PHD THESIS

In Partial Fulfilment of the Requirements for the
Degree of Doctor of Philosophy from Sorbonne University

Voice Anonymization: from Vocoder Drift to Neural Audio Codecs and Robust Evaluation

Michele PANARIELLO

Defended on 15 December 2025 before a committee composed of:

Reviewer	**Martha LARSON**, Radboud University, Nijmegen, Netherlands
Reviewer	**Mickael ROUVIER**, Avignon University, Avignon, France
Examiner	**Nicolas GENGEMBRE**, Orange, Rennes, France
Examiner	**Xiaoxiao MIAO**, Duke Kunshan University, Kunshan, China
Examiner	**Paolo PAPOTTI**, EURECOM, Sophia Antipolis, France
Thesis Director	**Nicholas EVANS**, EURECOM, Sophia Antipolis, France
Thesis Co-Director	**Massimiliano TODISCO**, EURECOM, Sophia Antipolis, France

Abstract

Voice is a uniquely rich biometric. The same recordings that fuel speech technologies also expose identity and sensitive attributes, which can be exploited for impersonation and fraud. This dissertation addresses these risks through voice anonymization (VA): the speaker's privacy is protected by transforming their voice identity into that of a pseudo-speaker, while preserving the utility of the audio for downstream use. This is typically achieved by means of voice conversion (VC) or analogous speech synthesis techniques.

In the VoicePrivacy Challenge (VPC), now at its third edition, VA is formalized as hiding voice identity while retaining linguistic and paralinguistic content. Privacy is evaluated against an attacker attempting to re-identify anonymized data through automatic speaker verification (ASV) models adapted to the anonymized domain. Utility is assessed using estimates of the word error rate (WER), complemented with secondary metrics that assess preservation of either prosody, voice distinctiveness or emotion. In this thesis, we describe the analysis of the results of the second VPC edition conducted by the candidate as one of the VPC organizers.

We then present novel research on VA. We first examine x-vector–based pipelines, whereby an x-vector is extracted from unprotected audio and anonymized into a pseudo-speaker embedding through some perturbation function, then supplied to a vocoder for synthesis along with other features. We then introduce *vocoder drift*: the discrepancy between the pseudo-speaker x-vector and that extracted from the anonymized waveform. Drift represents the vocoder's contribution to the anonymization strength — which, as we show, can be substantial. An adversary can exploit this behavior via *drift-reversal* attacks, approaching the effectiveness of then state-of-the-art attacks. We trace vocoder drift to input feature mismatch: speaker cues leaking through non-anonymized inputs (e.g., linguistic features) are re-entangled with anonymized x-vectors and cause unexpected vocoder behavior, resulting in drift. We conclude that the erasure of speaker attributes should not rely on x-vector perturbation, but on effective VC.

To that end, we propose anonymization with *neural audio codec* (NAC) language modeling. Speech is represented as a stream of concatenated discrete semantic and acoustic tokens; a language model, conditioned on a pseudo-speaker voice, performs voice conversion and generates a new tokens that are decoded to a waveform via the NAC. The approach yields strong privacy, distinctiveness, and pitch preservation, but initially modest WER. We then propose *character-level conditioning*, which injects textual information into the vocoder and improves the WER near to that of unprotected speech. The NAC system was adopted as a VPC 2024 baseline. These findings support a view of VA as a “down-sampling and up-sampling” task, with the former responsible for privacy protection, and the latter for utility preservation.

Finally, we study the importance of the *attacker’s model* in privacy evaluation. We define and document *privacy overestimation*: spuriously high EERs caused by suboptimal attacker training. We describe toy examples, identify cases in the literature, correct one, and propose a simple diagnostic to detect potential privacy overestimation in VA evaluation. Based on the previous observations, we propose to strengthen VA by *forcing* the attacker into a suboptimal training condition through the addition of adversarial noise to unprotected utterances. Contrary to unintentional overestimation, this represents the first approach to *proactive defense* in VA.

