Speaker information modification in the VoicePrivacy 2020 toolchain

Pierre Champion[1]*, Denis Jouvet*[1]*, Anthony Larcher*[2]

[1]Université de Lorraine, CNRS, Inria, LORIA, F-54000 Nancy, France.
[2]Le Mans Université, LIUM, France

{pierre.champion, denis.jouvet}@inria.fr, anthony.larcher@univ-lemans.fr

Abstract

This paper presents a study of the baseline system of the VoicePrivacy 2020 challenge. This baseline relies on a voice conversion system that aims at separating speaker identity and linguistic contents for a given speech utterance. To generate an anonymized speech waveform, the neural acoustic model and neural waveform model use the related linguistic content together with a selected pseudo-speaker identity. The linguistic content is estimated using bottleneck features extracted from a triphone classifier while the speaker information is extracted then modified to target a pseudo-speaker identity in the x-vector's space. In this work, we first proposed to replace the triphone-based bottleneck features extractor that requires supervised training by an end-to-end Automatic Speech Recognition (ASR) system. In this framework, we explored the use of adversarial and semi-adversarial training to learn linguistic features while masking speaker information. Last, we explored several anonymization schemes to introspect which module benefits the most from the generated pseudo-speaker identities.

Index Terms: VoicePrivacy 2020 Challenge, Speaker anonymization, Speech recognition.

1. Introduction

In many applications, such as virtual assistants, speech signal is sent from the device to centralized servers in which data is collected, processed and stored. Recent regulations, e.g., the General Data Protection Regulation (GDPR) [1] in the EU, emphasize on privacy preservation and protection of personal data. As speech data can reflect both biological and behavioral characteristics of the speaker, it is qualified as personal data [2]. The VoicePrivacy challenge is one of the first attempt of the speech community to encourage research on this topic define the task and introduce metrics, datasets, and protocols.

Anonymization is performed to suppress the personally identifiable paralinguistic information from a speech utterance while maintaining the linguistic content. The task of the VoicePrivacy challenge [3] is to degrade automatic speaker verification (ASV) performance, by removing speaker identity as much as possible, while keeping the linguistic content intelligible. This task is also referred to as *speaker anonymization*[4] or *de-identification* [5].

Anonymization systems in the VoicePrivacy challenge should satisfy the following requirements:

- output a speech waveform;
- conceal the speaker identity;
- keep the linguistic content intelligible;
- modify the speech signal of a given speaker to always sound like a unique target pseudo-speaker.

The fourth requirement constraints the system to have a one-to-one mapping between the real speaker identities and a pseudo-speaker. Such system can be considered as a voice conversion system where the output speaker resides in a pseudonymized space. Traditional voice conversion techniques consist of two modules, a conversion function, and a vocoder [6]. The conversion function learns a translation between acoustic features of the source speaker and acoustic features of the target speaker. Then, the vocoder uses the resulting acoustic features to synthesize a speech waveform of the target speaker. Traditionally, those systems heavily rely on parallel corpora to train the conversion function [7]. The introduction of phoneme posteriorgrams (PPGs), obtained from a speaker-independent automatic speech recognition system, allows to train the voice conversion system without parallel data [8]. PPGs represent the articulation of speech sounds corresponding to spoken content and are supposed to be independent from the speaker identity. In the VoicePrivacy baseline system, bottleneck features are extracted from a triphone classifier. Triphone-based bottleneck features share similar properties with PPGs which is why they are often used for voice conversion [8, 4, 9]. One drawback of those features is that the reference used to train the extractor requires linguistic knowledge on the language.

Intermediate representations of ASR-related neural networks convey information about the linguistic content in a non-linearly compressed form. Several papers have discussed how to make them to more speaker-invariant or speaker-independent [10, 11]. While those techniques may improve the performance on the task for which the network is trained, the assumption that the extracted features contain less information about the speaker identity has not been proven. Other works have study how to extract anonymized representations using adversarial neural networks [12]. Experiments on the speech modality were reported in [13], where the authors conclude that for end-to-end ASR architecture, adversarial training dramatically reduces the speaker identification classification accuracy. However, this observation does not match with the speaker verification results. Thus, it can be assumed that adversarial training does not help the network to better mask the speaker's identity. To better remove personal information from the bottleneck representation of written text, a semi-adversarial approach that actively adapts the neural network against unwanted inferences was proposed in [14].

