

Speech Privacy-Preserving Method Using Random Permutation of Speech Segments

1st Haruya Tominaga
Tokyo Metropolitan University
Tokyo, Japan
tominaga-haruya@ed.tmu.ac.jp

2nd Sayaka Shiota
Tokyo Metropolitan University
Tokyo, Japan
sayaka@tmu.ac.jp

Abstract—In this paper, we propose a method for preserving speech privacy by randomly permuting speech segments. With the advancement of speech-related applications, storing speech data on cloud servers has become increasingly common. Since cloud services are often managed by third-party providers, there is a risk of speech data leakage from the servers. Therefore, a privacy-preserving approach is required to reduce the risks of leakage. To realize the speech encryption, we propose a method that segments speech and randomly permutes the segment order. The proposed method conceals the speech content while preserving the speaker’s identity. Experiments evaluated the privacy-preserving performance of the proposed method under speaker verification tasks and speech recognition tasks. The experiments demonstrated that the proposed method offers a simple and effective approach to speech privacy preservation.

Index Terms—speech privacy-preserving, random permutation, speaker verification, speech recognition

I. INTRODUCTION

In recent years, the deployment of deep learning models for services such as automatic speech recognition and spoken dialogue systems on cloud platforms has been increasing. Since cloud services are often managed by external providers, there is a risk of speech data leakage from servers due to attacks from both internal and external sources [1], [2]. Speech data contains not only spoken content but also personally identifiable information such as language, age, and gender [3], [4]. Therefore, if such data is leaked, it can lead to the exposure of personal and confidential information. Against this backdrop, the Voice Privacy Challenge was held in 2020 to promote research and evaluation of voice anonymization technologies [5]. As a result, technologies for protecting speech privacy have begun to gain global attention, and methods such as speech pseudonymization and anonymization have been proposed [6]–[9]. These pseudonymization and anonymization techniques primarily focus on concealing the speaker identity, while conventional privacy-preserving methods [10] aim to protect both speaker identity and speech content. However, in the context of voice-based biometric authentication systems, there are scenarios where speaker identity must be preserved for legitimate authentication purposes while speech content requires protection from unauthorized access. Therefore, in this paper, we define speech privacy preservation as the protection of speech content while intentionally preserving speaker identity information for authentication purposes. A

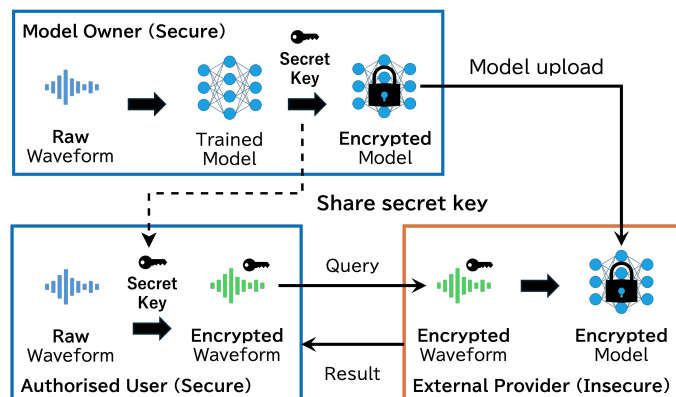


Fig. 1: Conventional privacy-preserving scenario

speech privacy-preserving method using a secret key based on a random orthogonal matrix has been proposed [10]. The conventional privacy-preserving method conceals both the speaker’s identity and the speech content under a secret key-sharing scenario between a model owner and an authorized user. In contrast, a speech encryption method that is free from model encryption eliminates the need for key sharing, providing a more convenient framework. Therefore, in this paper, we propose a speech privacy-preserving method that segments the speech and randomly permutes the segment order, without requiring key sharing. This approach aims to conceal speech content while retaining the speaker information by dividing the speech data into segments and randomly changing their order. The proposed method conceals speech content while retaining speaker information, making it suitable for privacy protection in biometric authentication.

