🍸🗣️: Continuous Speaker Shifting for Obviously Anonymized Speech

Xuankai Chang*, Jessica Huynh*, Jeremiah Milbauer*, Jiatong Shi*
Language Technology Institute
Carnegie Mellon University
Pittsburgh, PA 15213
{xuankaic, jhuynh, jmilbaue, jiatongs}@cs.cmu.edu

Abstract

The development of artificial intelligence not only brings convenience to our life but also concerns of privacy exposure. We are able to easily share information using audio and videos. It is important to maintain the privacy of the participants from these media while preserving the original information to be conveyed as much as possible. There are some existing methods that anonymize speech using signal processing or machine learning techniques. However, most of them result in either poor quality of the audio, poorly anonymized, or not obviously anonymized. In this paper, we would like to propose a novel method to convert the speech from one speaker to some other speaker identity through the pipeline: speech → linguistic units → speech. With our proposed cocktail speaker decoding (CSD), we can also dynamically change the speaker identity without re-training the network. The model can be applied to many scenarios, such as generating the speech in a standardized speaking style for all the operators in call center services.

1 Introduction

The increased prevalence of smartphones, as well as voice-activated smart home devices, has created a world in which millions of devices within arms reach of people are always on and, potentially, always listening [15]. Additionally, the rise of video-based social media platforms, such as Snapchat and TikTok, has resulted in millions of users sharing recordings of their voice around the world without control over how those recordings are used [19].

Increasingly, it is becoming important to provide consumers with sophisticated options to anonymize their voices in order to guarantee that: 1) the speech recognition component of smart home and virtual assistant services cannot match a user's activity with their identity; and 2) customers have the option to anonymize their speech when uploading content to social media. However, many existing approaches to anonymization produce audio that is either poor quality (and therefore not suitable for use in downstream speech recognition systems), poorly anonymized, not obviously anonymized (such that the voice cannot be immediately perceived as not coming from a real person), and characterless.

To overcome these issues, we propose a novel framework for speech anonymization, called the **cocktail speaker decoding** (CSD) model. Our approach first encodes a speech signal to a form devoid of identifying speaker characteristics and then synthesizes speech from the encoded form. During decoding, the synthesized speaking style is continually modified by adaptively changing the speaker embedding given to the decoder, which will result in

*Equal contribution; sorted alphabetically

a cocktail of ever-changing different human voices within the same utterance. Without retraining the network, this results in speech that is natural, clean, anonymous, and interesting to listen to. Our empirical studies on LibriTTS indicate that the proposed CSD could outperform other methods on subjective evaluation and a fair performance over objective scores.

2 Literature Review

2.1 Privacy

Speech is more identifiable than text; therefore, there are many privacy issues concerning the use of and release of speech data. Researchers have started to address these issues by trying to eliminate the identifiable characteristics in a person's speech so that their speech cannot be traced back to them [16]. One work has used a minimax filter to balance between the downstream task and the privacy the model provides [7]. Another work directly works to provide a better automatic speech recognition (ASR) system without sacrificing privacy [1]. Yet another changes both the text and the voice of the speaker for even more privacy [20]. The VoicePrivacy 2020 Challenge [24] is also striving to push for anonymization; one system that came out of this perturbed the x-vector of the speaker [23].

2.2 ASR Models

There are many off-the-shelf ASR models that are available: Amazon Transcribe, Cloud Speech-to-Text, and Kaldi models [4]. Researchers have also refined other ASR models which are publicly available that have word error rates (WERs) of below 4% [27]. One paper goes directly into the architecture of two multi-layer ASR models and determines how much speaker information is lost in each layer of the model [14]. The speaker information decreases as the model progresses, but the noise increases. Even though the model cannot figure out the speaker, it may be because the speech representation becomes too noisy and unusable for the model.

2.3 Voice Conversion (VC)

One method of reducing speaker identifiability in systems is to convert the voice of the speaker into the voice of another. This could be a single voice that has been chosen as a "golden voice", or a combination of voices into one. VC has recently been studied, and naturalness is a big factor that these systems are still lacking in [30]. Deep learning and statistical modeling have both been used as well [22].

