Enhancing Speech Privacy with Slicing

Mohamed Maouche, Brij Mohan Lal Srivastava, Nathalie Vauquier, Aurélien Bellet, Marc Tommasi, Emmanuel Vincent

ENHANCING SPEECH PRIVACY WITH SLICING

Mohamed Maouche[1]*, Brij Mohan Lal Srivastava*[1]*, Nathalie Vauquier*[1]*, Aurélien Bellet*[1]*, Marc Tommasi*[1]*, Emmanuel Vincent*[2]

[1]Univ. Lille, Inria, CNRS, Centrale Lille, UMR 9189 - CRIStAL, F-59000 Lille, France
[2]Université de Lorraine, CNRS, Inria, LORIA, France
<firstname>.<lastname>@inria.fr

ABSTRACT

Privacy preservation calls for speech anonymization methods which hide the speaker's identity while minimizing the impact on downstream tasks such as automatic speech recognition (ASR) training or decoding. In the recent VoicePrivacy 2020 Challenge, several anonymization methods have been proposed to transform speech utterances in a way that preserves their verbal and prosodic contents while reducing the accuracy of a speaker verification system. In this paper, we propose to further increase the privacy achieved by such methods by segmenting the utterances into shorter slices. We show that our approach has two major impacts on privacy. First, it reduces the accuracy of speaker verification with respect to unsegmented utterances. Second, it also reduces the amount of personal information that can be extracted from the verbal content, in a way that cannot easily be reversed by an attacker. We also show that it is possible to train an ASR system from anonymized speech slices with negligible impact on the word error rate.

Index Terms— anonymization, speaker verification, automatic speech recognition.

1. INTRODUCTION

With the increasing popularity of smart devices, more users have access to voice-based interfaces. They offer simple access to modern technologies and enable the development of new services. The building blocks behind these speech-based technologies are no more handcrafted but trained on large amounts of data. This is particularly true for automatic speech recognition (ASR) systems, which are often trained on speech data collected from the users to improve performance and adapt to new domains. The collection and exploitation of speech data however raises privacy threats. Indeed, speech carries personal or sensitive information about the speaker (e.g., gender, emotion, verbal content) [1,2] and it is a biometric characteristic that can be used to recognize the speaker through, e.g., i-vector [3] or x-vector [4] based speaker verification.

To address this privacy issue, various anonymization[1] methods have been proposed in the literature. These methods, which rely on simple feature transformation [6–8], feature perburbation [9], Gaussian mixture model based voice conversion [10, 11], or neural network based voice conversion [12–14], typically aim to transform speech signals in a way that preserves all content except features related with the speaker identity, thereby making it hard for an attacker to re-identify the speaker.

In addition to speech signals, ASR system training also requires the corresponding text transcripts, irrespective of whether the speech signals have been anonymized or not. These transcripts can contain personal information about the speaker too. Text sanitization methods, which delete sensitive words in the text or replace them by other words of the same type, can efficiently address this issue for text-only data [15–18]. Unfortunately, word replacement is unusable for ASR system training since it breaks the correspondence between the transcripts and the verbal contents of speech, while word removal often fails to detect and remove some sensitive words. As a result, the transcripts (or the verbal contents of speech) could be leveraged by an attacker to break the protection offered by speech anonymization.

The method we introduce in this paper follows a different path. We propose to segment every speech utterance into shorter slices after it has been anonymized. In this way, we reduce the amount of speech available to the attacker per slice, which is expected to reduce the risk of speaker re-identification with respect to unsegmented utterances. On top of that, the amount of personal information that can be extracted from each transcript slice is also reduced, since it becomes isolated from its context. We quantify the risk of speaker re-identification, as well as the risk that an attacker could reverse the slicing procedure by reassembling successive speech signals or text transcripts together. Most importantly, we also evaluate the impact of slicing on the utility of the data for ASR acoustic model training. Our experiments are conducted on LibriSpeech [19] and follow the VoicePrivacy 2020 Challenge setup [5, 20] with a stronger attacker. In particular, we employ the x-vector based voice conversion baseline [21] of the VoicePrivacy 2020 Challenge for anonymization, not only for reproducibility purposes but also because it is representative of modern neural network based anonymization methods and it still offers one of the best privacy/utility trade-offs today.

