as possible; whereas the D is often a binary classifier, distinguishing the sources (i. e., real samples or generated samples) of the inputs as accurately as possible. Therefore, the G and D are normally jointly trained in a two-player zero-sum game, where the total gains of the two players are zero. More details of the vanilla GANs can be found in [18].

One major problem of the above unconditioned GANs as aforementioned is the lack of label information when generating the data, which constrains its application to data augmentation. *Conditional GANs* (cGANs), however, utilize auxiliary information $\mathbf{c}$, such as the labels or a particular attribute setting, to control the output as desired [37].

Besides, Odena recently proposed *semi-supervised GANs* (sGANs) [38], where the D becomes a combination of a classifier and a discriminator. In detail, the discriminator D classifies the input into $K+1$ classes, where K is the number of classes of a classification task. Real samples are supposed to be classified into the first K classes and the generated samples into the $K+1$-th class (i. e., fake). In the framework, however, the generator G aims to generate data that is classified into *any* of the first K classes. The benefit of this strategy is two-fold: Firstly, the approach performs well to find the distinguishing boundary, hence creating a data-efficient classifier. Secondly, it empirically performs more efficient for generating higher quality samples than regular GANs[38], [39].

Motivated by this work [38], [39], we propose a novel structure, namely *semi-supervised conditional GANs* (scGANs), as structured in Fig. 2. They can be considered as extensions of cGANs by forcing the discriminator D to output class labels as well as distinguishing the real data from the fake data. Differing from sGANs [38], the G of the scGANs is conditioned with auxiliary information (i. e., label information in this case), and

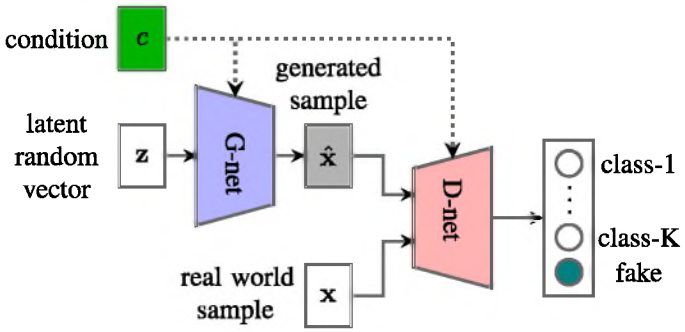


Fig. 2. The framework of semi-supervised conditional Generative Adversarial Network (scGANs).

aims to generate data that can be *correctly* classified into the first K classes given the condition c.

Mathematically, given real data $\mathbf{x}$ sampled from the distribution $p_{data}(\mathbf{x})$, a latent random vector $\mathbf{z}$ sampled following a simple prior distribution $p_{\mathbf{z}}(\mathbf{z})$ (e. g., uniform or Gaussian distribution), and the parameters θ_g and θ_d of the G and D networks, respectively, the generator G targets at maximizing the log-likelihood that it assigns to the correct classes:

$$\mathcal{L}_G = \mathbb{E}_{\mathbf{z}\sim p_c(\mathbf{z})}[\log P(y=k|\hat{\mathbf{x}})], \tag{1}$$

whilst the discriminator D aims to maximize the following log-likelihood:

$$\begin{aligned}\mathcal{L}_D = &\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}(\mathbf{x})}[\log P(y=k|\mathbf{x})] \\ &+ \mathbb{E}_{\mathbf{z}\sim p_z(\mathbf{z})}[\log P(y=fake|\hat{\mathbf{x}})],\end{aligned} \tag{2}$$

where k is among the first K classes, $\hat{\mathbf{x}} = G_{\theta_g}(\mathbf{z}|\mathbf{c})$, and $p_c(\mathbf{z})$ indicates the latent random distribution relating to the conditional information $\mathbf{c}$. By taking the class distribution into the objective function, an overall improvement in the quality of the generated samples is expected.

It has to be noticed that the proposed scGANs differ from the ones in [40], which are structured with two discriminators, used for unsupervised (with unlabelled real data) and supervised (with class-specific real data) true/false classification, respectively. The proposed scGANs, however, can be further extended with two discriminators in case of exploiting unlabelled data in future efforts, which is beyond the research scope of this article.

C. Dynamic Alternation and Ensemble of Semi-Supervised Conditional GANs

Generally, the training of G and D is conducted in an iterative manner, i. e., the corresponding neural weights θ_d, θ_g are updated in turns [18]. Once the training is completed, the generator G is able to generate more realistic samples, while the discriminator D can distinguish authentic data from fake data. The adversarial training process, however, suffers from two major issues: *training instability* and *mode collapse* [19], [20].

When training the adversarial networks, ensuring the balance and synchronization between the G and D plays an important role in obtaining reliable results [18]. That is, the optimization

goal of adversarial training lies in finding a saddle point of, rather than a local minimum between G and D. The inherent difficulty in controlling the synchronization of the two adversarial networks increases the risk of *instability* in the training process.

In this light, we introduce a simple and efficient way called *dynamic alternation*. That is, we dynamically alternate the training epochs between the generator G and the discriminator D, in contrast to the conventional approaches which often fix the training epochs for both (fixed alternation). It is hoped that this approach is able to keep the learning pace synchronously updated between G and D, so as to avoid the training instability.

Mathematically, we respectively define a loss threshold function for G and D with

$$\mathcal{L}_{TG/D} = \max(\Lambda^i + b, c), \tag{3}$$

where Λ, b, and c are the hyper-parameters which control the threshold together at the i-th training iteration. To guarantee $\mathcal{L}$ being a monotonically decreasing function, Λ is normally less than 1. In this article, these hyper-parameters are determined by empirical experience. Once the training loss from G is below a pre-defined loss $\mathcal{L}_{TG}$, the training process is altered to D. Similarly, once the training loss from D is below another pre-defined loss $\mathcal{L}_{TD}$, the training process is altered to G. Such an alternation keeps repeating until a training convergence of G and D. In doing this, we force the performance improvement of G and D at a similar pace.

Apart from the training instability, another issue is the *mode collapse*, which indicates that the generated samples have integrated into a small subset of similar samples (partial collapse), or even a single sample (complete collapse). In this case, the G exhibits very limited diversity amongst generated samples, thus reducing the usefulness of GANs.

To address this problem, some approaches are continually emerging. For example, the cost function of the generator can be modified to factor the diversity of generated batches [39]. Moreover, the unroll-GANs allow the generator to 'unroll' updates of the discriminator in a manner which is fully differentiable [41].

More recently, the work shown in [42], especially its advanced version [43], demonstrated that an ensemble of several networks with different network structures or initializations can improve the system performance significantly, in comparison with the aforementioned unroll-GANs [42], [43]. To this end, we implement a standard ensemble approach [42] in our experiments for the sake of easy comparison, hoping to obtain a better estimation of the real data distribution p_{data}. The framework of the ensemble of scGANs for data augmentation is depicted in Fig. 3. Instead of training a single scGANs pair, we train a set of scGANs. These scGANs are with different network structures (i. e., a different number of hidden nodes per layer in our experiments) in order to maximally explore their differences, and trained independently. When conducting data augmentation, we aggregate the data from all scGANs, and randomly select data from the pool which are further merged into the original training set. By doing this, it is expected to expand the diversity of the augmented data that come from separate scGANs.