2.4 Knowing a Voice is Synthesized

Along with voice synthesis, there comes the issue of knowing that a synthesized voice is not real. People do prefer more human-like voices [13], but the way they interact with systems is different than how they interact with other people. For example, the sentences are simpler, or the system may not understand when someone rephrases their command, among others [12]. Therefore, it is still important to indicate to a listener that the voice they are listening to, say in a dialogue system, is synthesized while keeping it as human-like as possible so that the listener is informed and can make decisions on how to proceed.

3 Baselines

In this section, we briefly introduce two baseline systems: cascaded voice conversion and HuBERT representation. They both follow the same mathematical formulation:

Denote X as speech utterance and s as it's corresponding speaker. The ground truth transcription is defined as L. The baseline models first extract linguistic information $\hat{L}$ with an ASR module as:

$$\hat{L} = \text{ASR}(X) \tag{1}$$

The linguistic information $\hat{L}$ can be either a text transcription, phoneme sequence, or hidden representation (e.g., hidden representations from ASR or self-supervised speech representations). In the next step, we perform a TTS module, conditioning on another speaker A's information.

$$\hat{X} = \text{TTS}(L, i_a) \tag{2}$$

where we note i_a as the speaker identity vector (e.g., speaker embedding) and a is a different speaker from s. In the case of handling unknown speakers, the i_a is usually extracted from an i-vector or x-vector algorithm, which does not depend on the training set speaker IDs.

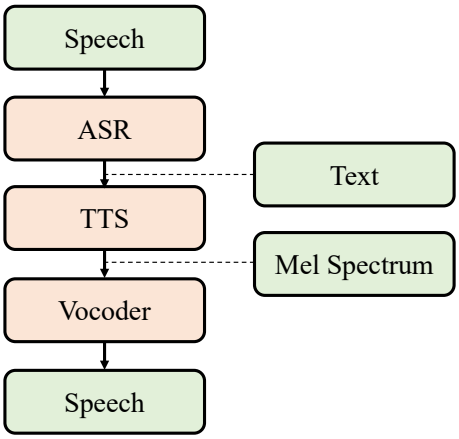

Figure 1: The framework for cascaded voice conversion discussed in Section 3.1

3.1 Cascaded Voice Conversion

VC is a good candidate for privacy protection during which we could change the speaker identity of a speech signal [22]. Cascaded voice conversion takes benefits from advances in both ASR and TTS, which have been shown to reach state-of-the-art performances in recent challenges [29]. A cascaded VC system often consists of an ASR module and a text-to-speech (TTS) module [11, 10]. The L in the VC systems is textual information (e.g., text or phoneme sequence). For VC purposes, the TTS module conditions on a dense speaker representation (e.g., x-vector [3]). During the training phase, the TTS module follows the typical TTS training procedures. However, during inference, the module accepts another speaker representation from another random speaker to support voice conversion. Following related works [29], the framework of our baseline system is shown in Figure 1. For each module shown in the Figure 1, we leverage the state-of-the-art models from ESPNet [25, 8, 9] and its corresponding vocoder toolkit.[2]

4 Cocktail Speaker Decoding (CSD)

As discussed in Section 1, we argue that it is also important to generate speech that is obviously anonymized. However, this purpose is different from the training objectives in baseline systems introduced in Section 3.1. If the baseline has good performance, it would be likely to be difficult for listeners to notice the speech is anonymized. On the other hand, if the baseline does not result in reasonable quality, the speech will have the risk of losing speech intelligibility and naturalness.