The structure of the paper is as follows. We describe the threat model in Section 2 and introduce the slicing method in Section 3. Section 4 reports the results of the evaluation on real data in terms of both privacy and utility. We present our study on the reversibility of the slicing in Section 5. We conclude in Section 6.

2. THREAT MODEL

The attack scenario is depicted in Fig. 1. *Speakers* process their voice through an *anonymization* method. This anonymization step

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Program under Grant Agreement No. 825081 COMPRISE (`https://www.compriseh2020.eu/`) and by the French National Research Agency under project DEEP-PRIVACY (ANR-18-CE23-0018). Experiments presented in this paper 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 (see `https://www.grid5000.fr`).

[1]In the legal community, the term "anonymization" means that this goal has been achieved. Following the VoicePrivacy 2020 Challenge [5], we use it to refer to the task, even when the method has failed.

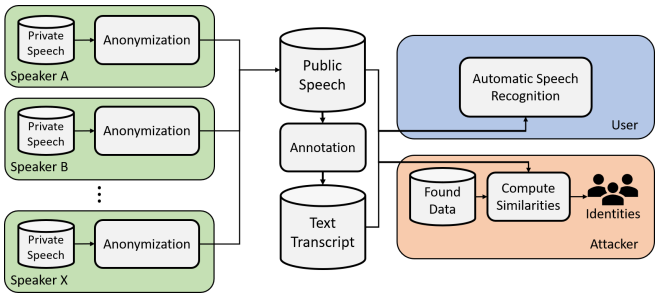

Fig. 1. Anonymization procedure and attack model.

takes as input one or more *private speech* utterances along with some configuration parameters, and outputs a new speech signal. The transformed utterances from one or more speakers form a *public speech* dataset that is processed by a third-party *user* for, e.g., ASR training/decoding or any other downstream task.

Given unprocessed or anonymized utterances from a known speaker, an *attacker* attempts to find which anonymized utterances in the public dataset are spoken by this speaker [22, 23]. Formally, an attacker has access to two sets of utterances: A (*enrollment/found data*) and B (*trial/public speech*), but knows the corresponding speakers in A only. The attacker designs a linkage function $LF(a, b)$ that outputs a score for any $a \in A$ and $b \in B$. Typically, this score is a similarity score obtained through a speaker verification system. The attacker then makes a binary decision (same vs. different speaker) based on this score.

Anonymization techniques must achieve a suitable privacy/utility trade-off. On the one hand, privacy is measured by the attacker's ability to re-identify the speaker using metrics such as equal error rate (EER) or linkability [24]. On the other hand, utility is measured by the performance of the desired downstream task(s), e.g., the word error rate (WER) of an ASR system or the intelligibility for a human listener. In the following, we are interested in the utility of the data for training an ASR acoustic model, assuming that the other components of the ASR system (lexicon, language model) are available or have been trained on text-only data (see Fig. 2).

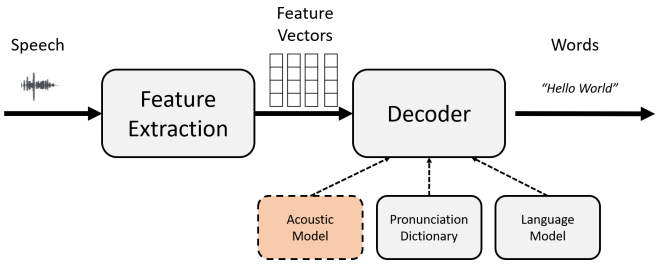

Fig. 2. ASR system architecture

3. WORD-LEVEL SLICING OF SPEECH

To increase privacy beyond the level achieved by the anonymization methods mentioned in Section 1, we propose to cut the anonymized utterances into multiple, shorter slices. To ensure that the sliced transcripts match the sliced spoken content, we constrain these cuts to happen between successive words rather than in the middle of a word. At the same time, we wish the duration of the slices to be close to a target duration δ in order to control the resulting privacy.