II. CONVENTIONAL METHOD

The conventional method [10] uses a secret key based on a random orthogonal matrix to protect speech privacy. Figure 1 illustrates the privacy protection scenario assumed by the conventional method. First, the model creator trains a model using unencrypted speech data in a secure environment. After training, the learned model is encrypted using a secret key. The model creator then uploads the encrypted model to a server and provides the secret key to authorized users. It’s important to note that external providers, such as cloud services, are

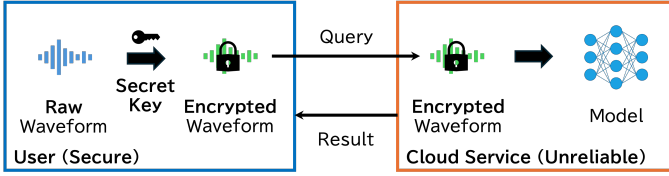


Fig. 2: Proposed privacy-preserving scenario

assumed to be insecure environments since they are managed by third parties. When an authorized user wants to use the model uploaded to the server, they encrypt their query speech data using the secret key received from the model creator. This encrypted speech data is then uploaded to the server. Finally, the server takes the encrypted speech data as input to the encrypted model and returns the result to the authorized user.

Under this conventional method, only the encrypted model and queries are uploaded or stored on the external provider's server. This means that only authorized users who possess the secret key can utilize the model as intended by the model creator. Furthermore, even if a third party without the secret key steals the encrypted speech data from the external provider, they cannot obtain the original information from the encrypted data. However, a drawback of this conventional method is the need for secret key sharing between the model creator and the users. If the secret key is not shared, users cannot access or utilize the system, which presents a challenge in its practical implementation.

III. PROPOSED METHOD

A. Privacy-preserving scenario

Figure 2 illustrates the privacy-preserving scenario of the proposed method. When a user accesses a model deployed on a server, query speech data is first encrypted with a secret key generated by the user. Then, the encrypted speech is uploaded to the server. Finally, the model processes the encrypted speech data and returns the result to the user. The conventional method [10] encrypts the model itself to completely preserve its performance when processing the encrypted speech without decryption. Since the same secret key is used for encrypting both the speech and the model, key-sharing with the user is required. By avoiding model encryption, the proposed method removes the need for key sharing, offering a more convenient and user-friendly framework.

B. Encryption of speech data

Since the proposed method operates under the scenario described in II-A, it focuses on encrypting speech data. The encryption procedure of the proposed method first partitions the input speech data into segments of configurable length. The speech encryption is then achieved by randomly permuting the order of these segments and recombining them into a continuous waveform. The randomization of speech segments leads to a disjointed and incoherent utterance, concealing the speech

content. The proposed method segments speech into fixed-length segments and randomly permutes their order before recombining them into a continuous waveform. Despite this encryption process, we assume that the proposed method maintains speaker verification performance with minimal degradation. The underlying reason for this preservation lies in the architecture of modern deep learning-based speaker verification systems [11], [12]. These systems employ speaker embedding extraction techniques, such as Emphasized Channel Attention, Propagation and Aggregation in Time Delay Neural Network (ECAPA-TDNN) [13], which are specifically designed to capture speaker-specific characteristics. The speaker embedding extraction network consists of three main components. First, an encoder transforms acoustic features on a frame-by-frame basis. Second, a pooling layer aggregates the feature sequences obtained from the encoder across the temporal dimension, converting them into fixed-dimensional vectors. Finally, a speaker identification module estimates the speaker identity from the aggregated vectors, with the intermediate output serving as the speaker embedding vector. The key component that enables robustness to temporal disruption is the pooling layer within the speaker embedding extraction network. This layer is designed to aggregate information across the entire utterance in the temporal dimension. Therefore, even when the temporal continuity of speech content is partially disrupted, the network can still appropriately extract speaker-specific characteristics. Consequently, we expect that randomly permuting the order of speech segments will not significantly impact the quality of speaker embeddings. This architectural design allows the system to maintain speaker verification performance even when the original temporal structure of the speech is altered through our encryption process.

The procedure for encrypting speech data using segmentation and random permutation is explained. Here, the speech data should be understood to represent variable-length, one-dimensional data, such as a speech waveform.