To mitigate the aforementioned issue, we propose a decoding strategy, namely cocktail speaker decoding (CSD), before TTS decoding within the cascaded VC module, shown in Figure 2. In the baselines explained in Section 3.1, we adopt a TTS to re-synthesize the speech from its corresponding linguistic information and another speaker A's embedding i_a to perform voice conversion. As illustrated in Figure 2(a), the speaker embedding is repeated

[2] `https://github.com/kan-bayashi/ParallelWaveGAN`

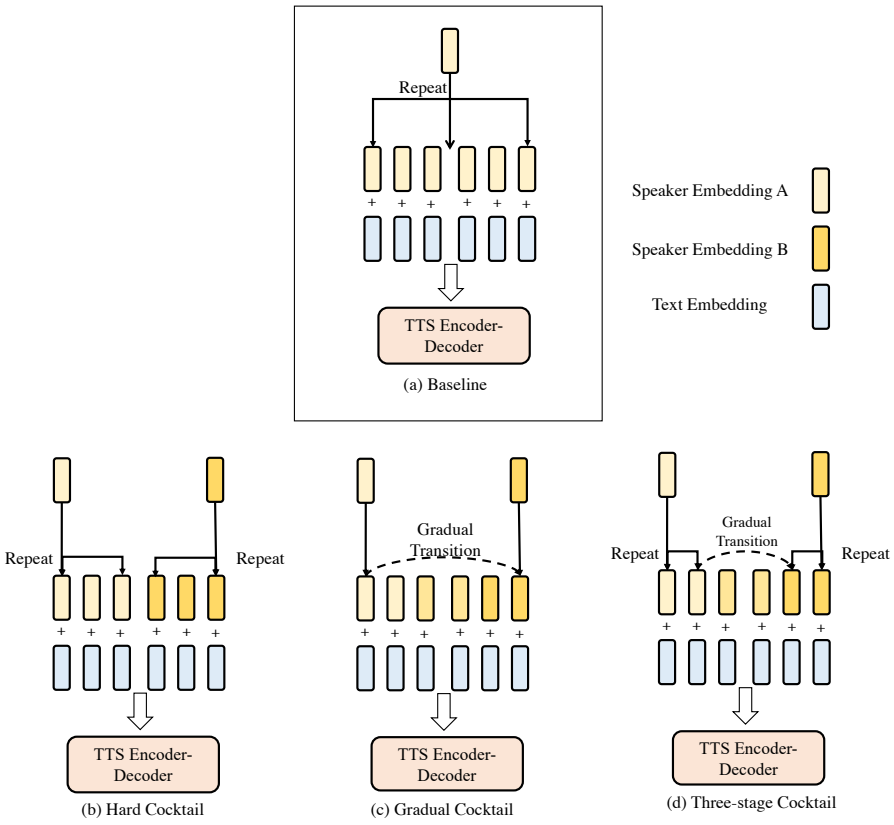

Figure 2: TBD

to align linguistic information $\hat{L}$. However, in CSD, we introduce other unknown speakers and cocktail their embeddings with i_a. For simplicity, we only add one additional speaker compared to the baseline.

We propose three ways to perform CSD as show in Figure 2(b)-(d). If we note the length of $\hat{L}$ as l. The first (Figure 2(b)) is simply adopting i_a from speaker A for the first half $\hat{L}_{1:l/2}$ and i_b from speaker B for the second half of the linguistic information $\hat{L}_{l/2:l}$. The second (Figure 2(c)) uses i_a for the first linguistic unit and i_b for the last linguistic unit. We define the gradual change factor as $(i_b - i_a)/l$, then for each unit:

$$i_t = i_a + \frac{t \cdot (i_b - i_a)}{l} \tag{3}$$

where i_t is the speaker embedding vector for $\hat{L}_t$. For the last (Figure 2(d)), we combine both the first two methods. Specifically, we divide $\hat{L}$ into three even sections. For the first and last part, we adopt i_a and i_b, respectively. Then for the middle part, we employ the gradual transition as noted in Eq. (3).

5 Experiments

5.1 Datasets

To train our models, we mainly use two publicly available datasets, LibriSpeech [17] and LibriTTS [28]. LibriSpeech, a corpus containing around 1,000 hours of reading speech, is usually used in the ASR task. The training set of LibriSpeech is split into three subsets: train-clean-100, train-clean-360, and train-other-500. Another essential part of our approach is the TTS module, for which we will use LibriTTS. The audios in LibriTTS are derived from those in LibriSpeech, with a higher sampling rate of $24kHz$ which benefits TTS research.