This is achieved by force-aligning each original (unsegmented) transcript with the corresponding utterance. For a given utterance u and transcript $w = w_1 \dots w_n$, alignment yields two series of timestamps $(t_k^s)_{1 \le k \le n}$ and $(t_k^e)_{1 \le k \le n}$ where $[t_k^s, t_k^e]$ is the time interval when word w_k has been uttered in u. To create the first slice, we start from the first word and include the following words one by one until we reach a number k such that the duration becomes at least δ. Besides the words, we keep the silence between them, as well as the silence before the first word and after the last word. We then start again from the $(k+1)$-th word to create the second slice, and so on. The final segment (if any) whose duration is shorter than δ is discarded. See Algorithm 1 for details.

Algorithm 1 Word-level slicing method

function SLICE(u: speech signal, w: transcript, δ: target (minimum) duration)
  $slices \leftarrow \emptyset$
  $\mathbb{A} \leftarrow Align(u, w)$
  $t_{prv} \leftarrow 0$
  $k_{prv} \leftarrow 0$
  for k in $range(|w|)$ **do**
    $t_k^s, t_k^e \leftarrow \mathbb{A}[k]$
    if $k+1 \le |w|$ **then**
      $t_{k+1}^s, t_{k+1}^e \leftarrow \mathbb{A}[k+1]$
    else
      $t_{k+1}^s \leftarrow duration(u)$
    end if
    if $t_{k+1}^s - t_{prv} \ge \delta$ **then**
      $slices \leftarrow slices \cup (u[t_{prv} : t_{k+1}^s], w[k_{prv} : k])$
      $t_{prv} \leftarrow t_k^e$
      $k_{prv} \leftarrow k+1$
    end if
  end for
  return $slices$
end function

4. UTILITY AND PRIVACY EVALUATION

In this section, we evaluate how the duration of the slices impacts the privacy/utility trade-off.

4.1. Experimental Setup

Anonymization: We conduct experiments using the first baseline of the VoicePrivacy 2020 Challenge [5] as the anonymization method. This method extracts pitch, bottleneck, and (source) x-vector features from the input speech. Then it re-synthesizes a speech signal using the original pitch and bottleneck features and a new target x-vector selected randomly using one of several possible strategies. In the following, we choose the *dense* strategy with *random* gender, which was reported to be the most successful in [21]. Data which have not been anonymized are referred to as *clear* data.

Slicing: Slicing is performed using forced-alignments obtained using the pretrained model Gentle.[2]

Privacy metric: We consider an attacker who assesses the speaker similarity between an enrollment and a trial utterance using

[2] https://lowerquality.com/gentle/

the probabilistic linear discriminant analysis (PLDA) score between their x-vectors. Privacy is evaluated via the linkability $D_{\leftrightarrow}^{\text{sys}}$, which measures how much the distributions of same- and different-speaker scores overlap [24]. We assume that the attacker is *semi-informed*: he/she knows the anonymization method (but not the mapping from source to target x-vectors) and uses that knowledge to anonymize the enrollment data and the training data for the x-vector and PLDA models [23]. Crucially, by contrast with the VoicePrivacy Challenge setup which maps all utterances of a given speaker to the same target x-vector, we map each training utterance to a different target x-vector. This greatly increases the attacker's strength and highlights the limited privacy offered by anonymization alone, with linkability jumping from 0.09 in [20] to 0.63 here. The x-vector and PLDA models are trained and tested using the Kaldi [25] recipe in [5], except that the training, enrollment and trial data are sliced.

ASR system and utility metric: To evaluate the utility of sliced utterances for ASR acoustic model training, we use the state-of-the-art Kaldi [25] ASR recipe for LibriSpeech involving a factorized time delay neural network (TDNN-F) acoustic model and a 3-gram language model. The recipe is identical to [5], except that we train it on sliced utterances and test it on unsegmented utterances. We report the resulting WER.

