Adversarial Disentanglement of Speaker Representation for Attribute-Driven Privacy Preservation

Paul-Gauthier Noé, Mohammad Mohammadamini, Driss Matrouf, Titouan Parcollet, Jean-François Bonastre

Adversarial Disentanglement of Speaker Representation for Attribute-Driven Privacy Preservation

Paul-Gauthier Noé[1], *Mohammad MohammadAmini*[1], *Driss Matrouf*[1], *Titouan Parcollet*[1,2], *Jean-François Bonastre*[1]

[1]Laboratoire Informatique d'Avignon (LIA), Avignon Université, France
[2]University of Cambridge, United-Kingdom
paul-gauthier.noe@univ-avignon.fr

Abstract

With the increasing interest over speech technologies, numerous Automatic Speaker Verification (ASV) systems are employed to perform person identification. In the latter context, the systems rely on neural embeddings as a speaker representation. Nonetheless, such representations may contain privacy sensitive information about the speakers (e.g. age, sex, ethnicity, ...). In this paper, we introduce the concept of *attribute-driven privacy preservation* that enables a person to hide one or a few personal aspects to the authentication component. As a first solution we define an adversarial autoencoding method that disentangles a given speaker attribute from its neural representation. The proposed approach is assessed with a focus on the sex attribute. Experiments carried out using the VoxCeleb data sets have shown that the defined model enables the manipulation (i.e. variation or hiding) of this attribute while preserving good ASV performance.

Index Terms: Speaker representation, adversarial disentanglement, sex-independent, attribute-driven privacy preservation

1. Introduction

Privacy concerns of speech technologies are at the heart of recent debates. With the raise of awareness toward this issue, people are increasingly advocating for privacy preserving technologies and an amplified control over their data. In the last few years, the performance of Automatic Speaker Verification (ASV) and automatic language recognition systems have been significantly improved with the introduction of the statistical i-vector approach [1], and more recently with the introduction of the neural embedding based x-vector method [2, 3, 4]. Nonetheless, and despite large-scale applications, it remains difficult to properly identify the nature of the information encoded in such neural embeddings. As an example, it has been shown that x-vectors contain privacy sensitive information that may be irrelevant for a speaker recognition task [5].

We propose to consider the disentanglement learning as a solution to facilitate the exploration of x-vectors. Indeed, it has been shown that distentangled representation allows a more interpretable representation that emphasises on independent explanatory factors [6]. Therefore, manipulations to alter or even hide speaker information might be easier with disentangled speaker representations.

As an example, and in the specific context of speaker authentication from speech, one may want to control the information s/he agrees to provide. Indeed, the voice contains many sensitive data such as the sex[1], age, ethnicity, education level... Unfortunately, most of speaker privacy preserving technologies encrypts or changes the entire speaker identity [8, 9, 10, 11], while some use-cases may require hiding only a few personal aspects and keeping all the remaining voice richness. To overcome this drawback, we define the concept of *attribute-driven privacy preservation* enabling one to hide selected personal aspects to the authentication side.

Various works have been proposed on linguistic/non-linguistic [12, 13, 14, 15] and speaker/noise information separation [16]. In [17] an adversarial approach is used in a end-to-end manner to hide the speaker's identity in Automatic Speech Recognition (ASR) representations. Most of the investigations on disentanglement methods are conduced directly on acoustic feature sequences, and few works have been proposed to operate on speaker representations. In [18] an autoencoder is applied on x-vectors to disentangle the speaking style and the identity of the speaker with two latent subspaces. More recently, [19] have used an unsupervised disentanglement method over x-vectors to separate the speaker's discriminating information and the remaining noises in the context of robust speaker recognition. However, none of the previous works considered to disentangle speakers or soft-biometric attributes directly from speaker representations.

In this paper, we present an approach to disentangle speaker attributes directly from x-vector embeddings using an adversarial autoencoding approach. Our goal is to allow everyone to modify or hide a specific personal attribute in the x-vector while preserving the remaining information in order to perform a targeted task. The use of the autoencoder paradigm allows us to remain in the x-vector space, thus allowing easy combinations with all systems using this representation. This article proposes a first solution to the *attribute-driven privacy preservation* concept, capable of managing the male/female attribute in x-vector representation. Its effectiveness is assessed on sex disentanglement evaluation and speaker verification tasks using the VoxCeleb corpora [20, 21]. The rest of this paper is organised as:

1. Section 2 presents our adversarial disentangling autoencoder[2] to operate over x-vectors,
2. Section 3 presents the experimental setup. Both disentanglement and ASV evaluations are detailed,
3. The results on the VoxCeleb corpora are then presented and discussed in Section 4,
4. Finally, Section 5 concludes and presents possible future works.