As the ASR trained on LibriSpeech is used in the Cascaded VC baseline, we need another ASR model for the fairness of the ASR evaluation. Therefore, we also adopt a pre-trained ASR model over the Gigaspeech corpus[2]. Gigaspeech is a multi-domain English ASR corpus with around 10,000 hours of speech, which should have enough generalizability for evaluation.

5.2 Baselines

5.2.1 Cascaded Voice Conversion

As discussed in Section 3.1, we employ three modules for the pipeline system.

For ASR, we use a conformer-based encoder-decoder architecture trained with hybrid CTC/Attention [26, 5, 6]. The encoder of the model has 12-layer conformer blocks with a convolutional kernel of 31. For the decoder, six transformer blocks are used. Both conformer and transformer blocks have 8-head attention with 2048-dimensional linear layers. The dropout rate is set to 0.1 for both attention weights and linear layers. We also use spec-augmentation [18] and speed perturbation for data augmentation. The training optimizer is Adam with a 0.0015 learning rate, while the schedule is with warm-up steps of 25,000. We use a bin-based sampler with a bin size of 140 million (corresponding to sampling rate) and train with 35 epochs.

For TTS module, we use Fastspeech2 [21]. The model consists of three modules apart from the initial phoneme embedding: an encoder, a variance adapter, and a Mel-spectrogram decoder. The encoder converts the phoneme embedding sequence into hidden representations. Later the variance adaptor adds different variance information into the hidden sequence, including the information of duration, pitch, and energy, respectively. The last decoder converts the adapted hidden sequence into Mel-spectrogram. In addition to the original settings in the Fastspeech2, we replace the Transformer encoder into Conformer encoder as ESPNet's implementation [8]. We employ four layers for both encoder and decoder, while for the duration predictor, we apply a three-layer convolutional network. Detailed settings follow the configuration in ESPNet.[3]

For x-vector computation, we adopt Kaldi's pre-trained DNN [3].

5.3 Experimental Setups

For the baseline, we randomly select a speaker A from LibriTTS training set as the target speaker for the voice conversion. Following the baseline settings, we conduct three experiments for each strategy discussed in Section 4. For simplicity, we just use two speakers as external speaker A[4] and B, while the first is from a female voice and the second is from a male voice. Both are randomly drawn from LibriTTS's training set. For ablation studies, we also incorporate two other experiments, which only use speaker A's identity. The first, namely as perturbed x-vectors, add random noise to the x-vector and then repeat it as the baseline. The second, called perturbed x-vector over time, first repeats the x-vector to align with the linguistic representation $\hat{L}$ and then applies different random noise to x-vectors over time. The random noise is drawn from a Gaussian distribution with a standard deviation of five.

5.4 Evaluation Metrics

Following the discussion in Section 1 on privacy concerns, we argue four evaluation factors: speaker verifiability, speech intelligibility, speech naturalness, and speech anonymization awareness.

Speaker Verifiability: the main purpose of this task is to remove the speaker identity information from the speech. Therefore, we will apply the common metric used in speaker verification tasks: equal-error-rate (EER). This corresponds to the point where the false discovery rate equals the false omission rate. To calculate EER, we have a trials file consisting of pairs of utterances for evaluation. We consider a pair of utterances spoken by two ground

[3] `https://github.com/espnet/espnet/blob/master/egs2/libritts/tts1/conf/tuning/train_xvector_conformer_fastspeech2.yaml`

[4] the same speaker as baseline

truth speakers as well as two generated speakers as correct speaker identifications. This trials file is balanced, which means the correct and incorrect speaker identifications are equal (downsampling the incorrect speaker identification to match).