Datasets: The experiments are conducted on LibriSpeech [19]. The *train-clean-360* set (∼1 k speakers, ∼100 k utterances and 360 h of speech) is used to train the x-vector and PLDA models and the ASR system. Part of the *test-clean* set (40 speakers, 1,496 utterances) forms the *trial/public* data. The remaining part (29 speakers, 438 utterances) is considered as *enrollment/found* data.

4.2. Effect of Slicing on Utility

We first explore which utterance durations are suitable for training an ASR acoustic model. Table 1 reports the WERs achieved in four cases, depending on whether the training data has been anonymized or not before slicing and whether the test data has been anonymized or not. In each case, we present the results for different slicing durations and for the original utterance duration. We notice an increase in the WER when decoding anonymized data with an ASR acoustic model trained on clear data and vice-versa, which can be attributed to a mismatch between the training and test data distributions. Conversely, the WER obtained when training and testing on anonymized data (4.86%) is similar to training and testing on clear data (4.26%). As for the effect of slicing itself, we notice that for clear training data 1.5 s is the shortest possible duration, below which the WER degrades a lot. With anonymized training data, the duration can also be shortened to 1.5 s when decoding anonymized speech. The resulting WER (4.90%) is statistically equivalent to anonymization alone.

Table 1. WER (%) achieved on (unsegmented) test data when training the ASR acoustic model on sliced data.

Training data	Slicing	Test data	
		Clear	Anonymized
Clear	None	4.26	7.56
	$\delta = 2$ s	4.44	7.58
	$\delta = 1.5$ s	4.66	8.00
	$\delta = 1$ s	6.11	11.4
Anonymized	None	10.93	4.86
	$\delta = 4$ s	11.83	4.89
	$\delta = 3$ s	13.38	4.90
	$\delta = 1.5$ s	21.46	4.90

4.3. Effect of Slicing on Privacy

In terms of privacy, we present in Fig. 3 the linkability achieved with utterances of different durations. We consider the setting where the attacker aims to re-identify speakers in the trial/public data, hence the focus is now on test data (instead of training data). The red and yellow curves are obtained by shortening the utterances to a fixed duration (irrespective of word boundaries). The results on clear data illustrate the positive impact of shorter utterances on linkability, especially for durations shorter than 1 s. Unfortunately, our utility experiment demonstrated that utterances shorter than 1 s are too short to train an ASR system. Also, for durations longer than 1 s the level of privacy offered by shortening alone is insufficient. For this reason, shortening must be combined with anonymization.

In addition, we show in the figure the results obtained with the word-level slicing method of Algorithm 1 and with the unsegmented anonymized utterances (black bars). Word-level slicing achieves consistent results with shortening to a fixed duration of 1.5 or 2 s. This shows that the word-level constraint, which is desirable in the context of ASR training, does not come at the cost of a loss of privacy. The linkability achieved when slicing anonymized utterances with $\delta = 1.5$ s decreases to 0.21, compared to 0.63 before slicing.[3]

To sum up, slicing anonymized data with $\delta = 1.5$ s greatly decreases the linkability while maintaining the utility for ASR training.

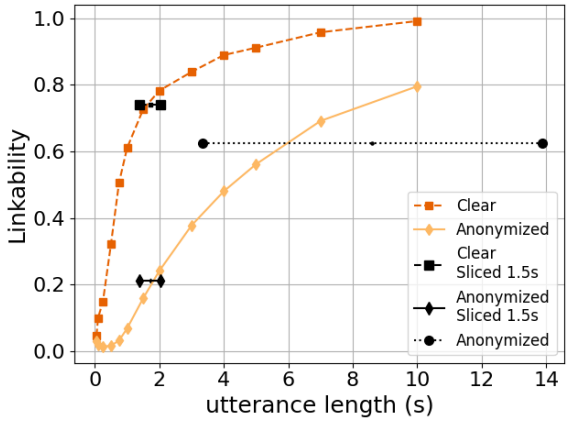

Fig. 3. Linkability achieved by shortening utterances to a fixed maximum duration (red and yellow curves), or by word-level slicing with $\delta = 1.5$ s or anonymization alone (black bars). Each black bar represents the mean and standard deviation of the utterance durations.