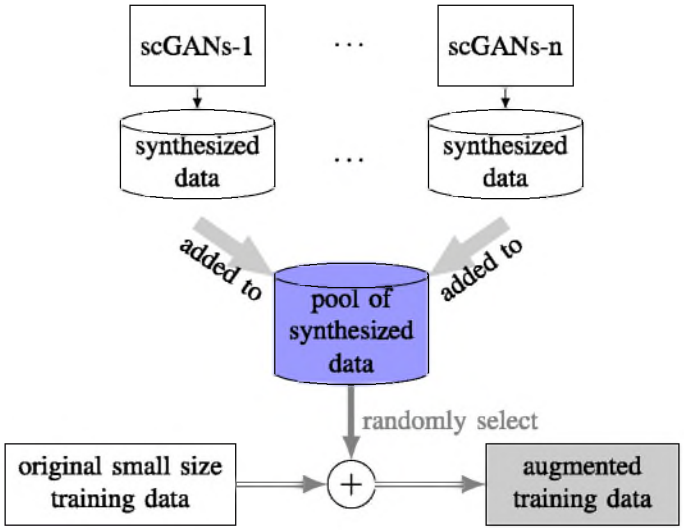


Fig. 3. Data augmentation by using an ensemble of semi-supervised conditional Generative Neural Networks (scGANs). n: the number of scGANS.

D. Sequence Generation

The snore data are normally structured in a sequence. However, most available GANs were particularly designed to generate standalone samples (e. g., images). In this section, we introduce a novel approach to generate sequential samples by means of the GANs equipped with Recurrent Neural Networks (RNNs) with Gated Recurrent Units (GRUs), since the GRU-RNNs have been widely known to be efficient in capturing long-range context information [44]–[46].

To be more specific, given a sequence $\mathbf{x}_{1:T} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T\}$, which comprises of T-length consecutive high-dimensional vectors $\mathbf{x}$, the goal of the recurrent GANs is to learn

$$f(\mathbf{x}_t|\mathbf{x}_1, \ldots, \mathbf{x}_{t-1}; \mathbf{c}), \tag{4}$$

while $\mathbf{x}_1 = f(\mathbf{z}; \mathbf{c})$. In doing this, they are able to generate a complete sequence $\hat{\mathbf{x}}_{1:T}$ by feeding a latent random noise $\mathbf{z}$, i. e., $\hat{\mathbf{x}}_{1:T} = G(\mathbf{z})$. Intuitively, we illustrate the sequence generation process in Fig. 4.

The generation process is indeed inspired by the Seq2Seq modeling [47], where the decoder component takes the last output at time $t-1$ as its input at time t, and takes the previous hidden state at time $t-1$ as its initial state at time t. Differently, the conditional vector $\mathbf{c}$ is consistently used to guide G to produce a designed sequence.

As to the discriminator D, we further utilize another GRU-RNN to distinguish the generated sequences from the real ones.

III. Training and Validation

To evaluate the performance of the proposed data augmentation approaches for ASSC, we selected the Munich-Passau Snore Sound Corpus (MPSSC). The corpus has been widely used in the intelligent health care research community [3], [10], and has been employed as an official database for an ASSC

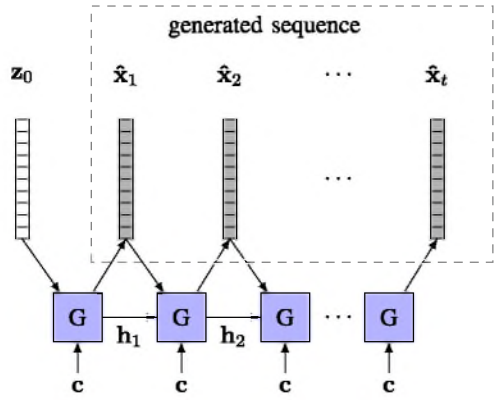


Fig. 4. Sequence generation through a recurrent generator (h_t: hidden states at time t, c: conditional vector).

sub-challenge in the INTERSPEECH 2017 Computational Paralinguistics challenges [12].

A. The Munich-Passau Snore Sound Corpus

The MPSSC was introduced to classify the vibration location within the upper airways when snoring [3], [10], [48]. Since the data analysis and annotation process did not involve any studies carried out by the authors on humans or animals, an ethical approval was not required. Starting material for the database were existing recordings of DISE examinations from three medical centres in Germany (i. e., Klinikum rechts der Isar, Technical University Munich; Alfried Krupp Hospital Essen, and University Hospital Halle/Saale) which were taken during clinical routine examinations between 2006 and 2015, using different recording devices among the medical centres. In a DISE examination, the patient is slightly sedated and put into a condition that resembles an artificial sleep state. By means of a flexible nasopharyngoscope, the upper airways are observed by an experienced ENT physician, identifying the locations of tissue vibration or airway narrowing while the patient snores or undergoes obstructive events. Both the audio signals from the microphone and the video signals from the nasopharyngoscope were recorded synchronously. For the database, in excess of 30 hours of DISE recordings were analyzed.

For our experiments, the audio signal was extracted from the mp4 recordings and stored in a wav-format (16 bit, 44.1 kHz). To detect the snore sound events from the audio files, an automated algorithm was employed. In details, we averaged the absolute value of the signal amplitude in 10 ms segments with no overlap and determined the background noise level by means of a 1024-step histogram averaging 10 s segments [10]. Only the segments, which exceed two times the predefined background noise level for a minimum duration of 300 ms, were annotated. After that, we added 100 ms of signals before and after the actual onset and end of the event, which was then extracted from the original audio file, normalized, and saved as separate wav files (16 bit, 16 kHz) [10]. Finally, an experienced human listener (the fourth author) listened to all selected events and classified them manually as either pure snoring (snore) or other sounds (non-snore, or the snore severely disturbed by non-static background noise) [10]. For more details of this pre-processing step, please refer to [10].

TABLE I
DATA DISTRIBUTION OF THE MUNICH-PASSAU SNORE SOUND CORPUS (MPSSC). V: VELUM; O: OROPHARYNX; T: TONGUE; E: EPIGLOTTIS

#	train	devel	test	$\sum$
V	161	168	155	484
O	75	76	65	216
T	15	8	16	39
E	32	30	27	89
$\sum$	283	282	263	828

The selected snore events were then classified by medical ENT (ear, nose, and throat) experts based on the findings from video recordings. Only events with a clearly identifiable (i. e., single site of vibration and without obstructive disposition) were included in the database, resulting in 828 snore events in total. Based on the VOTE scheme that is widely used to distinguish four structures involved in airway narrowing and obstruction [49], we defined four classes: V (i. e., Velum, including soft palate, uvula, lateral velopharyngeal walls), O (i. e., Oropharyngeal lateral walls, including palatine tonsils), T (i. e., Tongue, including tongue base and airway posterior to the tongue base), and E (i. e., Epiglottis).