We conclude with general considerations and suggestions for future research. We highlight how these findings fit into the general research trends in VA as the field continues to grow.

Abrégé

La voix est une biométrie d'une richesse singulière. Les mêmes enregistrements qui alimentent les technologies vocales exposent aussi l'identité et des attributs sensibles, pouvant être exploités pour l'usurpation et la fraude. Cette thèse s'attaque à ces risques par la *voice anonymization* (VA) : la vie privée du locuteur est protégée en transformant son identité vocale en celle d'un pseudo-locuteur, tout en préservant l'utilité de l'audio pour les usages en aval. Cela est typiquement réalisé au moyen de *voice conversion* (VC) ou de techniques analogues de synthèse de la parole. Dans le cadre de la VoicePrivacy Challenge (VPC), aujourd'hui à sa troisième édition, la VA est formalisée comme la dissimulation de l'identité du locuteur tout en conservant le contenu linguistique et paralinguistique. La confidentialité est évaluée face à un attaquant tentant de ré-identifier des données anonymisées au moyen de modèles d'*Automatic Speaker Verification* (ASV) adaptés au domaine anonymisé. L'utilité est principalement mesurée par le *Word Error Rate* (WER), complété par des métriques secondaires évaluant la préservation de la prosodie, de la distinctivité vocale ou de l'émotion. Dans cette thèse, nous décrivons l'analyse des résultats de la deuxième édition de la VPC réalisée par le candidat en tant que l'un des organisateurs de la VPC.

Nous présentons ensuite des contributions originales en VA. Nous examinons d'abord des chaînes fondées sur les *x-vector*, où un x-vector est extrait d'un signal non protégé, anonymisé en un pseudo-locuteur via une fonction de perturbation, puis fourni au vocodeur avec d'autres caractéristiques pour la synthèse. Nous introduisons alors la *vocoder drift* : l'écart entre le x-vector du pseudo-locuteur et celui que l'on peut extraire de l'onde anonymisée après synthèse. Cette dérive reflète la contribution du vocodeur à la force d'anonymisation — contribution que nos expériences montrent comme considérable. Nous démontrons qu'un adversaire peut l'exploiter via des attaques de *drift-reversal*, atteignant une efficacité proche de l'état de l'art d'alors. Nous attribuons la *vocoder drift* à un *feature mismatch* en entrée : des indices de locuteur fuyant par des entrées non anonymisées (p. ex. caractéristiques linguistiques) se ré-entrelacent avec les x-vectors anonymisés et induisent des comportements inattendus du vocodeur, produisant la dérive. Nous en concluons qu'en raison de cette non-fiabilité, l'effort de recherche doit se déplacer de la simple perturbation des x-vector vers des techniques de VC plus efficaces.

Dans cette optique, nous proposons une anonymisation fondée sur le *neural audio codec* (NAC) et le *language modeling*. La parole est représentée comme un flux de jetons sémantiques et acoustiques concaténés en un *prompt*; un modèle de langue, conditionné par une voix de pseudo-locuteur, réalise la conversion vocale en générant de nouveaux jetons, ensuite décodés en onde par le NAC. Cette approche offre une forte confidentialité, une bonne distinctivité et une bonne préservation de la hauteur, mais initialement un WER modeste ; nous introduisons alors une *character-level conditioning* qui injecte des informations textuelles dans le vocodeur et améliore le WER jusqu'à se rapprocher de celui de la parole non protégée. Le système NAC a été adopté comme *baseline* de la VPC 2024. Ces résultats soutiennent une vision de la VA comme un processus de "down-sampling / up-sampling", le premier assurant la protection de la vie privée, le second la préservation de l'utilité.