[1]Throughout this paper, the term *sex* refers to the biological differences between female and male [7].

[2]Code is available at https://github.com/LIAvignon/adversarial-disentangling-autoencoder-for-spk-representation

2. Adversarial Disentangling Autoencoder

The proposed model is similar to the one used in [22] for image processing. Actually, our system disentangles the sex information in x-vector based speaker representations. It is composed with four components: a pre-trained sex discriminator, an encoder, a decoder and an adversarial sex discriminator.

Notation. Let $D = \{(x_1, y_1, \tilde{y}_1), ...(x_m, y_m, \tilde{y}_m)\}$ be a set of x-vectors with their corresponding binary sex labels (i.e. 0 for male, 1 for female) and as soft labels, their posterior probability of being a female $\tilde{y} = P(F|x)$ with F the hypothesis giving the speech segment as being from a female speaker.

The pre-trained sex discriminator is a fixed pre-trained sex classifier used to extract the posteriors $\tilde{y}$ as soft labels. Soft labelling is used instead of hard labelling to make the *sex* variable continuous and to allow interpolation.

The encoder encodes an input x-vector x into a latent variable z. In addition to compressing the input information, it tries to cheat the adversarial sex discriminator.

The decoder reconstructs the original x-vector x from the latent variable z and a condition w (during the training $w = \tilde{y}$).

The adversarial sex discriminator tries to predict the *sex class* of the speech segment from the encoded vector z.

The training and testing procedure of the model are given in Figure 1. The sex information is disentangled from the rest in the latent space with an adversarial training that opposes the encoder-decoder to the adversarial sex discriminator. Thus, a first optimiser updates the neural parameters θ_d of the adversarial sex discriminator with respect to the correct expected class of x. Then, a second optimiser updates the parameters θ_E of the encoder and the parameters θ_D of the decoder in order to jointly cheat the adversarial sex discriminator and propose a good reconstruction of x. Therefore, the encoder-decoder and the sex discriminator are trained in an adversarial manner in order to make the encoded vector z sex independent. The two objective functions are:

$$L_d(\theta_d|\theta_E) = -\frac{1}{m}\sum_{(x,y)\in D} log(P_{\theta_d}(y|z)), \quad (1)$$

for the first optimiser, and:

$$L(\theta_E, \theta_D|\theta_d) = \frac{1}{m}\sum_{(x,y)\in D} r(\hat{x}, x) - log(P_{\theta_d}(1-y|z)), \quad (2)$$

for the second one, with $z = E_{\theta_E}(x)$ and $\hat{x} = D_{\theta_D}(z, \tilde{y})$ the output of the encoder and the decoder respectively, and $r(\hat{x}, x) = 1 - \frac{<\hat{x},x>}{||\hat{x}||||x||}$ the reconstruction error. During our experiments, both optimisers follow the Stochastic Gradient Descent (SGD) with the model parameters $\theta_E^{(t)}$, $\theta_D^{(t)}$ and $\theta_d^{(t)}$ being updated at each time step t as follow:

$$\theta_d^{(t+1)} = \theta_d^{(t)} \quad -\eta\nabla_{\theta_d} L_d(\theta_d^{(t)}|\theta_E^{(t)}, x^{(t)}, y^{(t)}), \quad (3)$$

$$\begin{bmatrix}\theta_E^{(t+1)} \\ \theta_D^{(t+1)}\end{bmatrix} = \begin{bmatrix}\theta_E^{(t)} \\ \theta_D^{(t)}\end{bmatrix} - \eta\nabla_{\theta_E,\theta_D} L(\theta_E^{(t)}, \theta_D^{(t)}|\theta_d^{(t)}, x^{(t)}, y^{(t)}), \quad (4)$$

with $x^{(t)}$ and $y^{(t)}$ the training sample and its corresponding class at time t.