The annotated audio samples were then separated into subject-independent training, development, and test partitions. Table I displays the data distribution by partitions and classes. The database is strongly imbalanced with a comparatively low number of T and E samples. This is consistent with earlier medical research, which has found that respiratory disturbances occur more frequently at velopharyngeal and oropharyngeal level, compared to the hypopharyngeal level [50]. For more specific information about the database, the readers are referred to [10].

B. Representations

To keep in line with the ASSC benchmark of the 2017 INTERSPEECH Computational Paralinguistics challenges [12], we chose three different kinds of acoustic feature sets at either the frame level (i. e., low-level descriptor) or the segment level (i. e., functional or Bag-of-Audio-Words).

1) Low-Level Descriptors: We used the ComParE16 high-dimensional acoustic feature set employed in [51], which contains 65 frame-wise Low-Level Descriptors (LLDs, e. g., energies, Mel-frequency cepstral coefficients, zero-cross rate, jitter, shimmer, probability of voicing) as well as their first derivations, leading to 130 LLDs. These LLDs are determined according to a set of brute-force empirical evaluations on computational paralinguistics [12], [51]. More detailed information about the ComParE16 LLDs can be found in Table II.

2) Functional-Based Features: Intuitively, the functional-based approach projects the temporal LLD contours onto a set of feature vectors with descriptive statistic functionals (see [52]

TABLE II
THE COMPARE ACOUSTIC FEATURE SET INCLUDES 65 LOW-LEVEL DESCRIPTORS (LLDS) OF DIFFERENT TYPES, AS WELL AS THEIR FIRST DERIVATIONS (DELTA), RESULTING IN 130 LLDS

4 energy-related LLDs	**Group**
RMS energy, zero-crossing rate	Prosodic
sum of auditory spectrum (loudness)	Prosodic
sum of RASTA-filtered auditory spectrum	Prosodic
55 spectral LLDs	**Group**
MFCC 1–14	Cepstral
psychoacoustic sharpness, harmonicity	Spectral
RASTA-filt. aud. spect. bds. 1–26 (0–8 kHz)	Spectral
spectral energy 250–650 Hz, 1 k–4 kHz	Spectral
spectral flux, centroid, entropy, slope	Spectral
spectral roll-off point 0.25, 0.5, 0.75, 0.9	Spectral
spectral variance, skewness, kurtosis	Spectral
6 frequency-related LLDs	**Group**
f_0 (SHS and Viterbi smoothing)	Prosodic
probability of voicing	Voice quality
log. HNR, jitter (local and δ), shimmer (local)	Voice quality

for more details). Mathematically, this can be written as follows:

$$\mathbf{f} = f([\mathbf{x}_i], i = 1, \ldots, T), \tag{5}$$

where $\mathbf{f}$ denotes the segment-level feature vector; $[\mathbf{x}_i]$ indicates the sequential frame-wise LLDs; T is the total frames of a given vocalization; and f denotes the *functionals* (i. e., statistic information) that are applied per LLD contour. Specifically, the functionals can include: extremes (minimum, maximum, ranges, etc.), mean (arithmetic, quadratic, geometric), moments (variance, skewness, kurtosis, etc.), percentiles (quantiles, ranges, etc.), peaks (number, distances, etc.), temporal variables (durations, positions, etc.), and regression (coefficients, error). For our experiments, the functional-based feature set contains 6 373 dimensional feature vectors [51].

3) Bag-of-Audio-Words: Bag-of-Audio-Words (BoAW) is another type of segment-level acoustic representation. Extracting BoAW involves three steps: i) codebook generation; ii) vector quantization; and iii) histogram construction. Differing from bag-of-words for linguistic analysis, the total number of audio-words (frame-wise LLDs) is indeed numerous with an equal occurrence frequency of one. To reduce the codebook size (S), a k-means clustering or a random sampling is conducted to determine the codewords (W) of the codebook (C) [53]. After that, a multi-assignment quantization technique is executed to map each audio-word to the first n closest codewords, measured by Euclidean distance. Finally, a histogram is constructed by calculating the counts of occurrence of each codeword in all acoustic frames over one vocalization segment. Mathematically, the histogram representation $\mathbf{b}$ for a given vocalization v with T_v frames is

$$\mathbf{b} = \left[\sum_{i=1}^{T_v} \phi_{i,m} \right], m = 1, \ldots, S, \tag{6}$$

where $\phi_{i,m}$ equals to 1 if the i-th frames is assigned to the m-th codeword, otherwise, to 0. To minimize the effects relating to the length disparities of different vocalizations, a normalization process is further undertaken over $\mathbf{b}$, to sum up all elements of $\mathbf{b}$ to one. More details about the BoAW generation can be found in [53].

C. Implementation Setups

As to the acoustic features, we utilized the open-source toolkit *openSMILE* [52] to extract the LLDs and functional-based features, and the toolkit *openXBOW* [53] to distill the BoAW representations.

When simulating the representations, we deployed GRU-RNN-based scGANs. The generator and discriminator used the same network structure, with two hidden layers and N nodes per hidden layer, where N was set to be 60. As to the discriminator, we appended an additional dense layer and a softmax activation function for pattern classification. As to the GRUs, we employed the standard version with sigmoid and tangent activation functions [44]. To train the networks, we employed the Adam optimization algorithm with an optimized learning rate of 0.001 for the generator and 0.01 for the discriminator. The batch size was set to be 64 to facilitate the training process. To improve the generalization of the neural networks, we further applied an L2 regularization term to the loss function with a regulation value of 10E-4. Note that all these hyper-parameters were optimized on the development set with the baseline system – the one without GAN-based data augmentation and taking LLD acoustic features as inputs. Thus, it avoids exhausting computation caused by the grid-searching in numerous experimental scenarios. Besides, we set the initial state of GRU to be zero, and the initial weights of neural networks to be random values with a standard deviation of 0.1. To find the saddle point between the generator and discriminator when training the GANs, we employed the dynamic alternation strategy as described in Section II-C to alternatively train the generator and discriminator. Specifically, with respect to Eq. (3), we set Λ, b, and c to be 0.95, 0, and 0.7 for the discriminator, and 0.95, 1.0, and 1.0 for the generator. These hyper-parameters were set according to empirical experience. In further, we could use more advanced approaches to search these values, for example, reinforcement learning [46].

For the sequence generation, we partitioned the original sequence with variable length into multiple continuous segments with a fixed window size of 400 ms and a step size of 100 ms. This partially reduces the complexity of sequence generation.