Speech intelligibility: to measure how well the linguistic information is kept, we will do the speech recognition for the generated speech. In speech recognition, the predicted transcription is referred to as the hypothesis (Hyp), while the ground truth transcription is referred to as the reference (Ref) with length N. The metric used is the word-error-rate (WER), which is computed as:

$$\text{WER} = \frac{\text{EditDistance}(Hyp, Ref)}{N}. \tag{4}$$

In the ASR evaluation, we adopt the same model architectures as the ASR model defined in Sec 5.2.1, but trained with Gigaspeech [2].

Naturalness and Anonymization Awareness: the generated speech should be natural and clearly anonymized. To evaluate this, we will use the mean-opinion-scores (MOS) from two aspects: 1) speech quality and 2) anonymization awareness of speech. The authors evaluated these two aspects from a score of 1 to 5 for 20 randomly chosen utterances from each of the six experiments in addition to the ground truth for a total of 140 utterances. The authors did not know which utterance was from which experiment during evaluation. A score of 1 for the speech quality denotes bad speech quality, which includes additional noise and warping, while a score of 5 for the speech quality denotes minimal to no noise and the utterance sounding like it came from a person. For fakeness of speech, a score of 1 corresponds to the ability to be able to tell that the speech was real and came from one person, while a score of 5 is the ability to clearly detect that it is generated speech and cannot be spoken by a single identifiable person.

5.5 Available Examples

We have posted some examples from our models in `https://github.com/ftshijt/DL21_samples`.

6 Results

For each of the six methods, in addition to the ground truth, we perform our evaluations in Tables 1, 2, and 3.

Table 1: EER of the generated speech compared with the ground truth

Model	EER (%)
Hard Cocktail	**43.17**
Gradual Cocktail	29.63
Three-stage Cocktail	40.29
Baseline	26.88
Perturbed x-vectors	38.4
Perturbed x-vector across time	**44.18**

7 Discussion

In Table 1, the best performing experiment is the perturbed x-vector across time, followed closely by the hard cocktail. The three-stage cocktail performs well also, but the gradual cocktail's performance is very low. The baseline performs the worst as expected, but this shows that the cocktail methods do work in not allowing for the speaker verifiability of the speech.

Table 2: WER of the generated speech compared with the ground truth

Model	WER (%)
Hard Cocktail	34.4
Gradual Cocktail	39.5
Three-stage Cocktail	36.7
Baseline	60.5
Perturbed x-vectors	31.6
Perturbed x-vectors over time	45.6
Ground truth	30.8

Table 3: Speech Quality and Anonymization Awareness of the generated speech

Model	Speech Quality	Anonymization Awareness
Hard Cocktail	3.50	4.38
Gradual Cocktail	3.41	2.9625
Three-stage Cocktail	3.56	4.49
Baseline	1.83	3.59
Perturbed x-vectors	3.54	2.56
Perturbed x-vectors over time	2.63	3.09
Ground truth	4.94	1.01

With regards to the word error rate in Table 2, the three original methods performed better than the baseline, which is surprising since it is expected that the baseline will provide the best performance. The WER is comparable between the three and slightly lower for the perturbed x-vector over time. Since there is no intermediate blurring of the speaker's x-vector for the hard cocktail, it would make sense for the speech to have fewer word errors. The random noise added in the perturbed x-vectors was also consistent across the utterance so that the lower WER would make sense as well.

Lastly, the speech quality for the three methods is above average on the 1 to 5 scale in Table 3. The anonymization awareness is also quite high, with the three-stage cocktail performing the best in both metrics. Overall, the authors believed that the three-stage cocktail worked the best in terms of speech quality and anonymization awareness compared to all of the other methods. Even though there was noise in the generated speech, it was generally obvious that the speech was generated and not from one speaker. The lower anonymization awareness from the gradual cocktail may have resulted from the inability to tell the gradual shift between the two speakers. The high anonymization awareness from the baseline may have been from the lower speech quality of the resulting speech, as it is the lowest of all the experiments, which would give anonymization a higher score since the speech would be heard as more obviously generated.