Contributions of this paper are threefold. First, as an alternative to the triphone classifier used in Voice Privacy's baseline, we propose to extract PPGs-like linguistic features from a deep encoder-decoder architecture trained for ASR without intermediate phonemic annotations. Such architecture only requires speech data and their transcription to be trained. We argue that the bottleneck features extracted from an end-to-end architecture share similar properties as phoneme based PPGs while removing the needs for expert knowledge on a specific language.

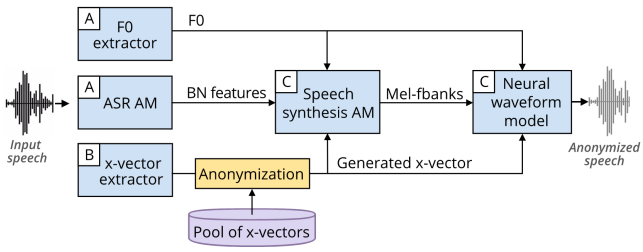

Figure 1: *The baseline speaker anonymization system.*

As the minimization of speaker information from PPGs-like features was discussed by the authors of the VoicePrivacy baseline architecture [4]. As the bottleneck features obtained via an end-to-end ASR system still carry residual information about the speaker's identity, we evaluate, as a second contribution, the ability of a semi-adversarial training to disentangle information conveyed by the speech utterances.

For our third contribution, we investigate how anonymization is performed in the toolchain by exploring several anonymization schemes. We introspect the generalization capability of some modules of baseline to a target pseudo-speaker.

In Section 2, we describe the VoicePrivacy baseline architecture. Section 3 explains our proposed enhancement over triphone classifier. Section 4 proposes an analysis on several anonymization schemes. Finally, Section 5 summarizes the key points and present possible extensions of this work.

2. The VoicePrivacy baseline system

The VoicePrivacy challenge provides two baseline systems: the *Baseline-1* that anonymizes speech utterances using x-vectors and neural waveform models [4] and the *Baseline-2* that performs anonymization using McAdams coefficient [15]. Our contributions are based on the *Baseline-1* which is referred to as the *baseline* system in the rest of this article.

The central idea of the baseline system, introduced in [4], is to separate speaker identity and linguistic content from an input speech utterance, assuming that those information can be disentangled. This assumption leads to the idea that an anonymized speech waveform can be obtained by altering only the feature that encodes the speaker's identity. The anonymization system illustrated in Figure 1 breaks down the anonymization procedure into three modules: the linguistic content extractors (*A*), the speaker identity extractor (*B*), the anonymization module, based on a pool of x-vectors, and the speech waveform synthesizer (*C*). For a given speech utterance, the system first extracts from the input waveform: an x-vector, the ASR bottleneck features, and the fundamental frequency (F0). Then, using knowledge gleaned from a pool of external speakers, a new x-vector is extracted for a pseudo-speaker. Eventually, the speech waveform is synthesized using the x-vector from the pseudo-speaker together with the original ASR bottleneck features, and the original F0 by using an acoustic model [3] and a neural waveform model [16].

ASR bottleneck features are used to encode the linguistic content of an utterance. They are extracted from the final layer of a factorized time delay neural network (TDNN-F) architecture [17, 18] trained to classify triphones. Those bottleneck features share similar properties with PPGs as they convey information about the spoken linguistic content in a space considered speaker-independent.