Due to the distinct characteristics of the three investigated features (cf. Section III-B), we considered one static model, i. e., Support Vector Machines (SVMs), which aims to learn the segmental-level features (i. e., based on functionals or BoAWs), and one dynamic model, i. e., GRU-RNNs, which attempts to learn the sequential frame-level LLDs. In our experiments, three learning systems have been implemented, referring to i) functional-based features with SVMs (*functionals + SVMs*), ii) BoAW-based features with SVMs (*BoAWs + SVMs*), and iii) sequential LLDs with GRU-RNNs (*LLDs + GRU-RNNs*), respectively. Particularly, the selection of SVMs rather than other typical classifiers mainly relates to two reasons: i) SVMs have been officially employed in the 2017 INTERSPEECH ASSC sub-challenge [12], as well as other related studies [54]–[56];

TABLE III

PERFORMANCE (UAR AS WELL AS CORRESPONDING STANDARD DEVIATION [SD]) COMPARISON ON BOTH THE DEVELOPMENT AND TEST SETS AMONG THE PROPOSED SCGAN-BASED DATA AUGMENTATION APPROACHES, TRADITIONAL DATA AUGMENTATION APPROACHES, AND THE BASELINE SYSTEMS IN THREE LEARNING SYSTEMS, I. E., FUNCTIONALS + SVMS, BOAWS + SVMS, AND LLDS + GRU-RNNS. EXPERIMENTS ARE REPEATED IN 20 INDEPENDENT RUNS. NET-60: DEFAULT SCGANS NETWORK STRUCTURE WITH 60 NODES PER HIDDEN LAYER; AVERAGE: AVERAGED RESULTS OVER FOUR DIFFERENT NETWORK STRUCTURES OF SCGANS; ENSEMBLE: AN ENSEMBLE OF SCGANS

approaches	functionals + SVMs		BoAWs + SVMs		LLDs + GRU-RNNs	
UAR_{SD} [%]	dev	test	dev	test	dev	test
baseline (wo DA)	$45.3_{\pm 0.0}$	$46.2_{\pm 0.0}$	$41.4_{\pm 0.0}$	$48.2_{\pm 0.0}$	$65.7_{\pm 5.1}$	$52.5_{\pm 2.8}$
transformation [15]	$46.8_{\pm 0.9}$	$48.0_{\pm 1.2}$	$41.1_{\pm 1.3}$	$48.0_{\pm 1.2}$	$67.8_{\pm 4.4}$	$53.6_{\pm 3.2}$
SMOTE*[17]	$45.1_{\pm 0.5}$	$47.0_{\pm 0.8}$	$41.3_{\pm 0.5}$	$47.9_{\pm 1.1}$	–	–
cGANs (net-60)	$45.1_{\pm 1.3}$	$46.0_{\pm 1.9}$	$41.3_{\pm 4.8}$	$43.9_{\pm 3.1}$	$67.0_{\pm 4.1}$	$53.3_{\pm 3.6}$
scGANs (net-60)	$52.7_{\pm 0.7}$	$49.9_{\pm 0.5}$	$45.8_{\pm 2.6}$	$54.8_{\pm 2.9}$	$66.7_{\pm 3.8}$	$52.3_{\pm 3.4}$
scGANs (average)	$51.4_{\pm 1.4}$	$50.3_{\pm 1.0}$	$46.3_{\pm 2.2}$	$51.9_{\pm 2.4}$	$66.3_{\pm 4.1}$	$53.2_{\pm 3.1}$
scGANs (ensemble)	$53.8_{\pm 2.4}$	$\mathbf{51.5_{\pm 1.1}}$	$46.8_{\pm 2.8}$	$\mathbf{56.7_{\pm 3.4}}$	$67.4_{\pm 4.0}$	$\mathbf{54.4_{\pm 3.8}}$

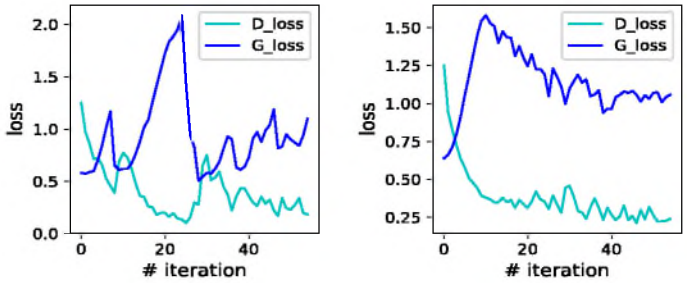
$\star$ SMOTE has not supported to synthesis sequences yet

ii) our previous experimental results have shown that SVMs generally perform more stable and better than other typical classifiers (e. g., random forest) on the MPSSC database. For example, we obtained the UARs on the development and test sets of 34.9% and 35.2% by using functional+RF system, and 38.9% and 55.2% by using BoAW + RF system. These results are generally inferior to the obtained results by SVMs (cf. Table III).

As to the SVMs, the complexity was determined on the development set through the baseline experiments without data augmentation by searching values among $[0.00001, 0.00005, 0.0001 \ldots, 0.5, 1, 5]$. Empirically, the complexity was optimized to be $10E-4$ and $10E-3$ in the cases of functional-based features and BoAW representations, respectively. For the GRU-RNNs model, we employed the same network structure as for the discriminator in the scGANs, but with only four output nodes (the discriminator has five output nodes due to the 'fake' prediction). When training the GRU-RNNs, a many-to-one strategy was used, i. e., after feeding a sequence of LLD vectors, only the last states from the hidden layers were considered for final classification. Again, the Adam optimizer was implemented with the same learning rate and L2 regularization value with the parameters of the discriminator.

To evaluate the performance of the investigated systems, we kept in line with the evaluation metric, i. e., *Unweighted Average Recall* (UAR), which was officially employed in the 2017 INTERSPEECH ASSC sub-challenge, for the sake of performance comparison. The UAR is calculated by the sum of recalls per class divided by the number of classes, and thus can reflect a meaningful overall accuracy despite class imbalances, such as the one we are facing.

Due to the imbalanced class distribution of the MPSSC database, we oversampled the data from the minority classes by means of replication, forcing an even distribution. Different from the proposed data augmentation approach that aims to synthesize completely new data, this strategy increases the weights of the losses from the minority samples. The advantage relates to the fact that it increases the contributions of the original minority samples which hold grounded class information when modeling ASSC. Notably, when using GRU-RNNs to classify each recording, a majority voting strategy was applied to a set of related segments to come up with a final prediction, since each recording was split into several sub-segments as aforementioned.



(a) fixed alternation (b) dynamic alternation

Fig. 5. The variation of losses while training the scGAN at every iteration using the fixed alternation strategy (a) or the dynamic alternation strategy (b).

IV. RESULTS AND DISCUSSION

In this section, we conducted comprehensive evaluations of the systems with proposed scGAN-based data augmentation (snore-GANs) on the selected MPSSC database.