3. Experimental and Evaluation Setup

This section first details the architecture and the training procedure of the proposed model (Section 3.1). Then, the corpora used to train and evaluate the model are presented (Section 3.2). Finally, evaluation metrics and protocols are described (Section 3.3).

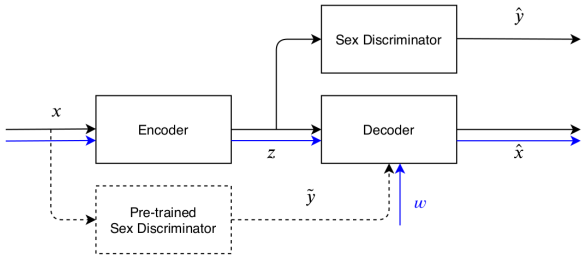

Figure 1: *Illustration of the Adversarial Disentangling Autoencoder. The black arrows illustrate the forward flow during the training phase. The dashed lines represent the part that is not updated during training: the pre-trained sex discriminator is used to extract soft labels for the decoder. The blue arrows illustrate the forward flow during the testing phase. Sex attribute manipulation is done by changing w on the decoder.*

3.1. Model Architecture and Training

First, a standardisation and a length normalisation [23] are applied on the x-vectors before being fed into the model. The encoder consists of a single dense layer with ReLU activation functions [24]. Batch-normalisation [25] is applied accross the resulting 128 dimensional representation z. Then, the decoder takes the encoded vector z concatenated with the sex information encoded as the soft variable $\tilde{y}$ as input, and consists of a single dense layer with a hyperbolic tangent for the activation function followed by a last length normalization. The adversarial discriminator is composed with two dense layers. The first one has 64 units with ReLU activation functions, and the second one has 1 unit with a sigmoid activation function. Both layers use dropout with a 0.3 probability. The model is trained using two standard stochastic gradient descents. Learning rates are set to 10^{-4} and the momentums are 0.9.

3.2. Corpora

Table 1 presents the data used for training and testing the adversarial disentangling autoencoder.

	Number of segments			Number of speakers		
	V1D	V2D	V2T	V1D	V2D	V2T
Male	61616	397032	9120	665	3682	81
Female	61616	397032	9120	546	2312	39

Table 1: *Description of the data sets.*

A subset (V1D) of VoxCeleb [20] is used to train the external sex discriminator used to extract the posterior probabilities $\tilde{y} = P(F|x)$ as soft labels. A subset (V2D) of VoxCeleb2 development part [21] is used as a training set for the adversarial disentangling autoencoder model. A subset (V2T) of VoxCeleb2 testing part is used to test the disentanglement ability. We use subsets instead of the full datasets to obtain a proper balance between the two sex classes.

	Accuracy [%]		Absolute difference between accuracy and randomness		C_{llr}^{min} [10^{-2}]	
Condition w	V2D	V2T	V2D	V2T	V2D	V2T
$w \sim N(\frac{1}{2}, 0.01)$	**50.6**	**53.5**	**0.6**	**3.5**	**99.97**	**99.69**
$w \sim C_{m_0,m_1}(\frac{1}{2})$	**50.1**	**49.7**	**0.1**	**0.3**	**99.99**	**99.99**
$w = 1 - \tilde{y}$	2.3	2	47.7	48	13.56	9.02
$w = 1 - y$	0	0	50	50	0	0
$w = \tilde{y}$	92.4	93.7	42.4	43.7	15.82	21.62
$w = y$	100	100	50	50	0	0
Original vector x	91.6	96.2	41.6	46.2	16.76	14.19

Table 2: *Effects of the manipulation of w on the sex classification results for a subset of V2D (2×5000 samples) and on V2T. The absolute difference between the accuracy and 50% is given in order to quantify the closeness of the prediction to randomness. The C_{llr}^{min} is also provided as an application-independent discrimination quality measurement.*