5. REVERSIBILITY OF THE SLICING

In this section, we quantify the risk that an attacker could reverse the slicing and successfully reassemble successive speech signals or text transcripts together. Our study focuses on the simpler task of linking two successive slices together, as an attacker who performs poorly on this problem is unlikely to be able to reassemble entire utterances.

[3] Surprisingly, the linkability value achieved on unsegmented utterances (0.63) is lower than on utterances shortened to 7 or 10 s (up to 0.79). We attribute this to the wide range of utterance durations (8.6 s average with ±5.2 s standard deviation), where each duration would have in average a different linkability compared to the fixed duration for longer segments.

5.1. Text Successiveness

Regarding text, we design an attacker that leverages a language model to construct a "text successiveness score" to be used as linkage function, similarly to the speaker verification attack described in Section 2. The language model estimates the probability $P(w)$ of any sentence w. The attacker uses P to compute a score function $SF(w, v)$ between two transcripts w and v. The higher the score, the higher the chance that the transcripts are successive.

In our experiments, we used a 3-gram language model

$$P(w) \overset{\text{3-gram}}{\approx} P(w_1)P(w_2 \mid w_1) \prod_{k=3}^{\text{len}(w)} P(w_k \mid w_{k-2:k-1}), \quad (1)$$

where $w_{i:j} = w_i w_{i+1} \dots w_j$. To restrict our attention to the terms involving both w and v, we define the score function as

$$SF(w, v) = P(w_{n-1} w_n v_1 v_2). \quad (2)$$

To retrieve the successor of a given slice w in the public dataset, the attacker computes the scores $SF(w, v)$ for all other slices v in the dataset and sorts them in decreasing order. The success of the attack can be quantified via the rank $r(w)$ of the correct successor. We consider the following ranking metrics: (1) Average normalized rank: mean of $r(w)$ over all w, divided by the maximum possible rank (that is the number of slices minus one); (2) Median normalized rank: median of $r(w)$ divided by the maximum possible rank; (3) Precision at top-1: how often the slice with top score is the successor; (4) Precision at top-10%: how often the successor belongs to the top-10% scores. In addition to the LibriSpeech test set, which may be more easily attacked due to speakers reading text from distinct books including specific words like character names, we also consider Mozilla Common Voice test set. In the latter case, we slice the transcripts into 3-word slices (the average number of words per second is 2.7) and we retrain the language model.

The results are presented in Table 2. We notice that the correct successive slice usually has large rank (27%–32% normalized rank in average and 16%–23% median rank) meaning that thousands of false successive slices have a better score. We also see that the correct slice almost never ranks first (less than 3% of the cases), and rarely in the top-10% (only a third of the cases).

Table 2. Text-based successor identification performance.

Test set	LibriSpeech				Common Voice
Target slice duration δ (s)	1	1.5	3	4	/
Number of slices	14,931	11,330	5,407	5,487	5,292
Average normalized rank (%)	27.25	28.31	29.37	30.24	32.38
Median normalized rank (%)	16.11	17.87	18.81	20.92	23.14
Precision at top-1 (%)	1.39	1.41	2.18	2.56	0.75
Precision at top-10% (%)	40.48	37.8	38.36	37.84	33.89

5.2. Speech Successiveness

We now consider the problem of linking two successive speech signals. Our approach is to concatenate the two signals and score them by the softmax score of a binary classifier trained to distinguish successive vs. non-successive pairs. We use the TDNN architecture proposed in [4] for speaker classification (see Table 3). To prevent the model from learning to classify most examples as non-successive, we train it on a balanced dataset: half of the training examples are successive, one fourth are non-successive from the same speakers and one fourth are non-successive from different speakers. The training data are taken from LibriSpeech *train-clean-360* sliced with $\delta = 1.5$ s and anonymized. To evaluate the attacker's performance, we sample 100 utterances from Librispeech *test-clean* sliced with $\delta = 1.5$ s and anonymized. This allows us to construct 232 successive pairs, $5,378$ non-successive same-speaker pairs, and $125,796$ non-successive different-speaker pairs. We observe that, even though the overall accuracy of the classifier is 80%, the vast majority of correct classifications are for the easier, non-successive different-speaker class.