A. Dynamic Alternation Evaluation

Before the systematic performance evaluation, we firstly investigated the efficiency of the introduced dynamic alternation training strategy for the scGANs. In Fig. 5, we plotted the obtained losses at each learning iteration of both the generator G and the discriminator D. From the figure, we can see that when using the conventional fixed-alternation training strategy, the obtained loss curves from both G and D severely vibrate along with the learning iterations, which clearly shows the training instability when the number of epochs per iteration is fixed for G and D (see Fig. 5(a)). In contrast, they are shown to be much smoother when using the dynamic alternation training strategy (see Fig. 5(b)). This suggests that the dynamic alternation training strategy is capable of improving the training stability of GANs and thus facilitates the convergence of the training

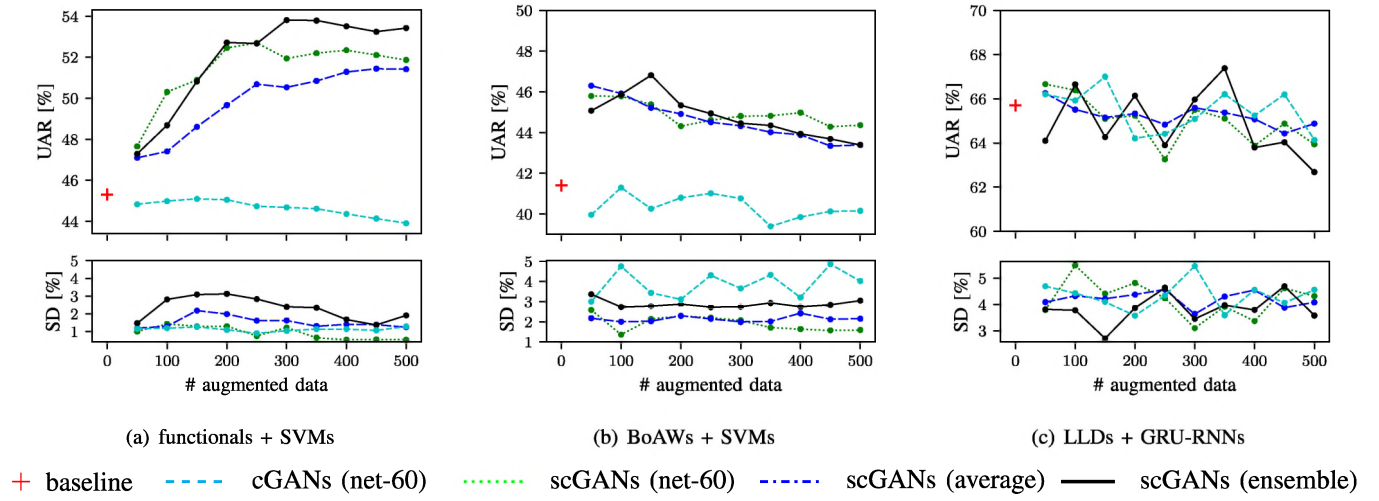


Fig. 6. Performance (UAR as well as corresponding Standard Deviation [SD]) of data augmentation on the development set when increasingly adding synthesized data to the three learning systems, i. e., functionals + SVMs (a), BoAWs + SVMs (b), and LLDs + GRU-RNNs (c). Experiments are repeated in 20 independent runs. scGANs: semi-supervised conditional GANs; cGANs: conditional GANs. net-60: default scGANs network structure with 60 nodes per hidden layer; average: averaged results over four different network structures of scGAN; ensemble: an ensemble of scGANs.

process, by forcing the learning process of both networks to be in a similar pace. Nevertheless, it is worth noting that to determine the hyper-parameters of the dynamic loss threshold requires empirical experience.

B. Results of scGANs and Discussion

When generating the data, we filtered out such samples that are not correctly recognized by the discriminator. In doing this, it potentially removes the noisy samples possibly falling beyond the scope of the original data distribution, and thus alleviates their adverse effect in learning. Moreover, when adding the generated data to the original training set, we randomly selected the generated samples evenly distributed over categories, in order to handle the imbalanced data distribution problem.

The dotted green curves in Fig. 6 depict the obtained performance of the three learning systems as aforementioned when increasingly adding generated data to the original training set, by using the default scGANs architecture (i. e., net-60; $N = 60$ nodes per hidden layer). To mitigate the performance fluctuation caused by the random selection of generated data and the random initialization of neural networks, we repeated 20 independent runs for each data augmentation experiment.

In the case of the 'functionals + SVMs' system (cf. Fig. 6(a)), it can be seen that the obtained UAR remarkably boosts from 45.3% to 47.8% when adding 50 synthesized samples per class, and dramatically to 52.7% when adding 250 synthesized samples per class. Notable gain can also be observed for the 'BoAWs + SVMs' system (i. e., from 41.4% to 45.9% UAR). For the 'LLDs + GRU-RNNs' system, a moderate improvement could be found (i. e., from 65.7% to 66.7% UAR). This tells us that a scGAN-based data augmentation approach can indeed improve the performance of the systems when dealing with sparse data. Besides, we compared this system with the LSTM-RNNs-based one with the same network architecture. The obtained results are

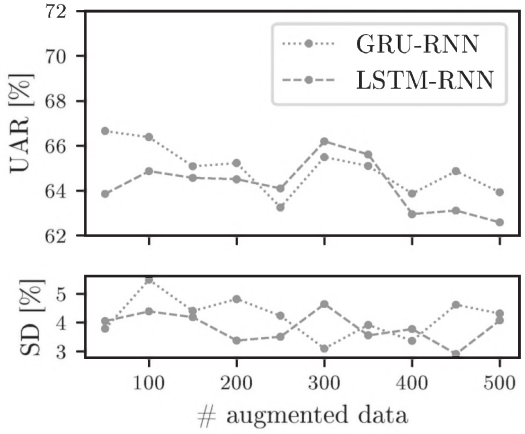
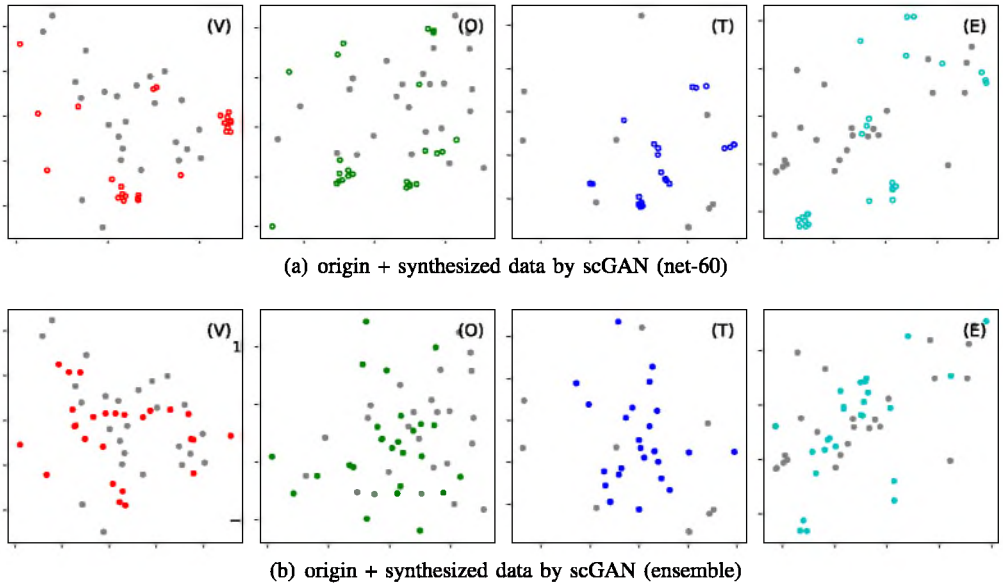


Fig. 7. Performance comparison between LSTM-RNNs and GRU-RNNs for the sequence generation.

shown in Fig. 7. It can be seen that GRU-RNNs are competitive to the LSTM-RNNs, but with fewer parameters to be trained.