3.3. Evaluation Protocol

Once trained, the condition w (Figure 1) allows an easy manipulation of the sex-information in a x-vector. To evaluate the ability of the model to hide the sex information contained in a x-vector, we first propose to sample w from a normal distribution centred around the value 0.5 ($w \sim N(\frac{1}{2}, 0.01)$) and then, to sample w from a categorical distribution that produces either m_0 or m_1 with an equal probability 0.5 ($w \sim C_{m_0,m_1}(\frac{1}{2})$) with $m_0 = 0.18$ and $m_1 = 0.70$ the means of a two gaussians mixture model trained on the posteriors $P(F|x)$ of V2D. These samplings are used to make the reconstructed x-vectors *sex-independent*. In addition, we propose to try $w = 1 - \tilde{y}$ and $w = 1 - y$ to check the manipulation abilities of the sex information. More precisely, we investigate both the case of sex interchanging (i.e. $w = 1 - \tilde{y}$ and $w = 1 - y$) and sex preserving (i.e. $w = \tilde{y}$ and $w = y$) to validate the quality of the reconstruction when the original sex is unchanged.

Disentanglement evaluation with classifier behaviour.
The effects of the transformed x-vectors on the pre-trained sex discriminator are analysed in order to evaluate the disentanglement capacity and the ability of the model to hide a specific attribute. The accuracy and its absolute difference with 50% are measured to derive the closeness of the predictions with respect to randomness. The C_{llr}^{min} is also computed in order to provide an application-independent discrimination quality measurement [26][3].

Disentanglement evaluation with mutual information.
The Mutual Information (MI) can also be used as an information theoretic privacy measure [27]. In our experiments, MI is used to measure the dependence between the transformed x-vectors and the sex class variable y. Thus, MI between the x-vectors components and y are estimated [28, 29] and the obtained MI are added up over the x-vector dimensions providing a measure on *how much x-vectors are sharing information with the sex class variable*.

Automatic speaker verification evaluation.
An important aim of this work is to hide the sex information of a speaker in a x-vector representation while maintaining good automatic speaker verification results. Thus, ASV performances on the transformed x-vectors are measured. The commonly used Probabilistic Linear Discriminant Analysis (PLDA) [30] is used for comparing enrolment and probe x-vectors.

[3]Here, the calibration transformation is affine and allows both monotonic decreasing and monotonic increasing mappings.

	Number of segments		Number of speakers	
	Enrolment	Test	Enrolment	Test
Male	11282	11277	81	81
Female	4558	4562	39	39

Table 3: *Description of the ASV data sets.*

It is worth noticing that the reference x-vectors provided by the user to the authentication side beforehand should also be converted. Indeed, there is no interest in not converting them because the authentication side would be able to match a converted probe x-vector with non-converted reference ones. In this case, the authentication side will be able to get the sex class of the speaker from the reference x-vectors. Therefore, we only consider the case in which both enrolment and probe x-vectors are converted. The Equal Error Rate (EER) and the C_{llr}^{min} are measured to assess the ASV[4]. The PLDA have been trained on $200,000$ x-vectors randomly chosen from the VoxCeleb 1 and 2 training subsets. Their dimension is reduced to 128 with a linear discriminant analysis. The description of the enrolment and test sets are shown in Table 3.

4. Results

This section presents the results of the disentanglement evaluation (Section 4.1) and the ASV performances on the resulting *sex-independent* x-vectors (Section 4.2).

4.1. Disentanglement Evaluation

First, the results of the evaluation based on the sex classifier behaviour are discussed. Then we present the results obtained with the evaluation based on the MI measurement.

The results of the disentanglement evaluation based on the sex classifier behaviour are shown in Table 2. Let us point out that if w is random (i.e. $w \sim N(\frac{1}{2}, 0.01)$ and $w \sim C_{m_0,m_1}(\frac{1}{2})$) the classifier is not able to predict correctly the sex and exhibits random predictions. However, with w being sampled from the categorical distribution, few males will be converted to females and vice versa. Whereas if w is sampled from the normal distribution, the transformation would drive the x-vectors closer to a representation identified as a *neutral-sex* x-vector. In the latter scenario, the predictor becomes confused and uncertain as depicted with the Log-Likelihood-Ratio (LLR) distributions given in Figure 2. The accuracies close to zero in Table 2 shows that we can manipulate w in order to cheat the sex predictor by taking $w = 1 - \tilde{y}$ or $w = 1 - y$. Indeed, with such a manipulation,

[4]Here, scores are calibrated using the *pool-adjacent-violators* algorithm [31].