3. Proposed linguistic features

Traditional ASR systems are composed of an acoustic model (AM), a hand-designed pronunciation lexicon, and a language model (LM) [19]. These components are often manually designed and independently trained on different datasets. Acoustic models take acoustic features and predict a set of sound units, typically phonemes in context. Next, a pronunciation lexicon specifies how each word can be pronounced in terms of phonemes. Finally, the LM assigns probabilities to word sequences. Over the last few years, acoustic-to-word or end-to-end models directly targeting words or sub-word units[20, 21] have grown in popularity. Those models attempt at jointly learning the AM and the pronunciation lexicon. By nature, end-to-end architectures don't require linguistic expert knowledge to build systems, making ASR more accessible for under-resourced languages [22, 23]. As an alternative to the triphone-based acoustic model used in the VoicePrivacy baseline, we propose to use an end-to-end model that output sub-words as output units.

3.1. Model description

To obtain features without relying on a phonemes-classifier, we train an end-to-end deep encoder-decoder for ASR. This neural network is implemented using the ESPnet toolkit [24], and follows the hybrid architecture described in [25] with one encoder and two decoders. The first decoder is based on connectionist temporal classification (CTC) while the second one uses an attention mechanism. Input features are 80-dimensional mel-scale filterbank coefficients augmented with pitch feature. The encoder transforms a vector of 81 speech features into a vector of 256-dimensional continuous values. Work in [10] has shown that representations from deeper layers in networks are more robust to unwanted inferences, such as speaker identity. To keep the speaker information as low as possible, we chose a deep network rather than a shallow one. The encoder is composed of 3 bidirectional long short-term memory (BLSTM) layers with a hidden state size of 1024 features, followed by a fully connected layer. The encoder output is then fed to both CTC and attention-based decoders to predict the sequence of sub-words. The multi-objective learning function $\mathcal{L}_{\text{asr}}$, described in [25], is used to train the whole network without linguistic resources.

The bottleneck features used to represent the linguistic content are extracted from the encoder's output and used by the speech synthesis acoustic model to synthesize pseudo-anonymized mel-fbanks.

3.2. Semi-adversarial scheme

Works in [10, 13] have shown that adversarial training dramatically reduces the speaker identification accuracy. However, this observation does not transpose to the task of verification. One assumption is that the loss used to train the system targets identification performance but does not remove the speaker information. In this first attempt, the adversarial training does not help the network to better mask the speaker's identity. Recently, a "semi-adversarial" training scheme introduced in [14] has shown great success when applied on a written digit dataset. The authors succeeded in reducing the performance of an unwanted inference, i.e., the classification of fonts, from a digit classifier.

In our work, we defined the unwanted inference as speaker classification from extracted linguistic features. The adversary network is implemented following the x-vector architecture [26]

and takes as input the encoder's output vector. The goal of the adversary network is to minimize speaker classification error. The objective function $\mathcal{L}_{\text{spk}}$ corresponds to the cross entropy. With this framework, we applied semi-adversarial training to train the encoder for ASR while also masking speaker-related information. The adversarial objective function to minimize is the following:

$$\min_{\theta_e}\left[\min_{\theta_{\text{asr}}}\mathcal{L}_{\text{asr}}\left(\theta_e,\theta_{\text{asr}}\right)-\alpha\min_{\theta_{\text{spk}}}\mathcal{L}_{\text{spk}}\left(\theta_e,\theta_{\text{spk}}\right)\right]$$

where θ_e denotes the parameters of the encoder, θ_{asr} refers to the parameters of the ASR decoders, θ_{spk} the parameters of the speaker classifier, and α is a trade-off parameter between ASR and speaker loss, empirically set to 3.0.

As described in [14], training is performed in 3 steps:

- optimize the ASR network and speaker classifier with a α value of 0;
- optimize the encoder with the adversarial objective described above;
- optimize both ASR decoders and speaker classifier while freezing the encoder.

Step 2 and 3 can be repeated many times until $\mathcal{L}_{\text{spk}}$ does not evolve anymore.

