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 anonymization methods which hide the speaker's identity in speech signals while minimizing the impact on downstream tasks such as automatic speech recognition (ASR) training or decoding. In the VoicePrivacy 2020 Challenge, voice 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, segmentation.

1. Introduction

With the increasing popularity of smart devices, more users have access to voice-based interfaces. The underlying technologies, especially automatic speech recognition (ASR), are often trained on speech data collected from the users to improve performance and adapt to new domains. The collection and exploitation of this data raises privacy threats. Indeed, speech carries personal or sensitive information about the speaker (e.g., gender, age, emotion) [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 voice 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], aim to transform speech signals in a way that preserves all content except features related to the speaker identity, thereby making it hard for an attacker to re-identify the speaker.

Besides speech signals, ASR system training also requires the corresponding text transcripts, irrespective of whether the speech signals have been subject to voice anonymization or not. These transcripts can contain personal information about the speaker too. Text sanitization methods, which redact or replace sensitive words in the text, can mitigate 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 redaction often fails to detect and remove some sensitive words. As a result, the transcripts (or the verbal content of speech) could be exploited by an attacker to break the protection offered by voice 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 in each slice, which is expected to lower 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 the transcript of each slice is also reduced, since it becomes isolated from its context. We quantify the risk of speaker re-identification, and 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 use the x-vector based voice conversion baseline [21, 22] of the Challenge for anonymization, not only for reproducibility purposes but also because it is representative of modern neural network based voice anonymization methods and it still offers one of the best privacy/utility trade-offs today.

While slicing may be seen as a simple method, it has not been studied in the context of privacy so far. Previous studies on the impact of utterance duration on speaker verification performance focused on clear (non-anonymized) utterances and did not evaluate the utility for ASR [23–25]. Our study is the first one that tackles privacy, utility and reassembling together. Our results highlight a sweet spot in the choice of slice length for which our method provides a large increase in privacy (larger than the one provided by voice anonymization itself) with no loss of utility. In addition, we show the difficulty for an attacker to reassemble the utterance from short slices of speech or text.

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 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 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

[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.

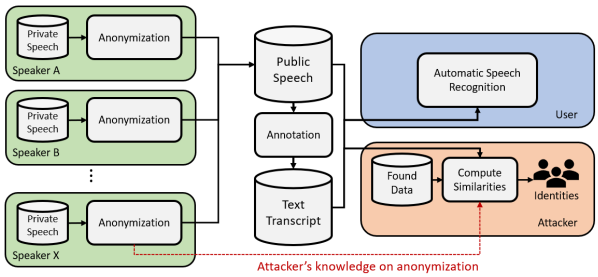

Figure 1: *Anonymization procedure and attack model.*

user for, e.g., ASR training/decoding or any 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 [26, 27]. 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 obtained through a speaker verification system. The attacker then makes a binary decision (same vs. different speaker) based on this score. We also consider that the attacker knows the anonymization method used and can leverage it to enhance the attack.

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 [28]. 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).

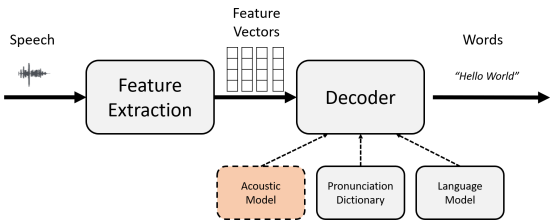

Figure 2: *ASR system architecture.*

3. Word-Level Slicing of Speech

To increase privacy beyond the level achieved by the voice 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 trade-off between privacy and utility.