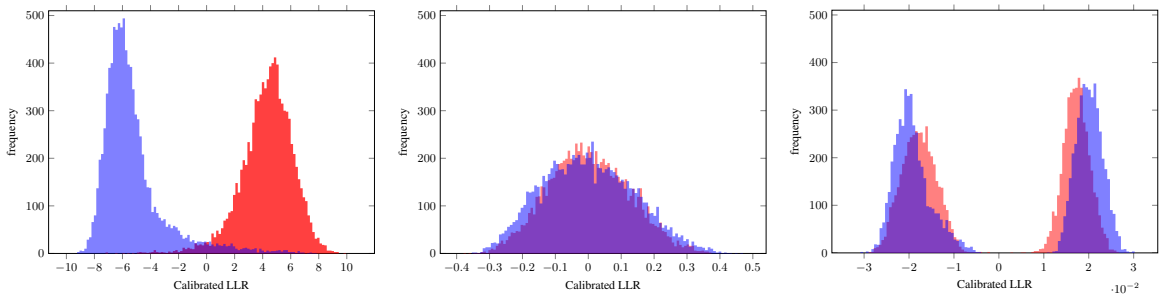

Figure 2: *Histograms of the calibrated*[3] *Log-Likelihood-Ratios on the sex classification with V2T original (left), converted with* $w \sim N(\frac{1}{2}, 0.01)$ *(middle) and converted with* $w \sim C_{m_0,m_1}(\frac{1}{2})$ *(right). Females are in blue and males are in red.*

one can make the classifier predict the opposite sex class. It is also worth noticing that hard conditioning (i.e. using the hard labeling y) forces the predicted class.

The results of the disentanglement evaluation based on the mutual information are shown in Table 4. The comparison between the results with the condition w replaced by the two kinds of noises with the results on the original x-vectors highlight a significant decrease of MI, thus illustrating that the x-vectors tends to be more *sex-independent*. Conversely, we observe a significant increase of MI when hard labels are injected in the decoder (i.e. $w = y$ and $w = 1 - y$). Such a behavior is easily explained by the fact that the sex class variable y used to compute the MI is equal to the one injected into the decoder.

	$I(\hat{x}, y)$ [bits]	
Condition w	V2D	V2T
$w \sim N(\frac{1}{2}, 0.01)$	**4.83**	**9.81**
$w \sim C_{m_0,m_1}(\frac{1}{2})$	**4.17**	**11.0**
$w = 1 - \tilde{y}$	105.62	99.3
$w = 1 - y$	175.6	180.9
$w = \tilde{y}$	107.2	108.2
$w = y$	173.6	182.6
Original vector x	96.1	111.8

Table 4: *Mutual information measurements between the x-vectors and the sex class variable* y *on a subset of V2D (*2×5000 *samples) and on V2T.*

In addition we propose to visualise few principal components of the original data and its converted version with $w \sim N(\frac{1}{2}, 0.01)$. In order to do so, one Principal Component Analysis (PCA) is trained on a subset of V2D and another on its converted version. As the sex information is not necessarily embedded in the first components, we visualise the components which have the highest mutual information with the sex class variable y. More precisely the first and the third components for the original data and the 86th and 95th components for the converted one. The corresponding planes are shown in Figure 3 and 4 respectively. One can notice the absence of sex clusters for the converted data which confirms the ability to hide the sex information[5].

The reported results confirm our initial intuition that the proposed method provides control over the sex attribute contained on a x-vector.

[5]It might be confusing to visualise different couples of components. However, we have to be aware that if most of the sex information is originally embedded in specific components, they might not be the same anymore for the converted x-vectors.

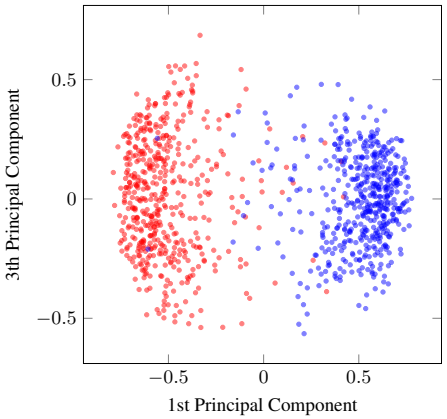

Figure 3: *First and third (i.e. the ones with the highest MI with the sex class variable) principal components of original x-vectors from a subset of V2D (*2×5000 *samples). The PCA have been trained on this subset. Females are in blue and males are in red.*

4.2. Automatic Speaker Verification Evaluation

Table 5 presents the results obtained for the ASV task on the transformed x-vectors. First, in the case of $w = \tilde{y}$, it is clear that the obtained reconstruction is of sufficient quality to ensure good ASV performences with a 2.2% EER. Then, it is important to highlight convincing performances (5.6% EER) while the x-vectors are *sex-independent* based on a sex attribute sampled from a Gaussian noise ($w \sim N(\frac{1}{2}, 0.01)$).