3.3. Data recipes and training

First, we use the datasplits *train-clean-100* and *train-other-500* [27] provided by the VoicePrivacy challenge to train the ASR network without the speaker classifer. With a language model used to rescore the ASR hypotheses, the network obtains a 7.4 $\text{WER}_{\%}$ score on the LibriSpeech test clean partition. Then, we apply the semi-adversarial scheme on the network. As the speakers in the original LibriSpeech *train*/*dev*/*test* splits are disjoint. The original *train-clean-100* subset is split into three subsubsets to train and evaluate speaker identification on a closed-set of 251 speakers. The semi-adversarial scheme reduces the speaker accuracy from 11.93% down to 2.9%, while only penalizing $\text{WER}_{\%}$ by less than a 2.0 $\text{WER}_{\%}$ absolute increase. Note that retraining of both speech synthesis acoustic and neural waveform models is necessary to encounter for the new linguistic features.

3.4. Results and discussion

The performance of the proposed bottleneck features were evaluated by two objective metrics: speaker verification metric, and speech intelligibility. The metrics are computed using the shared evaluation tools of the VoicePrivacy challenge [3]. The challenge protocol imposes participants to evaluate their systems using shared ASR and ASV models trained on non-anonymized speech.

Table 1 and 2 compare the ASV and ASR performance of the proposed systems to the baseline system. The columns denoted as **ASR-Bn** correspond to the bottleneck features trained for ASR only, and the columns denoted as **ASR-Adv-Bn** correspond to the bottleneck features trained with the semi-adversarial scheme. The two columns **ASR-Bn** and **ASR-Adv-Bn** share similar results in both ASR and ASV metrics, this shows that masking speaker's information via semi-adversarial training does not help the VoicePrivacy toolchain to better generate anonymized speech according to speaker verification performance (given as EER). This similarity might be explained by multiple factors: speaker identification is not the same task as speaker verification, hence the semi-adversarial scheme isn't able to mask the information that speaker verification system uses; the speech synthesis AM used in the toolchain generates anonymized mel-fbanks with multiple inputs, F0, linguistic features, and x-vector. Leakage of speaker information can be found in two inputs, namely the F0 and the linguistic features. Masking speaker information from only one source might not be sufficient. Indeed, analysis of the fundamental frequency (F0), which is typically higher in female voices than in male voices, can be used as a simple gender classifier [28]. The author of the VoicePrivacy baseline argued that the F0 shouldn't be modified in order to preserve context-related information such as pitch, accents and intonation [29].

Bottleneck linguistic features obtained by an end-to-end network, trained to directly target sub-words as output units, are suitable to replace phoneme based PPGs. Table 2 shows that the ASR system using the proposed linguistic features doesn't perform as well as the baseline but $\text{WER}_{\%}$ stays reasonably low, and retraining the evaluation systems on anonymized training data allows to reduce the ASR performance drop. We note that the performance impact on both libri_test and vctk_test partitions are similar, with an absolute increase of 25% and 23% respectively.

A state-of-the-art x-vector speaker verification system is used to assess system's performance to hide the speakers identity. The threat model is as followed; an attacker gains access to speech data generated by the anonymization toolchain; he then attempts to verify speakers' identities, using either clean *original* enrollment speech and *anonymized* trial speech or anonymized enrollment and trial speech. The Speaker verification scores presented in table 1 correspond to the results of both *original* and *anonymized* enrollment scenarios. The baseline system performs well in the *original* enrollment scenario which is expected since the evaluation model was not trained on anonymized speech. Our proposed linguistic features perform equally well in this scenario. In the scenario where anonymized enrollment and trial data are used (row 3, 6, 9, 12, 15, and 18) our features show higher scores in terms of both EER and log-likelihood-ratio cost function C_{llr}, providing then a better privacy. The proposed linguistic features seem to have the highest impact on the male vctk test set (row 12) where a C_{llr}^{min} absolute improvement of 36% is observed. From informal listening tests, it appears that the voices produced are more saturated than when using the baseline system. The provided privacy gain observed might come from this aspect, further analysis is on-going.