8 Conclusion

We have demonstrated a new approach for text-to-speech decoding, *Cocktail Speaker Decoding*. By transitioning between speakers during the generation process, *CSD* enables a TTS system to produce high-quality speech while also remaining transparently computer-generated.

Our new approach has important applications in speech anonymization. While improvements in TTS quality have allowed for truly anonymous speech, these improvements simultaneously motivate a desire to guarantee that the anonymized speech is understood to be computer-generated. Additionally, because of the high fidelity with which modern methods can

reproduce the speech sounds of a target speaker, it is important to both disguise the target speaker's voice (to disguise the source of the training data for the anonymization system) and make it clear that it is not *actually* the target speaker who is speaking. Additionally, *CSD* maintains the transparency about the disguised nature of speech that previous signal-based approaches to speech anonymization provide.

Additionally, *CSD* may have additional uses in general-purpose speech generation. As speech-based interaction becomes both higher-quality and more prevalent, it is important to preserve transparency because users interact differently with computer systems than they do other humans [12]. We also hope that *CSD* can provide a foundation for further research in speech-based human-computer interaction, enabling completely new approaches to personality design for virtual assistants. We hope that our *CSD* will open new possibilities for both speech anonymization and speech-based human-computer interaction at large.

References

[1] Shimaa Ahmed, Amrita Roy Chowdhury, Kassem Fawaz, and Parmesh Ramanathan. Preech: a system for privacy-preserving speech transcription. In *29th {USENIX} Security Symposium ({USENIX} Security 20)*, pages 2703–2720, 2020.

[2] Guoguo Chen, Shuzhou Chai, Guanbo Wang, Jiayu Du, Wei-Qiang Zhang, Chao Weng, Dan Su, Daniel Povey, Jan Trmal, Junbo Zhang, et al. Gigaspeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio. *arXiv preprint arXiv:2106.06909*, 2021.

[3] Daniel Garcia-Romero, David Snyder, Gregory Sell, Alan McCree, Daniel Povey, and Sanjeev Khudanpur. x-vector dnn refinement with full-length recordings for speaker recognition. In *Interspeech*, pages 1493–1496, 2019.

[4] Kallirroi Georgila, Anton Leuski, Volodymyr Yanov, and David Traum. Evaluation of off-the-shelf speech recognizers across diverse dialogue domains. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6469–6476, 2020.

[5] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al. Conformer: Convolution-augmented transformer for speech recognition. *arXiv preprint arXiv:2005.08100*, 2020.

[6] Pengcheng Guo, Florian Boyer, Xuankai Chang, Tomoki Hayashi, Yosuke Higuchi, Hirofumi Inaguma, Naoyuki Kamo, Chenda Li, Daniel Garcia-Romero, Jiatong Shi, et al. Recent developments on espnet toolkit boosted by conformer. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5874–5878. IEEE, 2021.

[7] Jihun Hamm. Preserving privacy of continuous high-dimensional data with minimax filters. In *Artificial Intelligence and Statistics*, pages 324–332. PMLR, 2015.

[8] Tomoki Hayashi, Ryuichi Yamamoto, Katsuki Inoue, Takenori Yoshimura, Shinji Watanabe, Tomoki Toda, Kazuya Takeda, Yu Zhang, and Xu Tan. Espnet-tts: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit. In *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 7654–7658. IEEE, 2020.

[9] Tomoki Hayashi, Ryuichi Yamamoto, Takenori Yoshimura, Peter Wu, Jiatong Shi, Takaaki Saeki, Yooncheol Ju, Yusuke Yasuda, Shinnosuke Takamichi, and Shinji Watanabe. Espnet2-tts: Extending the edge of tts research. *arXiv preprint arXiv:2110.07840*, 2021.

[10] Wen-Chin Huang, Tomoki Hayashi, Xinjian Li, Shinji Watanabe, and Tomoki Toda. On prosody modeling for asr+ tts based voice conversion. *arXiv preprint arXiv:2107.09477*, 2021.