- 1 The one-dimensional speech data \mathbf{X} is divided into segments of length M as:

$$\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_i, \dots, \mathbf{X}_K], \quad (1)$$

where T is the total data length, $\lceil T/M \rceil$. Note that if T is not divisible by M , the final segment has length $(T \bmod M)$.

- 2 An array \mathbf{S} of randomly permuted indices is generated as:

$$\mathbf{S} = [s(1), s(2), \dots, s(i), \dots, s(K)], \quad (2)$$

where $s(i) \in \{1, 2, \dots, K\}, s(i) \neq s(j), i, j \in \{1, 2, \dots, K\}, i \neq j$.

- 3 A $K \times K$ permutation matrix \mathbf{P} is created based

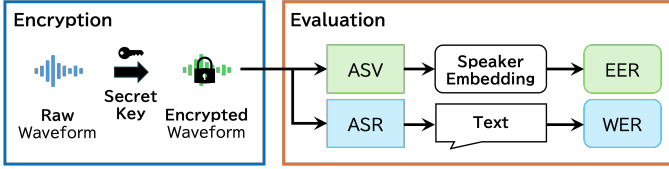


Fig. 3: Encryption and evaluation flow of the proposed method

on \mathcal{S} to randomly permute the segments.

$$P = \begin{pmatrix} a_{11} & \dots & a_{1j} & \dots & a_{1K} \\ \vdots & & \vdots & & \vdots \\ a_{i1} & \dots & a_{ij} & \dots & a_{iK} \\ \vdots & & \vdots & & \vdots \\ a_{K1} & \dots & a_{Kj} & \dots & a_{KK} \end{pmatrix} \quad (3)$$

$$a_{ij} = \begin{cases} 1 & (j = s(i)) \\ 0 & (\text{otherwise}) \end{cases} \quad (4)$$

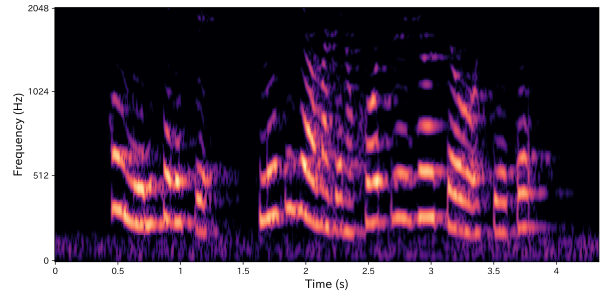
- 4 By multiplying the segmented speech data $\mathbf{X} = [\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_i, \dots, \mathbf{X}_K]$ on the right by the permutation matrix \mathbf{P} , the order of the segments is randomly permuted, resulting in the encrypted speech data \mathbf{X}' as

$$\mathbf{X}' = \mathbf{X}\mathbf{P}. \quad (5)$$

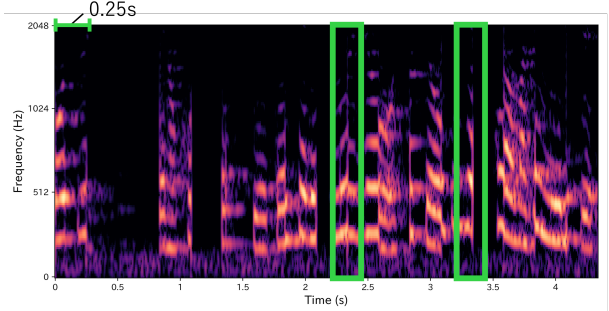
A key advantage of this method is that the model itself is not encrypted. This allows the system to be used without the need to convert the encrypted speech data back to its original form. Consequently, there's no need for secret key sharing between the model creator and the users.