4. Analysis of the pseudo-speaker

In this section, we do not investigate the strategy used to generate the pseudo-speaker x-vectors, but we focus on the impact that the x-vector information has on the modules used to generate the speech waveform: the speech synthesis AM and the neural waveform model (cf. modules *C*, Fig. 1). In order to understand which module benefits the most from the generated pseudo-speaker identities, we explore three additional anonymization schemes, all of them based on the baseline system. First, we replace the anonymized x-vector provided to the neural waveform model, by the original x-vector. Then for each combination of x-vector and model we generated the corresponding speech waveform. Results are reported in Table 3. As scores were identical in each individual test sets, the $\text{WER}_{\%}$ and $\text{EER}_{\%}$ values reported in this table are averaged values. Interestingly, our results show that the neural waveform model does not benefit from the anonymization of the x-vector. Af-

Table 1: *Speaker verification results for **Baseline-1**, **ASR-Bn** and **ASR-Adv-Bn** on test partitions (**o** – original, **a** – anonymized speech data for enrollment and trial parts; **Gen** denotes speaker gender: **f** – female, **m** – male).*

#	Enroll	Trial	Gen	Test set	Baseline-1			ASR-Bn			ASR-Adv-Bn		
					$\mathbf{EER}_{\%}$	$\mathbf{C}_{llr}^{min}$	$\mathbf{C}_{llr}$	$\mathbf{EER}_{\%}$	$\mathbf{C}_{llr}^{min}$	$\mathbf{C}_{llr}$	$\mathbf{EER}_{\%}$	$\mathbf{C}_{llr}^{min}$	$\mathbf{C}_{llr}$
1	o	o	f	libri_test	8	0.18	27						
2	o	a	f	libri_test	49	1.00	151	51	1.00	156	52	1.00	155
3	a	a	f	libri_test	30	0.80	14	31	0.84	30	30	0.81	29
4	o	o	m	libri_test	1	0.04	15						
5	o	a	m	libri_test	53	1.00	167	55	1.00	159	55	1.00	160
6	a	a	m	libri_test	33	0.83	27	32	0.84	42	32	0.82	38
7	o	o	f	vctk_test_com	3	0.09	1						
8	o	a	f	vctk_test_com	50	0.99	158	51	1.00	177	52	1.00	172
9	a	a	f	vctk_test_com	31	0.83	9	**34**	**0.89**	24	**34**	**0.88**	24
10	o	o	m	vctk_test_com	1	0.04	1						
11	o	a	m	vctk_test_com	56	1.00	189	54	1.00	171	53	1.00	173
12	a	a	m	vctk_test_com	22	0.66	14	**36**	**0.90**	35	**37**	**0.88**	33
13	o	o	f	vctk_test_dif	5	0.17	1						
14	o	a	f	vctk_test_dif	49	1.00	142	53	1.00	157	53	1.00	152
15	a	a	f	vctk_test_dif	**34**	**0.88**	12	29	0.81	30	30	0.82	31
16	o	o	m	vctk_test_dif	2	0.07	2						
17	o	a	m	vctk_test_dif	54	1.00	166	56	1.00	166	56	1.00	169
18	a	a	m	vctk_test_dif	26	0.74	16	**32**	**0.86**	46	**31**	**0.86**	39

Table 2: *Speech recognition results for **Baseline-1**, **ASR-Bn** and **ASR-Adv-Bn** in terms of $WER_{\%}$ for test data (**o** – original, **a** – anonymized speech data) for two trigram language models, a large one $\mathbf{LM}_l$ and a small one $\mathbf{LM}_s$*

#	Test set	Data	Baseline-1		ASR-Bn		ASR-Adv-Bn	
			$\mathbf{LM}_s$	$\mathbf{LM}_l$	$\mathbf{LM}_s$	$\mathbf{LM}_l$	$\mathbf{LM}_s$	$\mathbf{LM}_l$
1	libri_test	o	5.55	4.14				
2	libri_test	a	**9.06**	**6.77**	11.22	8.52	11.08	8.42
3	vctk_test	o	16.39	12.81				
4	vctk_test	a	**19.24**	**15.53**	22.85	19.11	22.88	19.19