[11] Wen-Chin Huang, Tomoki Hayashi, Shinji Watanabe, and Tomoki Toda. The sequence-to-sequence baseline for the voice conversion challenge 2020: Cascading asr and tts. *arXiv preprint arXiv:2010.02434*, 2020.

[12] Irina Kondratova and Bruno Emond. Voice interaction for training: Opportunities, challenges, and recommendations from hci perspective. In *International Conference on Human-Computer Interaction*, pages 59–75. Springer, 2020.

[13] Katharina Kühne, Martin H Fischer, and Yuefang Zhou. The human takes it all: Humanlike synthesized voices are perceived as less eerie and more likable. evidence from a subjective ratings study. *Frontiers in neurorobotics*, 14:105, 2020.

[14] Chung-Yi Li, Pei-Chieh Yuan, and Hung-Yi Lee. What does a network layer hear? analyzing hidden representations of end-to-end asr through speech synthesis. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6434–6438. IEEE, 2020.

[15] Sapna Maheshwari. Hey, alexa, what can you hear? and what will you do with it?, 2018.

[16] Andreas Nautsch, Abelino Jiménez, Amos Treiber, Jascha Kolberg, Catherine Jasserand, Els Kindt, Héctor Delgado, Massimiliano Todisco, Mohamed Amine Hmani, Aymen Mtibaa, et al. Preserving privacy in speaker and speech characterisation. *Computer Speech & Language*, 58:441–480, 2019.

[17] 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)*, pages 5206–5210. IEEE, 2015.

[18] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

[19] Sarah Perez. Tiktok just gave itself permission to collect biometric data on us users, including 'faceprints and voiceprints', 2021.

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

[21] Yi Ren, Chenxu Hu, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. Fastspeech 2: Fast and high-quality end-to-end text to speech. *arXiv preprint arXiv:2006.04558*, 2020.

[22] Berrak Sisman, Junichi Yamagishi, Simon King, and Haizhou Li. An overview of voice conversion and its challenges: From statistical modeling to deep learning. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2020.

[23] Kyoto Team, Yaowei Han, Sheng Li, Yang Cao, and Masatoshi Yoshikawa. System description for voice privacy challenge.

[24] Natalia Tomashenko, Brij Mohan Lal Srivastava, Xin Wang, Emmanuel Vincent, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Jose Patino, Jean-François Bonastre, Paul-Gauthier Noé, et al. The voiceprivacy 2020 challenge evaluation plan, 2020.

[25] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, et al. Espnet: End-to-end speech processing toolkit. *arXiv preprint arXiv:1804.00015*, 2018.

[26] Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R Hershey, and Tomoki Hayashi. Hybrid ctc/attention architecture for end-to-end speech recognition. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1240–1253, 2017.

[27] Qiantong Xu, Alexei Baevski, Tatiana Likhomanenko, Paden Tomasello, Alexis Conneau, Ronan Collobert, Gabriel Synnaeve, and Michael Auli. Self-training and pre-training are complementary for speech recognition. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3030–3034. IEEE, 2021.

[28] Heiga Zen, Viet Dang, Rob Clark, Yu Zhang, Ron J Weiss, Ye Jia, Zhifeng Chen, and Yonghui Wu. Libritts: A corpus derived from librispeech for text-to-speech. *arXiv preprint arXiv:1904.02882*, 2019.

[29] Jing-Xuan Zhang, Li-Juan Liu, Yan-Nian Chen, Ya-Jun Hu, Yuan Jiang, Zhen-Hua Ling, and Li-Rong Dai. Voice Conversion by Cascading Automatic Speech Recognition and Text-to-Speech Synthesis with Prosody Transfer. In *Proc. Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020*, pages 121–125, 2020.

[30] Yi Zhao, Wen-Chin Huang, Xiaohai Tian, Junichi Yamagishi, Rohan Kumar Das, Tomi Kinnunen, Zhenhua Ling, and Tomoki Toda. Voice conversion challenge 2020: Intra-lingual semi-parallel and cross-lingual voice conversion. *arXiv preprint arXiv:2008.12527*, 2020.