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**
    $t_k^s, t_k^e \leftarrow \mathbb{A}[k]$ # *w_k is uttered in $[t_k^s, t_k^e]$*
    if $k + 1 \le |w|$ **then**
      $t_{k+1}^s, t_{k+1}^e \leftarrow \mathbb{A}[k+1]$ # *Next word*
    else
      $t_{k+1}^s \leftarrow duration(u)$ # *Last word*
    end if
    if $t_{k+1}^s - t_{prv} \ge \delta$ **then** # *Slice complete*
      $slices \leftarrow slices \cup (u[t_{prv} : t_{k+1}^s], w[k_{prv} : k])$
      $t_{prv} \leftarrow t_k^e$ # *Start of new slice*
      $k_{prv} \leftarrow k + 1$ # *Update starting word*
    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

Voice anonymization: We use the first baseline of the VoicePrivacy 2020 Challenge [5] as the voice anonymization method. This method extracts pitch, bottleneck, and (source) x-vector features from the input speech. It then re-synthesizes a speech signal using the original pitch and bottleneck features and a new target x-vector generated from a public pool of x-vectors using one of several possible strategies. In the following, we do not use the default strategy reported in [5]. Instead, we choose the so-called *dense* strategy with *random* gender, which was reported to be the most successful in [22]. 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 (`https://lowerquality.com/gentle/`).

Privacy metric: The attacker assesses the speaker similarity between an enrollment and a trial utterance using the probabilistic linear discriminant analysis (PLDA) score between their x-vectors. Privacy is evaluated via the linkability $D_{\leftrightarrow}^{\text{sys}}$, which measures the non-overlap between the distributions of same- and different-speaker scores [28]. Lower linkability means higher privacy. We assume a *semi-informed* attacker who 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 [27]. In contrast to the VoicePrivacy 2020 Challenge where all training utterances of a given speaker are mapped to the same target x-vector, our attacker maps each training utterance to a different target.[2] This greatly increases the attacker's strength and highlights the limited privacy offered by voice anonymization alone, with linkability jumping from 0.18 in [22, Fig. 11 right] to 0.63 here. The x-vector and PLDA models are trained and tested using the Kaldi [30] recipe in [5], except that the 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 [30] 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 x-vector and PLDA models and the ASR system are trained on the *train-clean-360* set (∼1k speakers, ∼100k utterances and 360 h of speech). Part of the *test-clean* set (40 speakers, 1,496 utterances) forms the *trial/public* data. The remaining part (29 speakers, 438 utterances) forms the *enrollment/found* data. This is the established VoicePrivacy 2020 setup [5].

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 report the results achieved with the original training utterances and with different slicing durations. 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 training/test mismatch. Nevertheless, the WER obtained when training and testing on unsegmented anonymized data (4.86%) is similar to training and testing on unsegmented 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 be shortened to 1 s only when decoding anonymized speech. The resulting WER (4.92%) is statistically equivalent to training on unsegmented anonymized data.

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 purple and orange 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 used together with anonymization: this combination yields a high level of privacy for $\delta = 1$ s.

[2]This newly proposed attacker has recently been selected for evaluation in the VoicePrivacy 2022 Challenge [29].

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
	$\delta = 3$ s	4.31	7.36
	$\delta = 2$ s	4.44	7.58
	$\delta = 1.5$ s	4.66	8.00
	$\delta = 1$ s	6.11	11.4
	$\delta = 0.5$ s	26.59	35.67
Anonymized	None	10.93	4.86
	$\delta = 3$ s	13.38	4.90
	$\delta = 2$ s	15.94	4.95
	$\delta = 1.5$ s	21.46	4.90
	$\delta = 1$ s	30.13	4.92
	$\delta = 0.5$ s	71.28	8.77

In addition, we show the results obtained with word-level slicing (Algorithm 1) and with unsegmented anonymized utterances. Word-level slicing achieves consistent results with shortening to a fixed duration of 1 s or 1.5 s. We observe that the word-level constraint, which is desirable for ASR training, does not come at the cost of a privacy loss. The linkability achieved when slicing anonymized utterances with $\delta = 1$ s decreases to 0.14, compared to 0.63 before slicing.[3]

To sum up, the results of Sections 4.2–4.3 show that slicing anonymized data with $\delta = 1$ s greatly decreases the linkability while maintaining the utility for ASR training.