IV. EXPERIMENT

In the experiments, the privacy-preserving performance of the proposed method was evaluated using two tasks: automatic speaker verification (ASV) and automatic speech recognition (ASR). The task of ASV involves determining whether a claimed speaker matches the true identity by comparing speech samples. For the ASV system, a pretrained RawNet3 model was used [14]. ASR is the process of transcribing the speech content into text. For the ASR task, the Google Speech API was used as the speech recognizer. The encryption and evaluation flow of the proposed method is illustrated in Figure 3. As depicted in Figure 3, the encrypted speeches were subjected to ASV and ASR, from which the evaluation metrics equal error rate (EER) and word error rate (WER) were computed. The objective of the proposed method is to achieve encryption that causes little degradation in EER and a significant increase in WER simultaneously. In all the experiments, the evaluation set of the J-SpAW corpus [15] was used. Since the proposed method allows for adjustable segment lengths, we investigated the performance across segment durations of 2.0, 1.0, 0.5, 0.25, and 0.1 seconds. Additionally, we also conducted trials where segments were randomly sized between 0.1 and 0.5 seconds (Mix). Although the segment lengths vary in the Mix



(a) Original speech



(b) Encrypted speech with 0.25-second segments

Fig. 4: Mel-spectrograms showing the effect of segment permutation

procedure, the procedure remained the same as with fixed-length segmentation. Furthermore, under the Mix condition, the impact on the privacy-preserving performance was investigated by randomly reversing longer segments. The minimum length of segments eligible for reversal was tested at 0.25, 0.3, 0.35, and 0.4 seconds. The probability of reversal was fixed at 25%.

Table 1 shows the EERs and the WERs for each segment duration when evaluating the encrypted speeches applied using the proposed method. Comparing the proposed method with the case without encryption “None”, the EERs of the proposed method increased slightly for the segment durations from 2.0 to 0.25 seconds, or under the Mix condition. To explain the slight increase in EER achieved by the proposed method, mel-spectrograms for the None case and the case with a segment duration of 0.25 seconds are shown.

Comparing the two mel-spectrograms, noise can be observed in Figure 4. This noise is presumed to result from the abrupt waveform changes or discontinuities caused by randomly reordering and reconnecting segments. In other words, this noise likely caused a slight degradation in speaker verification performance, even when the segment duration was reduced from 2.0 seconds to 0.25 seconds. These results indicate that the speaker's identity was preserved in the conditions. However, the EER significantly increased when the segment duration was 0.1 seconds, indicating the degradation in the performance of the ASV model. On the other hand, the segment shuffling of the proposed method leads to a significant increase in the WER under all conditions. This suggests that

TABLE I: EER (%) and WER (%) for Each Segment Duration (sec.)

Segment	None	2.0	1.0	0.5	0.25	0.1	Mix
EER	1.88	1.90	2.05	2.09	2.45	3.93	2.30
WER	3.30	52.06	74.14	68.41	90.51	98.71	80.30

TABLE II: EER (%) and WER (%) on the data with randomly reversed segments under the mix condition

		Minimum length for reversal				
		Mix	0.4	0.35	0.3	0.25
EER	Avg	2.30	2.59	2.86	2.95	2.96
	Var	0.0172	0.0052	0.0108	0.0007	0.0593
WER	Avg	80.30	83.17	83.82	84.61	84.95
	Var	27.7998	0.0069	0.0094	0.0195	0.0210

the speech content is effectively concealed. However, the WER alone fails to reveal word recognition accuracy, and there is an issue with evaluating the level of privacy that can be quantitatively measured. To investigate the issue, the ASR results were analyzed. From the analyses, when the segment duration was 0.5 seconds or longer, a part of individual words could still be recognized. This denotes the possibility of reconstructing the original sentence by rearranging the recognized words. On the other hand, only the wake-up words were recognized when the segment duration was 0.25 seconds or shorter, or under the Mix condition. This means most of the important words within the utterance were concealed, making it difficult to infer the speech content. Detailed examples of speech recognition results for different segment durations are provided in the Appendix. These results suggest that, when the segment duration is 0.25 seconds or under the Mix condition, the speech content can be concealed while the speaker’s identity is preserved in the proposed method.