Table 3. TDNN architecture for speech successiveness classification. The inputs are 23-dimensional MFCCs of utterance1–silence–utterance2 and the output is the posterior with label successive-true or successive-false

Layer	Layer context	Total context	Input Size	Output Size
frame1	$\{t-2 : t+2\}$	5	23x5	512
frame2	$\{t-2, t, t+2\}$	9	512x3	512
frame3	$\{t-3, t, t+3\}$	15	512x3	1,500
stats-pooling	$[0, L]$	L	1,500xL	3,000
segment	/	L	3,000	512
softmax	/	L	512	2

Table 4. Speech-based successor identification performance.

Test set	Number of slices	Average normalized rank (%)	Median normalized rank (%)	Precision at top-1 (%)	Precision at top-10% (%)
LibriSpeech Sliced $\delta = 1.5$ s & anonymized	364	43.48	19.83	2.48	38.29

We consider the same ranking metrics as in Section 5.1. The results given in Table 4 show that the average rank of the correct slice is 158 (top-43%), with a median of 72 (top-20%). Furthermore, the top-1 precision is again lower than 3%. Overall, these results show that it is very difficult for an attacker to consistently find the correct successive slice, and thus it is even harder to reassemble entire utterances.

6. CONCLUSION

We improved the privacy gain from state-of-the-art x-vector based anonymization methods by reducing the length of the utterances by means of slicing the speech into smaller segments. We showed that slices of 1.5 s of speech can be used to train an ASR system with 4.90% word error rate (WER) (compared to 4.86% without slicing) and that such slices reduce x-vector based speaker verification linkability to 0.21 (compared to 0.63 without slicing). On top of that the sensitive information contained on the text is reduced because each utterance contains few words, which become isolated from their context which is "hidden in the crowd" of all other slices. We also showed that reversing the slicing to rebuild the initial utterances is a difficult task for an attacker.

7. REFERENCES

[1] Slobodan Ribaric, Aladdin Ariyaeeinia, and Nikola Pavesic, "De-identification for privacy protection in multimedia content: A survey," *Signal Processing: Image Communication*, vol. 47, pp. 131–151, 2016.

[2] Andreas Nautsch, Abelino Jiménez, Amos Treiber, Jascha Kolberg, Catherine Jasserand, Els Kindt, Héctor Delgado, Massimiliano Todisco, Mohamed Amine Hmani, Aymen Mtibaa, Mohammed Ahmed Abdelraheem, Alberto Abad, Francisco Teixeira, Driss Matrouf, et al., "Preserving privacy in speaker and speech characterisation," *Computer Speech and Language*, vol. 58, pp. 441–480, 2019.

[3] Najim Dehak, Patrick J. Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2010.

[4] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.

[5] Natalia Tomashenko, Brij Mohan Lal Srivastava, Xin Wang, Emmanuel Vincent, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Jose Patino, Jean-François Bonastre, Paul-Gauthier Noé, and Massimiliano Todisco, "Introducing the VoicePrivacy initiative," in *Interspeech*, 2020, pp. 1693–1697.

[6] Carmen Magariños, Paula Lopez-Otero, Laura Docio-Fernandez, Eduardo Rodriguez-Banga, Daniel Erro, and Carmen Garcia-Mateo, "Reversible speaker de-identification using pre-trained transformation functions," *Computer Speech and Language*, vol. 46, pp. 36–52, 2017.

[7] Jianwei Qian, Haohua Du, Jiahui Hou, Linlin Chen, Taeho Jung, Xiang-Yang Li, Yu Wang, and Yanbo Deng, "Voicemask: Anonymize and sanitize voice input on mobile devices," *arXiv preprint arXiv:1711.11460*, 2017.

[8] Jose Patino, Natalia Tomashenko, Massimiliano Todisco, Andreas Nautsch, and Nicholas Evans, "Speaker anonymisation using the McAdams coefficient," in *Interspeech*, 2021, pp. 1099–1103.