When using the 'functionals + SVMs' system, we see that its performance is continuously improving from the beginning but then remains almost stable when increasingly adding synthesized data. This may attribute to the fact that the model is prone to learn more from the synthesized data due to their higher weights. Although increasing the capacity of a network may partially alleviate this problem, for the sake of better performance comparison, we retained the network architecture in all experimental scenarios. For the 'BoAWs + SVMs' and 'LLDs + GRU-RNNs' systems, the maximum positive effect is achieved with a comparatively low number of only 50 synthesized samples per class, with a slight deterioration when adding more. A possible explanation is a mode collapse problem, where the generated data do not well reflect the whole picture of the original data distribution.



(a) origin + synthesized data by scGAN (net-60)

(b) origin + synthesized data by scGAN (ensemble)

Fig. 8. Visualization (t-SNE) of the mixed original and synthesized data (i. e., functional-based features) in V, O, T, and E four categories. The synthesized data were generated through a mono-scGAN (net-60) (a) or an ensemble of scGANs (ensemble) (b). The grey points: original data; red, green, blue, and cyan circles/points: synthesized V, O, T, and E samples.

Therefore, it is of importance to find the optimal balance between original and synthesized data. Such an observation, however, is not obvious and the ideal ratio might best be determined by experiments.

We further notice that in the case of the 'LLDs + GRU-RNNs' system, the data augmentation provides a limited performance enhancement. This underlines the toughness of generating sequential LLDs and requires further improvement in future efforts.

Moreover, we compared scGANs with cGANs, of which the performance is shown in Fig. 6 with cyan curves. Obviously, scGANs are notably superior to cGANs in our case. This conclusion relates to the essential drawback of cGANs (cf. Section II-B). That is, although cGANs can simulate the overall data distribution, they are unable to guarantee the data distribution match for a particular snore sound category.

C. Results of the Ensemble of scGANs and Discussion

To assess the effectiveness of an ensemble of scGANs, we conducted the experiments with four different network structures of scGANs, i. e., N = 40, 60, 80, and 100. The black curves in Fig. 6 illustrate the system performance through an ensemble of scGANs (i. e., four scGANs).

Generally speaking, it can be seen that the ensemble of scGANs (i. e., ensemble) outperforms the mono-scGAN (i. e., net-60; dotted green curves) for data augmentation. The obtained UARs on the development set go up to 53.8%, 46.2%, and 67.4%, respectively, in the cases of 'functionals + SVMs', 'BoAWs + SVMs', and 'LLDs + GRU-RNNs'. We further averaged the performance of four mono-scGANs as outlined previously. Similar performance improvement of the ensemble of scGANs can be observed. Particularly, one can notice that more synthesized data are required to achieve the best performance when using an ensemble of scGANs. This implicitly indicates that the generated data are more diverse than the ones generated by a mono-scGAN, such that adding more generated data delivers better system performance.

To intuitively demonstrate this conclusion, Fig. 8 illustrates the data distribution of the mixed original data and synthesized data (based on functionals), either simulated by a mono-scGAN (a) or ensemble of GANs (b). Generally speaking, compared with the mono-scGAN, the ensemble of scGANs is capable of generating more diverse data that better reflect the original data distribution in all V, O, T, and E cases. The data diversity, on the other hand, potentially results in less stable system performance, as shown in Fig. 6 where the ensemble of scGANs generally shows higher standard deviations in 20 independent runs.

Particularly, we investigated two classic data augmentation approaches, i. e., audio transformation and SMOTE as briefly described in Section I. For the transformation data augmentation, we degraded the original audio signals through diverse noises (i. e., the CHiME noise[1] in rooms and the white noise) in different signal-to-noise ratios of $10 \sim 25$ dB. In total, data of ten times the size of the original set were generated. For the SMOTE approach, we synthesized the data belonging to the minority classes up to the number of the majority class (i. e., 161 samples).

In Table III we compare the best UARs achieved on both the development and test sets through distinct approaches, i. e., baseline systems without data augmentation, traditional data augmentation approaches, and the proposed scGAN-based approaches. Obviously, the results indicate that the scGAN-based data augmentation promotes the baseline systems without any data augmentation in the most scenarios. Furthermore, it is distinctly superior to the other two traditional data augmentation approaches. It can be seen that the SMOTE approach is merely

[1] obtained from http://spandh.dcs.shef.ac.uk/chime_challenge/chime2016/data.html

TABLE IV
PERFORMANCE COMPARISON IN TERMS OF UAR BETWEEN THE PROPOSED SYSTEM WITH SCGAN-BASED DATA AUGMENTATION AND OTHER STATE-OF-THE-ART APPROACHES ON THE MPSSC DATABASE. THE EXPERIMENTS WITH SGANS AND SCGANS WERE CONDUCTED IN 20 INDEPENDENT RUNS

approaches (UAR_{SD} [%])	dev	test
related state-of-the-art approaches		
end-to-end [57]	40.3	40.3
fused end-to-end and BoAW + SVM [12]	45.1	46.0
dual source filter + SVM [58]	49.6	—
fused GMM SV + SVM/RF, Spec. + CNN [59]	57.1	51.7
fused FV/func. + (W)KPLS/KELM [60]	—	64.2
sGANs (functionals)	$50.9_{\pm 1.9}$	$44.1_{\pm 3.5}$
sGANs (BoAWs)	$49.0_{\pm 2.7}$	$51.1_{\pm 3.9}$
sGANs (LLDs)	$63.2_{\pm 3.8}$	$52.8_{\pm 4.1}$
proposed snore-GANs (data augmentation by scGANs)		
functionals + SVMs	$53.8_{\pm 2.4}$	$51.5_{\pm 1.1}$
BoAWs + SVMs	$46.8_{\pm 2.8}$	$56.7_{\pm 3.4}$
LLDs + GRU-RNNs	$67.4_{\pm 4.0}$	$54.4_{\pm 3.8}$

competitive to the baseline, possibly because an upsampling operation has been applied to the original training set before the experiments (cf. Section III-C). Moreover, the system performance can be further improved when considering an ensemble of scGANs that is capable of dealing with the mode collapse problem of GANs.

D. Performance Comparison With Other State of the Art

To further compare the performance of our proposed data augmentation systems (snore-GANs) with other recently reported systems, we made a summary of the obtained UARs in Table IV. Generally speaking, it can be seen that our best-achieved results are competitive with, or even superior to, most of the other state-of-the-art systems. Particularly, we found that our systems can remarkably outperform the end-to-end system [57] (i. e., 56.7% vs 40.3% UARs) that recently has been consistently regarded as one of the most attractive systems in the audio analysis [57]. This somewhat confirms the data sparsity challenge for the deep learning-based approaches that often prefer to a large amount of training data. Although some promising results were achieved in previous works, such as [60], i) the results on the development set were not provided; ii) the results were obtained by fusing several different systems, in contrast to our results delivered by merely one system. Furthermore, we also observe that the snore-GANs outperform the conventional sGANs in three different feature scenarios. This implies that the synthesized data indeed can help provide additional class-specific information for the classification models.