Condition w	EER [%]	C_{llr}^{min}
$w \sim N(\frac{1}{2}, 0.01)$	5.6	0.205
$w \sim C_{m_0,m_1}(\frac{1}{2})$	15.8	0.478
$w = \tilde{y}$	2.2	0.091
Original vector x	1.7	0.067

Table 5: *Effects of the manipulation of* w *on the automatic speaker verification. In addition to the application-independent cost* C_{llr}^{min}*, the Equal Error Rate (EER) is also provided.*

Interestingly, an important increase of EER is obtained with $w \sim C_{m_0,m_1}(\frac{1}{2})$. Such a decrease of performances is due to the fact that the average distance between the original soft label and the sampled w is bigger than with $w \sim N(\frac{1}{2}, 0.01)$. Therefore, in the former case, the resulting x-vectors will be in

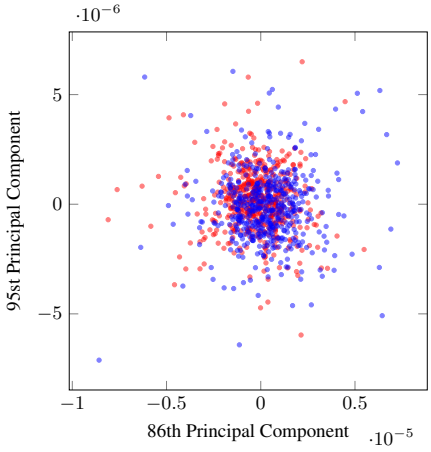

Figure 4: *86th and 95th (i.e. the ones with the highest MI with the sex class variable) principal components of the converted ($w \sim N(\frac{1}{2}, 0.01)$) x-vectors from a subset of V2D (2×5000 samples). The PCA have been trained on this subset. Females are in blue and males are in red.*

average *farther* from the original ones than in the latter scenario. Nonetheless, the conduced experiments have shown that the sex information contained in a x-vector speaker representation can be efficiently reduced with our approach while preserving good speaker verification capabilities.

5. Conclusion

In this paper we have introduced *attribute-driven privacy preservation* as the idea of enabling a speaker to hide only a few personal aspects in their representation while maintaining the remaining particularities unaltered. As a first solution, we proposed and released an adversarial autoencoding approach that disentangles the sex information from the rest in a x-vector neural embedding. This method is based on an encoder-decoder architecture combined with an additional discriminator that tries to predict the sex class from the encoded representation. Both are trained in an adversarial manner to make the encoded representation *sex independent*. Through the experiments conduced on the Voxceleb dataset, it has been shown that the sex variable used as an additional condition to the decoder enables the manipulation or hiding of the sex attribute in the speaker representation space while preserving good automatic speaker verification performances.

Even though the proposed approach reduces the amount of sex information contained in a x-vector, it is not guaranteed that an attacker in possession of the transformed data and their original sex class can not take advantage of the remaining sex information and train a new predictor that will detect the original sex. This work have been done on the sex attribute as an initial step toward more general solutions, and the proposed method could be generalised to other speaker attributes. An other insightful application could be the use of speech synthesisers conditioned on x-vector speaker representations. Indeed, this approach combined to our method could be useful for sex transformation while maintaining other speaker attributes unaltered.

6. Acknowledgements

This work was supported by the JST-ANR Japanese-French project VoicePersonae and the Robovox project ANR-18-CE33-0014

7. References

[1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, pp. 788 – 798, 06 2011.

[2] 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, pp. 5329–5333.

[3] W. Cai, J. Chen, and M. Li, "Exploring the encoding layer and loss function in end-to-end speaker and language recognition system," in *Proc. Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 74–81.