Table 3: *Influence of the selected x-vector on the generated speech (**orig** – original x-vector, **anon** – anonymized x-vector). ASR results in terms of $WER_{\%}$ on vctk test with LM_l, ASV results are averaged over test partitions of the same attack scenario.*

#	x-vector		$\mathbf{EER}_{\%}$ Enroll: original speech Trial: anonymized speech	$\mathbf{EER}_{\%}$ Enroll: anonymized speech Trial: anonymized speech	$\mathbf{WER}_{\%}$
	Speech synthesis acoustic model	Neural waveform model			
1	anon	anon	52	31	15.53
2	anon	orig	52	32	15.32
3	orig	anon	22	15	14.64
4	orig	orig	22	15	14.63

ter an informal listening of the transformed data, it appears that speech generated by methods 1 and 2 is perceived as being uttered by the same speaker; and speech generated by methods 3 and 4 appears as being uttered by another speaker. It is worth noting that copy synthesis (row 4) shows a high EER score of 22 in the *original to anonymized* attack, exhibiting a possible limitation of the toolchain.

5. Conclusions and future work

In this work, we proposed to extract bottleneck features using an end-to-end Automatic Speech Recognition (ASR) system to replace the triphone-based bottleneck features extractor. Our results show that such bottleneck representations can be used by speaker anonymization systems. Features extracted from an end-to-end architecture share similar properties as phoneme based PPGs while removing the need of expert knowledge on a specific language. Besides, it offers a small privacy improvement when compared to the baseline system, at the cost of degraded intelligibility. We also showed that the semi-adversarial training scheme does not help the system to produce better anonymized speech. We hypothesize that this deficiency might be attributed to the disparity of sensitive information that speech encapsulates. Lastly, the experimental analysis of several anonymization schemes, to introspect which module benefits the most from the generated pseudo-speaker identities, demonstrates that the neural waveform model does not use the generated pseudo-speaker information. The toolchain seems to only rely on the speech synthesis AM to perform speaker anonymization. We hypothesize that this behavior comes from the repetition of speaker information in x-vector and mel-fbanks.

6. Acknowledgements

This work was supported in part by the French National Research Agency under project DEEP-PRIVACY (ANR-18-CE23-0018) and Région Lorraine. Experiments were carried out using the Grid'5000 testbed, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations.

7. References

[1] E. Parliament and Council, "Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec," *General Data Protection Regulation*, 2016.

[2] A. Nautsch, C. Jasserand, E. Kindt, M. Todisco, I. Trancoso, and N. Evans, "The GDPR & Speech Data: Reflections of Legal and Technology Communities, First Steps Towards a Common Understanding," in *Proc. Interspeech 2019*, 2019, pp. 3695–3699.

[3] N. Tomashenko, B. M. L. Srivastava, X. Wang, E. Vincent, A. Nautsch, J. Yamagishi, N. Evans, J. Patino, J.-F. Bonastre, P.-G. Noé, and M. Todisco, "Introducing the VoicePrivacy Initiative," *ArXiv*, 2020.

[4] F. Fang, X. Wang, J. Yamagishi, I. Echizen, M. Todisco, N. Evans, and J.-F. Bonastre, "Speaker Anonymization Using X-vector and Neural Waveform Models," in *Proc. 10th ISCA Speech Synthesis Workshop*, 2019, pp. 155–160.

[5] C. Magariños, P. Lopez-Otero, L. Docio-Fernandez, E. Rodriguez-Banga, D. Erro, and C. Garcia-Mateo, "Reversible speaker de-identification using pre-trained transformation functions," *Computer Speech & Language*, vol. 46, pp. 36–52, 2017.

[6] S. H. Mohammadi and A. Kain, "An overview of voice conversion systems," *Speech Communication*, vol. 88, pp. 65 – 82, 2017.