E. Discussion

In future, we will keep collecting more snore sound data from different hospitals and patients to increase the data size and diversity, on which we will re-evaluate the proposed methods. Besides, more advanced or potential novel sequence generation systems (e. g., variational recurrent autoencoders) [61]–[63] will be further proposed and evaluated in our following work to improve the acoustic sequence generation models. Moreover, we intend to apply the approaches to other health care tasks (e. g., cardiopathy and epilepsy) associated with other modalities, such as biological signals, images, and video recordings.

V. CONCLUSION

To address the data scarcity problem for automatic snore sound classification (ASSC), we introduced a novel data augmentation approach based on semi-supervised conditional Generative Adversarial Networks (scGANs) in this article. By performing extensive experiments on the Munich-Passau snore sound corpus, we find that the scGANs-based data augmentation, especially its ensemble variation, is capable of generating new data which share a similar distribution with the original data, resulting in an increased quantity of training data without any human annotation efforts. By combining the synthesized and original data, the performance of ASSC systems was remarkably improved, indicating the effectiveness and robustness of the proposed approach for ASSC.

REFERENCES

[1] H. D. Nguyen, B. A. Wilkins, Q. Cheng, and B. A. Benjamin, "An online sleep apnea detection method based on recurrence quantification analysis," *IEEE J. Biomed. Health Inform.*, vol. 18, no. 4, pp. 1285–1293, Jul. 2014.

[2] J. Behar *et al.*, "SleepAp: An automated obstructive sleep apnoea screening application for smartphones," *IEEE J. Biomed. Health Inform.*, vol. 19, no. 1, pp. 325–331, Jan. 2015.

[3] K. Qian *et al.*, "Classification of the excitation location of snore sounds in the upper airway by acoustic multi-feature analysis," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 8, pp. 1731–1741, Aug. 2017.

[4] M. Młyńczak, E. Migacz, M. Migacz, and W. Kukwa, "Detecting breathing and snoring episodes using a wireless tracheal sensor—A feasibility study," *IEEE J. Biomed. Health Inform.*, vol. 21, no. 6, pp. 1504–1510, Nov. 2017.

[5] S. Gutta, Q. Cheng, H. Nguyen, and B. Benjamin, "Cardiorespiratory model-based data-driven approach for sleep apnea detection," *IEEE J. Biomed. Health Inform.*, vol. 22, no. 4, pp. 1036–1045, Jul. 2018.

[6] N. M. Punjabi, "The epidemiology of adult obstructive sleep apnea," *Proc. Amer. Thoracic Soc.*, vol. 5, no. 2, pp. 136–143, Feb. 2008.

[7] C. Karmakar, A. Khandoker, T. Penzel, C. Schobel, and M. Palaniswami, "Detection of respiratory arousals using photoplethysmography (PPG) signal in sleep apnea patients," *IEEE J. Biomed. Health Inform.*, vol. 18, no. 3, pp. 1065–1073, May 2014.

[8] J. M. Perez-Macias, M. Tenhunen, A. Värri, S.-L. Himanen, and J. Viik, "Detection of snores using source separation on an Emfit signal," *IEEE J. Biomed. Health Inform.*, vol. 22, no. 4, pp. 1157–1167, Jul. 2018.

[9] H. Yoon, S. H. Hwang, J.-W. Choi, Y. J. Lee, D.-U. Jeong, and K. S. Park, "Slow-wave sleep estimation for healthy subjects and OSA patients using R–R intervals," *IEEE J. Biomed. Health Inform.*, vol. 22, no. 1, pp. 119–128, Jan. 2018.

[10] C. Janott *et al.*, "Snoring classified: The Munich–Passau snore sound corpus," *Comput. Biol. Med.*, vol. 94, pp. 106–118, Mar. 2018.

[11] D. He, R. Ye, S. Chan, M. Guizani, and Y. Xu, "Privacy in the Internet of Things for smart healthcare," *IEEE Commun. Mag.*, vol. 56, no. 4, pp. 38–44, Apr. 2018.

[12] B. Schuller *et al.*, "The INTERSPEECH 2017 computational paralinguistics challenge: Addressee, cold & snoring," in *Proc. 18th Annu. Conf. Int. Speech Commun. Assoc.*, Stockholm, Sweden, 2017, pp. 3442–3446.

[13] Z. Zhang, N. Cummins, and B. Schuller, "Advanced data exploitation for speech analysis—An overview," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 107–129, Jul. 2017.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, Lake Tahoe, NV, USA, 2012, pp. 1106–1114.

[15] D. Amodei *et al.*, "Deep Speech 2: End-to-end speech recognition in English and Mandarin," in *Proc. 33rd Int. Conf. Mach. Learn.*, New York, NY, USA, 2016, pp. 173–182.

[16] T. Ko, V. Peddinti, D. Povey, and S. Khudanpur, "Audio augmentation for speech recognition," in *Proc. 16th Annu. Conf. Int. Speech Commun. Assoc.*, Dresden, Germany, 2015, pp. 3586–3589.

[17] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002.

[18] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, Montreal, QC, Canada, 2014, pp. 2672–2680.

[19] K. Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y. Wang, "Generative adversarial networks: Introduction and outlook," *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 4, pp. 588–598, Sep. 2017.

[20] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An overview," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 53–65, Jan. 2018.

[21] J. Han, Z. Zhang, N. Cummins, and B. Schuller, "Adversarial training in affective computing and sentiment analysis: Recent advances and perspectives," *IEEE Comput. Intell. Mag.*, vol. 14, no. 2, pp. 68–82, May 2019.

[22] J. Han, Z. Zhang, Z. Ren, F. Ringeval, and B. Schuller, "Towards conditional adversarial training for predicting emotions from speech," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Calgary, AB, Canada, 2018, pp. 6822–6826.

[23] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proc. 4th Int. Conf. Learn. Represent.*, San Juan, Puerto Rico, 2016.

[24] L. Yu, W. Zhang, J. Wang, and Y. Yu, "SeqGAN: Sequence generative adversarial nets with policy gradient," in *Proc. 31st Conf. Assoc. Adv. Artif. Intell.*, San Francisco, CA, USA, 2017, pp. 2852–2858.

[25] A. Shrivastava, T. Pfister, O. Tuzel, J. Susskind, W. Wang, and R. Webb, "Learning from simulated and unsupervised images through adversarial training," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Honolulu, HI, USA, 2017, pp. 2242–2251.

[26] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," Dec. 2017, arXiv:1712.04621.

[27] A. Antoniou, A. Storkey, and H. Edwards, "Data augmentation generative adversarial networks," Mar. 2018, arXiv:1711.04340.

[28] X. Zhu, Y. Liu, Z. Qin, and J. Li, "Data augmentation in emotion classification using generative adversarial networks," Dec. 2017, arXiv:1711.00648.

[29] S. Pascual, A. Bonafonte, and J. Serrà, "SEGAN: Speech enhancement generative adversarial network," in *Proc. 18th Annu. Conf. Int. Speech Commun. Assoc.*, Stockholm, Sweden, 2017, pp. 3642–3646.