[4] D. Snyder, D. Garcia-Romero, A. McCree, G. Sell, D. Povey, and S. Khudanpur, "Spoken language recognition using x-vectors." in *Odyssey*, 2018, pp. 105–111.

[5] D. Raj, D. Snyder, D. Povey, and S. Khudanpur, "Probing the information encoded in x-vectors," *arXiv preprint arXiv:1909.06351*, 2019.

[6] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[7] V. Prince, "Sex vs. gender," *International Journal of Transgenderism*, vol. 8, no. 4, pp. 29–32, 2005.

[8] M. Pathak, J. Portêlo, B. Raj, and I. Trancoso, "Privacy-preserving speaker authentication," vol. 7483, 09 2012, pp. 1–22.

[9] A. Nautsch, A. Jiménez, A. Treiber, J. Kolberg, C. Jasserand, E. Kindt, H. Delgado, M. Todisco, M. A. Hmani, A. Mtibaa, M. A. Abdelraheem, A. Abad, F. Teixeira, D. Matrouf, M. Gomez-Barrero, D. Petrovska-Delacrétaz, G. Chollet, N. Evans, T. Schneider, J.-F. Bonastre, B. Raj, I. Trancoso, and C. Busch, "Preserving privacy in speaker and speech characterisation," *Computer Speech & Language*, vol. 58, pp. 441 – 480, 2019.

[10] 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," 09 2019, pp. 155–160.

[11] B. M. L. Srivastava, N. Vauquier, M. Sahidullah, A. Bellet, M. Tommasi, and E. Vincent, "Evaluating voice conversion-based privacy protection against informed attackers," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.

[12] H. Kameoka, T. Kaneko, K. Tanaka, and N. Hojo, "Acvae-vc: Non-parallel voice conversion with auxiliary classifier variational autoencoder," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 9, pp. 1432–1443, 2019.

[13] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from unaligned corpora using variational autoencoding wasserstein generative adversarial networks," in *Proc. Interspeech 2017*, 2017, pp. 3364–3368.

[14] L. Yingzhen and S. Mandt, "Disentangled sequential autoencoder," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80, 2018, pp. 5670–5679.

[15] S. Khurana, S. R. Joty, A. Ali, and J. Glass, "A factorial deep markov model for unsupervised disentangled representation learning from speech," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 6540–6544.

[16] W.-N. Hsu, Y. Zhang, R. J. Weiss, Y.-A. Chung, Y. Wang, Y. Wu, and J. Glass, "Disentangling correlated speaker and noise for speech synthesis via data augmentation and adversarial factorization," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5901–5905.

[17] 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. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2415

[18] J. Williams and S. King, "Disentangling style factors from speaker representations," in *Proc. Interspeech*, 2019, pp. 3945–3949.

[19] R. Peri, M. Pal, A. Jati, K. Somandepalli, and S. Narayanan, "Robust speaker recognition using unsupervised adversarial invariance," in *Proceedings of ICASSP*, May 2020.

[20] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: A large-scale speaker identification dataset," in *Proc. Interspeech 2017*, 2017, pp. 2616–2620.

[21] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *Proc. Interspeech 2018*, 2018, pp. 1086–1090.

[22] G. Lample, N. Zeghidour, N. Usunier, A. Bordes, L. Denoyer, and M. Ranzato, "Fader networks: Manipulating images by sliding attributes," in *Advances in Neural Information Processing Systems*, 2017, pp. 5967–5976.

[23] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Proc. Interspeech*, 2011, pp. 249–252.

[24] V. Nair and G. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 807–814.

[25] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul 2015, pp. 448–456.

[26] N. Brümmer and J. Du Preez, "Application-independent evaluation of speaker detection," *Computer Speech & Language*, vol. 20, no. 2-3, pp. 230–275, 2006.

[27] K. Kalantari, L. Sankar, and O. Kosut, "On information-theoretic privacy with general distortion cost functions," in *2017 IEEE International Symposium on Information Theory (ISIT)*, 2017, pp. 2865–2869.

[28] B. C. Ross, "Mutual information between discrete and continuous data sets," *PloS one*, vol. 9, no. 2, 2014.

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[30] S. Ioffe, "Probabilistic linear discriminant analysis," in *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 531–542.

[31] N. Brummer and J. Preez, "The pav algorithm optimizes binary proper scoring rules," 04 2013.