Enfin, nous étudions le *rôle de l'attaquant* dans l'évaluation. Nous définissons et documentons la *privacy overestimation* : des *Equal-Error Rate* (EER) artificiellement élevés dus à un apprentissage sous-optimal de l'ASV. Nous décrivons des exemples jouets, identifions des cas dans la littérature, en corrigeons un, et proposons un diagnostic simple pour détecter une sur-estimation potentielle de la confidentialité dans l'évaluation de la VA. Forts de ces observations, nous proposons de renforcer la VA en *forçant* l'attaquant à un apprentissage sous-optimal par l'ajout de perturbations adversariales à des énoncés non protégés. Contrairement à la sur-estimation involontaire, cela constitue la première approche de *défense proactive* en VA.

Nous concluons par des considérations générales et des pistes pour de futurs travaux. Nous soulignons enfin la place de ces résultats dans les tendances actuelles de la recherche, alors que le domaine de la VA continue de se développer.

Contents

List of Figures

List of Tables

List of Abbreviations

$G_{\mathbf{VD}}$ Gain of Voice Distinctiveness.

AAM Additive Angular Margin.

AI Artificial Intelligence.

AM Acoustic Model.

ASR Automatic Speech Recognition.

ASV Automatic Speaker Verification.

BN Bottleneck.

CTC Connectionist Temporal Classification.

DeID De-Identification.

DP Differential Privacy.

DTW Dynamic Time Warping.

EER Equal Error Rate.

GAN Generative Adversarial Network.

GDPR General Data Protection Regulation.

GMM Gaussian Mixture Model.

GRU Gated Recurrent Unit.

GST Global Style Token.

k-NN k-Nearest Neighbors.

LAM Large Audio Model.

LLM Large Language Model.

LLR Log-Likelihood Ratio.

LSTM Long-Short Term Memory.

MSE Mean Squared Error.

NAC Neural Audio Codec.

NLP Natural Language Processing.

NSF Neural Source-Filter.

OHNN Orthogonal Householder Neural Network.

PESQ Perceptual Evaluation of Speech Quality.

PGD Projected Gradient Descent.

PLDA Probabilistic Linear Discriminant Analysis.

SER Speech Emotion Recognition.

SNR Signal-to-Noise Ratio.

SPSC Security and Privacy in Speech Communication.

SS Speech Synthesis.

SSL Self-Supervised Learning.

STFT Short-time Fourier Transform.

STOI Short-Time Objective Intelligibility.

TDNN Time-Delay Neural Network.

TDNN-F Factorized Time-Delay Neural Network.

TTS Text-to-Speech.

UAR Unweighted Average Recall.

VA Voice Anonymization.

VC Voice Conversion.

VoIP Voice over IP.

VPC VoicePrivacy Challenge.

VQ Vector Quantization.

WER Word Error Rate.

Baseline systems of the VoicePrivacy Challenge 2022

B1.a x-vector–based with pool of external speakers and separate mel-spectrogram synthesis and vocoder.

B1.b x-vector–based with pool of external speakers and unified vocoder.

B2 Formant shift with McAdams coefficient.

For a summary of the submitted systems, refer to Section 2.3.2 and Table 2.3.

Baseline systems of the VoicePrivacy Challenge 2024

B1 Same as **B1.b** in the VoicePrivacy 2022 Challenge.

B2 Same as **B2** in the VoicePrivacy 2022 Challenge.

B3 Phonetic features are extracted from input utterances by ASR system. Pitch, energy and duration of phonemes are also extracted and perturbed. A GAN generates a new speaker embedding. All obtained representations are used by TTS system to generate new waveform.

B4 Semantic encoder extracts semantic tokens from input utterance. NAC encoder extracts acoustic tokens from pseudo-speaker utterance. Autoregressive Trasformers generate new acoustic tokens from these representation and NAC decoder converts them to waveform.

B5 Wav2vec features are extracted from input waveform and quantized. They are converted to waveform by a vocoder conditioned on the original F0 curve and a one-hot representation of the pseudo-speaker.