[30] D. Stoller, S. Ewert, and S. Dixon, "Adversarial semi-supervised audio source separation applied to singing voice extraction," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Calgary, AB, Canada, 2018, pp. 2391–2395.

[31] Z. Chen, C.-W. Wu, Y.-C. Lu, A. Lerch, and C.-T. Lu, "Learning to fuse music genres with generative adversarial dual learning," in *Proc. IEEE Int. Conf. Data Mining*, New Orleans, LA, USA, 2017, pp. 817–822.

[32] S. Sahu, R. Gupta, G. Sivaraman, W. AbdAlmageed, and C. Y. Espy-Wilson, "Adversarial auto-encoders for speech based emotion recognition," in *Proc. 18th Annu. Conf. Int. Speech Commun. Assoc.*, Stockholm, Sweden, 2017, pp. 1243–1247.

[33] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," Mar. 2017, arXiv:1701.07875.

[34] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," in *Proc. 2nd Int. Conf. Learn. Represent.*, Banff, AB, Canada, 2014.

[35] A. van den Oord, N. Kalchbrenner, and K. Kavukcuoglu, "Pixel recurrent neural networks," in *Proc. 33rd Int. Conf. Mach. Learn.*, New York, NY, USA, 2016, pp. 1747–1756.

[36] A. van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, and K. Kavukcuoglu, "Conditional image generation with PixelCNN decoders," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, Barcelona, Spain, 2016, pp. 4790–4798.

[37] M. Mirza and S. Osindero, "Conditional generative adversarial nets," Nov. 2014, arXiv:1411.1784.

[38] A. Odena, "Semi-supervised learning with generative adversarial networks," in *Proc. Int. Conf. Mach. Learn. Workshop Data-Efficient Mach. Learn.*, New York, NY, USA, 2016.

[39] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, Barcelona, Spain, 2016, pp. 2234–2242.

[40] K. Sricharan, R. Bala, M. Shreve, H. Ding, K. Saketh, and J. Sun, "Semi-supervised conditional GANs," Aug. 2017, arXiv:1708.05789.

[41] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein, "Unrolled generative adversarial networks," in *Proc. 5th Int. Conf. Learn. Represent.*, Toulon, France, 2016.

[42] Y. Wang, L. Zhang, and J. van de Weijer, "Ensembles of generative adversarial networks," in *Proc. Adv. Neural Inf. Process. Syst. Conf. Workshop Adversarial Training*, Barcelona, Spain, 2016.

[43] I. O. Tolstikhin, S. Gelly, O. Bousquet, C.-J. Simon-Gabriel, and B. Schölkopf, "Adagan: Boosting generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, Long Beach, CA, USA, 2017, pp. 5424–5433.

[44] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," in *Proc. Adv. Neural Inf. Process. Syst. Workshop Deep Learn. Represent. Learn.*, Montreal, QC, Canada, 2014.

[45] Z. Zhang, D. Liu, J. Han, and B. Schuller, "Learning audio sequence representations for acoustic event classification," Jul. 2017, arXiv:1707.08729.

[46] Z. Zhang, J. Han, K. Qian, and B. Schuller, "Evolving learning for analysing mood-related infant vocalisation," in *Proc. 19th Annu. Conf. Int. Speech Commun. Assoc.*, Hyderabad, India, 2018, pp. 142–146.

[47] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Annu. Conf. Neural Inf. Process. Syst.*, Montreal, QC, Canada, 2014, pp. 3104–3112.

[48] K. Qian, C. Janott, Z. Zhang, C. Heiser, and B. Schuller, "Wavelet features for classification of VOTE snore sounds," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Shanghai, China, 2016, pp. 221–225.

[49] E. J. Kezirian, W. Hohenhorst, and N. de Vries, "Drug-induced sleep endoscopy: The VOTE classification," *Eur. Arch. Otorhinolaryngol.*, vol. 268, no. 8, pp. 1233–1236, Aug. 2011.

[50] N. S. Hessel and N. de Vries, "Diagnostic work-up of socially unacceptable snoring," *Eur. Arch. Oto-Rhino-Laryngol.*, vol. 259, no. 3, pp. 158–161, Mar. 2002.

[51] B. Schuller *et al.*, "The INTERSPEECH 2016 computational paralinguistics challenge: Deception, sincerity & native language," in *Proc. 17th Annu. Conf. Int. Speech Commun. Assoc.*, San Francisco, CA, USA, 2016, pp. 2001–2005.

[52] F. Eyben, M. Wöllmer, and B. Schuller, "OpenSMILE: The Munich versatile and fast open-source audio feature extractor," in *Proc. 18th ACM Int. Conf. Multimedia*, Firenze, Italy, 2010, pp. 1459–1462.

[53] M. Schmitt and B. Schuller, "openXBOW—Introducing the Passau open-source crossmodal bag-of-words toolkit," *J. Mach. Learn. Res.*, vol. 18, no. 96, pp. 1–5, Oct. 2017.

[54] J. Wiens and E. S. Shenoy, "Machine learning for healthcare: On the verge of a major shift in healthcare epidemiology," *Clin. Infectious Dis.*, vol. 66, no. 1, pp. 149–153, Aug. 2017.

[55] F. Jiang *et al.*, "Artificial intelligence in healthcare: Past, present and future," *Stroke Vascular Neurol.*, vol. 2, Jun. 2017, Art. no. e000101.

[56] A. L. Beam and I. S. Kohane, "Big data and machine learning in health care," *J. Amer. Med. Assoc.*, vol. 319, no. 13, pp. 1317–1318, Apr. 2018.

[57] P. Tzirakis, G. Trigeorgis, M. A. Nicolaou, B. Schuller, and S. Zafeiriou, "End-to-end multimodal emotion recognition using deep neural networks," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 8, pp. 1301–1309, Dec. 2017.

[58] A. Rao, S. Yadav, and P. K. Ghosh, "A dual source-filter model of snore audio for snorer group classification," in *Proc. 18th Annu. Conf. Int. Speech Commun. Assoc.*, Stockholm, Sweden, 2017, pp. 3502–3506.

[59] T. L. Nwe, H. D. Tran, and B. Ma, "An integrated solution for snoring sound classification using Bhattacharyya distance based GMM supervectors with SVM, feature selection with random forest and spectrogram with CNN," in *Proc. 18th Annu. Conf. Int. Speech Commun. Assoc.*, Stockholm, Sweden, 2017, pp. 3467–3471.

[60] H. Kaya and A. A. Karpov, "Introducing weighted kernel classifiers for handling imbalanced paralinguistic corpora: Snoring, addressee and cold," in *Proc. 18th Annu. Conf. Int. Speech Commun. Assoc.*, Stockholm, Sweden, 2017, pp. 3527–3531.

[61] O. Press, A. Bar, B. Bogin, J. Berant, and L. Wolf, "Language generation with recurrent generative adversarial networks without pre-training," Jun. 2017, arXiv:1706.01399.

[62] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Proc. Adv. Neural Inf. Process. Syst.*, Montreal, QC, Canada, 2015, pp. 2980–2988.

[63] O. Fabius and J. R. van Amersfoort, "Variational recurrent auto-encoders," Dec. 2014, arXiv:1412.6581.