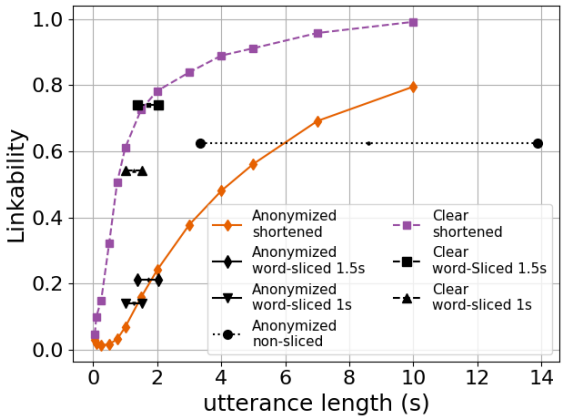

Figure 3: *Linkability achieved by shortening utterances to a fixed duration (purple and yellow curves), by word-level slicing (short horizontal bars) and without slicing (long horizontal bar). The horizontal bar of a given setup is positioned by the set's overall linkability and spans the mean and standard deviation of the utterance durations.*

5. Reversibility of the Slicing

We assess the risk that an attacker manages to reverse the slicing and *reassemble* speech slices or text transcripts together. We focus on the task of linking two successive slices, since an attacker

[3]Surprisingly, the linkability 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), which increases x-vector variability.

who performs poorly on this task is unlikely to *reassemble* entire utterances. Due to the novelty of slicing for privacy, there exists no reassembling method which we can compare to.

5.1. Text Successiveness

Regarding text, we design an attacker that leverages a language model (LM) to construct a "text successiveness score" to be used as linkage function, similarly to the speaker verification attack described in Section 2. The LM 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 LM

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

where $w_{i:j} = w_i w_{i+1} \dots w_j$ and $w_p w_q$ denotes concatenation between words/sentences. 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 the 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 LM.

Table 2 presents the results. 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 wrong 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 one 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. 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 $\delta = 1$ s or (resp., $\delta = 1.5$ s) and anonymized. To evaluate the attacker's performance, we sample five times 100 utterances from Librispeech *test-clean* sliced with $\delta = 1$ s (resp., $\delta = 1.5$ s) and anonymized. This allows us to construct on average for $\delta = 1$ s, 528 successive pairs, $15,142$ non-successive same-speaker pairs, and $378,821$ non-successive different-speaker pairs. For $\delta = 1.5$, we obtain on average 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 classifiers is 80%, the vast majority of correct classifications are for the easier, non-successive different-speaker class.

Table 3: *Speech-based successor identification performance. We report the average results of 5 repeated experiments (the standard deviation is not reported as it is lower than 0.01)*

Anonymized Librispeech Test Set	Number of slices	Average normalized rank (%)	Median normalized rank (%)	Precision at top-1 (%)	Precision at top-10% (%)
Sliced $\delta = 1.5$ s	364	43.48	19.83	2.48	38.29
Sliced $\delta = 1$ s	627	42.77	25.28	1.34	29.52

We consider the same ranking metrics as in Section 5.1. The results given in Table 3 show that for $\delta = 1$, the average rank of the correct slice is 268 (top-43%), with a median of 158 (top-25%). Furthermore, the top-1 precision is again lower than 2%. 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 provided the first study on slicing speech utterances into shorter segments in the context of speech anonymization. Combining our slicing approach with state-of-the-art x-vector based anonymization methods, we showed that slices of 1 s of speech can be used to train an ASR system with 4.92% WER (compared to 4.86% without slicing) while reducing speaker verification performance to 0.14 linkability (compared to 0.63 without slicing). In addition, our approach naturally helps to obfuscate sensitive information contained in the verbal content as each slice contains few words that become isolated from their context. Finally, we showed that reversing the slicing to reconstruct the original utterances is a very difficult task.

7. Acknowledgements

This work was supported in part by the European Union's Horizon 2020 Research and Innovation Program under Grant Agreements No. 825081 COMPRISE and No. 952215 TAILOR 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`).

8. References

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

[2] 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 *et al.*, "Preserving privacy in speaker and speech characterisation," *Computer Speech and Language*, vol. 58, pp. 441–480, 2019.

[3] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. 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] 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)*, 2018, pp. 5329–5333.

[5] 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," in *Interspeech*, 2020, pp. 1693–1697.

[6] 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 and Language*, vol. 46, pp. 36–52, 2017.