[7] Z. Wu, T. Virtanen, E. S. Chng, S. Member, and H. Li, "Exemplar-based sparse representation with residual compensation for voice conversion," in *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP*, 2014, pp. 1506–1521.

[8] L. Sun, K. Li, H. Wang, S. Kang, and H. Meng, "Phonetic posteriorgrams for many-to-one voice conversion without parallel data training," in *IEEE International Conference on Multimedia and Expo (ICME)*, 2016, pp. 1–6.

[9] Y. Saito, Y. Ijima, K. Nishida, and S. Takamichi, "Non-parallel voice conversion using variational autoencoders conditioned by phonetic posteriorgrams and d-vectors," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5274–5278.

[10] Y. Adi, N. Zeghidour, R. Collobert, N. Usunier, V. Liptchinsky, and G. Synnaeve, "To reverse the gradient or not: an empirical comparison of adversarial and multi-task learning in speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3742–3746.

[11] T. Tsuchiya, N. Tawara, T. Ogawa, and T. Kobayashi, "Speaker invariant feature extraction for zero-resource languages with adversarial learning," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 2381–2385.

[12] C. Feutry, P. Piantanida, Y. Bengio, and P. Duhamel, "Learning anonymized representations with adversarial neural networks," *ArXiv*, vol. abs/1802.09386, 2018.

[13] B. M. L. Srivastava, A. Bellet, M. Tommasi, and E. Vincent, "Privacy-Preserving Adversarial Representation Learning in ASR: Reality or Illusion?" in *Proc. Interspeech 2019*, 2019, pp. 3700–3704.

[14] T. Ryffel, D. Pointcheval, F. Bach, E. Dufour-Sans, and R. Gay, "Partially encrypted deep learning using functional encryption," in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Eds. Curran Associates, Inc., 2019, pp. 4517–4528.

[15] S. McAdams, "Spectral fusion, spectral parsing and the formation of the auditory image," *Ph. D. Thesis, Stanford*, 1984.

[16] X. Wang and J. Yamagishi, "Neural Harmonic-plus-Noise Waveform Model with Trainable Maximum Voice Frequency for Text-to-Speech Synthesis," in *Proc. 10th ISCA Speech Synthesis Workshop*, 2019, pp. 1–6.

[17] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proc. Interspeech 2015*, 2015, pp. 3214–3218.

[18] D. Povey, G. Cheng, Y. Wang, K. Li, H. Xu, M. Yarmohammadi, and S. Khudanpur, "Semi-orthogonal low-rank matrix factorization for deep neural networks," in *Proc. Interspeech 2018*, 2018, pp. 3743–3747.

[19] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Vesel, "The kaldi speech recognition toolkit," *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, 01 2011.

[20] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Association for Computational Linguistics*, 2016, pp. 1715–1725.

[21] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates," in *Association for Computational Linguistics*, 2018, pp. 66–75.

[22] C.-X. Qin, D. Qu, and L.-H. Zhang, "Towards end-to-end speech recognition with transfer learning," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2018, 12 2018.

[23] L. Samarakoon, B. Mak, and A. Y. S. Lam, "Domain adaptation of end-to-end speech recognition in low-resource settings," in *IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 382–388.

[24] S. Watanabe, T. Hori, S. Karita, T. Hayashi, J. Nishitoba, Y. Unno, N. Enrique Yalta Soplin, J. Heymann, M. Wiesner, N. Chen, A. Renduchintala, and T. Ochiai, "Espnet: End-to-end speech processing toolkit," in *Proc. Interspeech 2018*, 2018, pp. 2207–2211.

[25] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid ctc/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.

[26] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018.

[27] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5206–5210.

[28] J. SAS and A. SAS, "Gender recognition using neural networks and asr techniques," *Journal of MIT*, vol. 22, pp. 179–187, 2013.

[29] C. Gussenhoven, *Pitch in Language I: Stress and Intonation*, ser. Research Surveys in Linguistics. Cambridge University Press, 2004, p. 12–25.