[9] Yaowei Han, Sheng Li, Yang Cao, Qiang Ma, and Masatoshi Yoshikawa, "Voice-indistinguishability: Protecting voiceprint in privacy-preserving speech data release," *arXiv preprint arXiv:2004.07442*, 2020.

[10] Qin Jin, Arthur R. Toth, Tanja Schultz, and Alan W. Black, "Speaker de-identification via voice transformation," in *2009 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2009, pp. 529–533.

[11] Miran Pobar and Ivo Ipšić, "Online speaker de-identification using voice transformation," in *37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2014, pp. 1264–1267.

[12] Fahimeh Bahmaninezhad, Chunlei Zhang, and John H. L. Hansen, "Convolutional neural network based speaker de-identification," in *Odyssey*, 2018, pp. 255–260.

[13] Fuming Fang, Xin Wang, Junichi Yamagishi, Isao Echizen, Massimiliano Todisco, Nicholas Evans, and Jean-François Bonastre, "Speaker anonymization using x-vector and neural waveform models," in *10th ISCA Speech Synthesis Workshop (SSW)*, 2019, pp. 155–160.

[14] Gauri P. Prajapati, Dipesh K. Singh, Preet P. Amin, and Hemant A. Patil, "Voice privacy through x-vector and CycleGAN-based anonymization," in *Interspeech*, 2021, pp. 1684–1688.

[15] Min Tang, Dilek Hakkani-Tür, and Gokhan Tur, "Preserving privacy in spoken language databases," in *ECML/PKDD International Workshop on Privacy and Security Issues in Data Mining*, 2004.

[16] Özlem Uzuner, Yuan Luo, and Peter Szolovits, "Evaluating the state-of-the-art in automatic de-identification," *Journal of the American Medical Informatics Association*, vol. 14, no. 5, pp. 550–563, 2007.

[17] David Ifeoluwa Adelani, Ali Davody, Thomas Kleinbauer, and Dietrich Klakow, "Privacy guarantees for de-identifying text transformations," in *Interspeech*, 2020, pp. 4666–4670.

[18] Aitor García-Pablos, Naiara Perez, and Montse Cuadros, "Sensitive data detection and classification in Spanish clinical text: Experiments with BERT," *arXiv preprint arXiv:2003.03106*, 2020.

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

[20] Natalia Tomashenko, Xin Wang, Emmanuel Vincent, Jose Patino, Brij Mohan Lal Srivastava, Paul-Gauthier Noé, Andreas Nautsch, Nicholas Evans, Junichi Yamagishi, Benjamin O'Brien, Anaïs Chanclu, Jean-François Bonastre, Massimiliano Todisco, and Mohamed Maouche, "The VoicePrivacy 2020 Challenge: Results and findings," *arXiv preprint arXiv:2109.00648*, 2021.

[21] Brij Mohan Lal Srivastava, Natalia Tomashenko, Xin Wang, Emmanuel Vincent, Junichi Yamagishi, Mohamed Maouche, Aurélien Bellet, and Marc Tommasi, "Design choices for x-vector based speaker anonymization," in *Interspeech*, 2020, pp. 1713–1717.

[22] Jianwei Qian, Feng Han, Jiahui Hou, Chunhong Zhang, Yu Wang, and Xiang-Yang Li, "Towards privacy-preserving speech data publishing," in *2018 IEEE Conference on Computer Communications (INFOCOM)*, 2018, pp. 1079–1087.

[23] Brij Mohan Lal Srivastava, Nathalie Vauquier, Md Sahidullah, Aurélien Bellet, Marc Tommasi, and Emmanuel Vincent, "Evaluating voice conversion-based privacy protection against informed attackers," in *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 2802–2806.

[24] Mohamed Maouche, Brij Mohan Lal Srivastava, Nathalie Vauquier, Aurélien Bellet, Marc Tommasi, and Emmanuel Vincent, "A comparative study of speech anonymization metrics," in *Interspeech*, 2020, pp. 1708–1712.

[25] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlíček, Yanmin Qian, Petr Schwarz, Jan Silovský, Georg Stemmer, and Karel Veselý, "The Kaldi speech recognition toolkit," in *2011 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2011.