[7] J. Qian, H. Du, J. Hou, L. Chen, T. Jung, X.-Y. Li, Y. Wang, and Y. Deng, "Voicemask: Anonymize and sanitize voice input on mobile devices," *arXiv preprint arXiv:1711.11460*, 2017.

[8] J. Patino, N. Tomashenko, M. Todisco, A. Nautsch, and N. Evans, "Speaker anonymisation using the McAdams coefficient," in *Interspeech*, 2021, pp. 1099–1103.

[9] Y. Han, S. Li, Y. Cao, Q. Ma, and M. Yoshikawa, "Voice-indistinguishability: Protecting voiceprint in privacy-preserving speech data release," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, 2020, pp. 1–6.

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

[11] M. Pobar and I. 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] F. Bahmaninezhad, C. Zhang, and J. H. L. Hansen, "Convolutional neural network based speaker de-identification," in *Odyssey*, 2018, pp. 255–260.

[13] 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 *10th ISCA Speech Synthesis Workshop (SSW)*, 2019, pp. 155–160.

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

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

[16] Ö. Uzuner, Y. Luo, and P. 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] D. I. Adelani, A. Davody, T. Kleinbauer, and D. Klakow, "Privacy guarantees for de-identifying text transformations," in *Interspeech*, 2020, pp. 4666–4670.

[18] A. García-Pablos, N. Perez, and M. Cuadros, "Sensitive data detection and classification in Spanish clinical text: Experiments with BERT," in *12th Language Resources and Evaluation Conference (LREC)*, 2020, pp. 4486–4494.

[19] V. Panayotov, G. Chen, D. Povey, and S. 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] N. Tomashenko, X. Wang, E. Vincent, J. Patino, B. M. L. Srivastava, P.-G. Noé, A. Nautsch, N. Evans, J. Yamagishi, B. O'Brien, A. Chanclu, J.-F. Bonastre, M. Todisco, and M. Maouche, "The VoicePrivacy 2020 Challenge: Results and findings," *Computer Speech and Language*, vol. 74, p. 101362, 2022.

[21] B. M. L. Srivastava, N. Tomashenko, X. Wang, E. Vincent, J. Yamagishi, M. Maouche, A. Bellet, and M. Tommasi, "Design choices for x-vector based speaker anonymization," in *Interspeech*, 2020, pp. 1713–1717.

[22] B. M. L. Srivastava, M. Maouche, M. Sahidullah, E. Vincent, A. Bellet, M. Tommasi, N. Tomashenko, X. Wang, and J. Yamagishi, "Privacy and utility of x-vector based speaker anonymization," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, to appear.

[23] A. Poddar, M. Sahidullah, and G. Saha, "Speaker verification with short utterances: a review of challenges, trends and opportunities," *IET Biometrics*, vol. 7, no. 2, pp. 91–101, 2018.

[24] ——, "Quality measures for speaker verification with short utterances," *Digital Signal Processing*, vol. 88, pp. 66–79, 2019.

[25] I. Viñals, A. Ortega, A. Miguel, and E. Lleida, "An analysis of the short utterance problem for speaker characterization," *Applied Sciences*, vol. 9, no. 18, p. 3697, 2019.

[26] J. Qian, F. Han, J. Hou, C. Zhang, Y. Wang, and X.-Y. Li, "Towards privacy-preserving speech data publishing," in *2018 IEEE Conference on Computer Communications (INFOCOM)*, 2018, pp. 1079–1087.

[27] 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, pp. 2802–2806.

[28] M. Maouche, B. M. L. Srivastava, N. Vauquier, A. Bellet, M. Tommasi, and E. Vincent, "A comparative study of speech anonymization metrics," in *Interspeech*, 2020, pp. 1708–1712.

[29] N. Tomashenko, X. Wang, X. Miao, H. Nourtel, P. Champion, M. Todisco, E. Vincent, N. Evans, J. Yamagishi, and J.-F. Bonastre, "The VoicePrivacy 2022 Challenge evaluation plan," https://www.voiceprivacychallenge.org/vp2020/docs/VoicePrivacy_2022_Eval_Plan_v1.0.pdf, 2022.

[30] 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. Veselý, "The Kaldi speech recognition toolkit," in *2011 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2011.