Table 2 shows the results when segments were randomly reversed under the Mix condition. By incorporating the partial segmental reversal, all WERs were increased. This indicates that the partial segmental reversal works to conceal the speech content more than the Mix condition without the reversal. Whereas all EERs were slightly increased, it can still be regarded that the speaker’s identities are maintained. These results suggest that speaker-related information is robust to slight temporal distortions caused by partial segmental reversal. From the perspective of the variances in the WERs, by comparing the partial segmental reversal with the Mix condition without the reversal, the reversal yielded precisely low variances in the WERs, indicating more stable performance. Consequently, the proposed method demonstrated that segmenting speech data and randomly shuffling the segments allows for preserving speech privacy.

V. CONCLUSION

In this paper, we proposed a privacy-preserving method by randomly permuting the speech segments. The method does not require key sharing with others, and conceals the speech content while maintaining the speaker’s identity. In the experiments, the proposed method demonstrated to maintain

the EERs and increase the WERs. For future work, we will investigate model training with the encrypted speeches and conduct comprehensive security analysis and evaluation of the proposed method.

ACKNOWLEDGMENT

This work was supported in part by JSPS KAKENHI (Grant Number JP24K14993), SCAT, and ROIS DS-JOINT (026RP2025) to S. Shiota.

APPENDIX

SPEECH RECOGNITION TRANSCRIPTION EXAMPLES SHOWING THE EFFECT OF SEGMENT PERMUTATION ON SPEECH CONTENT CONCEALMENT

Segment Duration	Recognition Result
Original	hei merusedesu kafe ni tsurete itte
2.0	tsurete itte hei merusedesu
1.0	tsurete itte su kafe ni hei
0.5	kafe ni tsurete itte tsu he
0.25	he
0.1	No recognition
Mix	merusedesu

REFERENCES

- [1] H. Tabrizchi et. al., “A survey on security challenges in cloud computing: issues, threats, and solutions,” *Trans on. the journal of supercomputing*, vol. 76, no. 12, pp. 9493–9532, 2020.
- [2] A. Singh et. al., “Cloud security issues and challenges: A survey,” *Trans on. journal of network and computer applications*, vol. 79, pp. 88–115, 2017.
- [3] J. L. Kröger et. al., “Privacy implications of voice and speech analysis–information disclosure by inference,” *Trans on. privacy and identity management. data for better living: AI and privacy*, pp. 242–258, 2020.
- [4] G. Singh et. al., “Spoken language identification using deep learning,” *Trans on. computational intelligence and neuroscience*, no. 5123671, pp. 1–12, 2021.
- [5] N. Tomashenko et. al., “The voicePrivacy 2022 challenge evaluation plan,” *arXiv: 2203.12468*, 2020.
- [6] H. Kai et. al., “Lightweight and irreversible speech pseudonymization based on data-driven optimization of cascaded voice modification modules,” *Trans on. computer speech & language*, vol. 72, pp. 101315, 2022.
- [7] S.-X. Zhang et. al., “Encrypted speech recognition using deep polynomial networks,” In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5691–5695, 2019.
- [8] H. Kai et. al., “Robustness of signal processing-based pseudonymization method against decryption attack,” In *Proc. the speaker and language recognition workshop (Odyssey)*, pp. 287–293, 2022.
- [9] F. Teixeira et. al., “Towards End-to-End private automatic speaker recognition,” In *Proc. Interspeech*, pp. 2798–2802, 2022.
- [10] S. Niwa et. al., “Speech privacy-preserving methods using secret key for convolutional neural network models and their robustness evaluation,” *Trans on. APSIPA*, vol. 13, no. 1, p. e26, 2024.
- [11] D. Snyder et. al., “Deep neural network embeddings for text-independent speaker verification,” In *Proc. Interspeech*, pp. 999–1003, 2017.
- [12] G. Heigold et. al., “End-to-End text-dependent speaker verification,” In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5115–5119, 2016.
- [13] B. Desplanques et. al., “ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN based speaker verification,” In *Proc. Interspeech 2020*, pp. 3830–3834, 2020.
- [14] J.-W. Jung et. al., “Pushing the limits of raw waveform speaker recognition,” In *Proc. Interspeech*, pp. 2228–2232, 2022.
- [15] S. Shiota et. al., “J-SpAW: Japanese speaker verification and spoofing attacks recorded in-the-wild dataset,” In *Proc. Interspeech 2025(accepted)*.