B6 Same as **B5**, but uses a TDNN-F instead of Wav2vec.

For a summary of the submitted systems, refer to Section 2.4.4.

Chapter 1

Introduction

1.1 Scope of this dissertation

1.1.1 The value of data, and how to safeguard it

In the age of information technology, data has become, in many ways, a currency and a source of wealth. Algorithms and models are data hungry: recommender systems, home assistants, authentication systems, and more recently Large Language Models (LLMs) require immense amounts of data to function reliably across diverse contexts. Academic research is not exempt, with multiple fields becoming increasingly dominated by deep learning. To possess data is to hold knowledge, to potentially develop superior products as a company, or to be able to propose more disruptive innovation as a researcher. It is therefore unsurprising that, oftentimes, individuals find themselves in the position of trading their own data in exchange for the use of some web service, phone app, or other devices — a transaction that is often handled through a mundane, sometimes inattentive click on the ubiquitous request of "accepting the terms and conditions" [1].

For an active user of internet-based services (as many readers probably are), keeping track of who is storing their data and what it is being used for is nearly impossible. Trying to mitigate that issue by renouncing specific services can result in significant drawbacks: not using a certain messaging app can cut someone off from a social circle [2], while avoiding API-based LLMs can make an employee less productive compared to the competition [3]. To prevent such scenarios, it should be the responsibility of public authorities such as the European Union and national data protection agencies to provide regulations that enforce data privacy, and of the technical community — including academic researchers and industry practitioners — to provide effective tools that enable such regulations to be enforced. Individuals should not have to give up their freedom to protect their privacy: it is the role of a legal and technological framework to safeguard it in a manner that is easy, seamless, and transparent.

To make that possible, the privacy-preservation tools developed by practitioners should function equally seamlessly and transparently. The work presented in this thesis seeks to contribute to that objective. Our work focuses on Voice Anonymization (VA), the process of transforming speech recordings so that the speaker's identity is concealed. Its ultimate goal is to preserve the privacy of the individuals whose voice was recorded (even by themselves): unauthorized parties could use the audio for nefarious purposes, or extract sensitive information about the speaker from it. Once anonymized, a recording cannot be linked back to the original speaker, thereby protecting both their privacy and their voice identity.

1.1.2 The value of *voice* data, and how to safeguard it

Speech technologies and voice data are becoming increasingly important in today's world. In consumer tech, voice-powered smart assistants have found their way into homes, smartphones, and cars [4, 5], sometimes being integrated with LLMs for enhanced capabilities [6].

Speech analysis has found applications in healthcare, particularly in disease monitoring through vocal patterns [7, 8, 9, 10]; voice authentication is increasingly adopted for remote banking authentication [11, 12] and fraud prevention in call centers [13]; voice interfaces also provide essential accessibility for visually impaired users navigating digital applications [14].

Specific applications aside, voice data is, first and foremost, *data*. And, as all other kinds of data, increasingly large amounts of it are generated and stored daily. While precise statistics are difficult to obtain, it is commonly estimated that around 80 – 90% of all digital data is unstructured data, a large part of which is audio and speech data [15]. An immense amount of video content posted on social media contains voice; countless voice messages are sent on messaging apps every day; Voice over IP (VoIP) services, movies, and shows on streaming platforms further contribute to this abundance.

With such a large amount of data and information arise several potential privacy and security threats. Modern Automatic Speaker Verification (ASV) systems are capable of recognizing individuals from even short recordings of their voice [16]. Speech recordings contain sensitive attributes such as age, gender, and emotional state, whose unauthorized inference carries risks of profiling and discrimination [17]. Stored vocal biometric templates can be stolen and due to data breaches and, unlike traditional passwords, cannot simply be "reset". Obtaining samples of an individual's voice allows for acts of impersonation known as *replay attacks*, whereby an attacker replays a stolen voice recording of a victim in the attempt to authenticate or be recognized as them [18].

Last, a particularly concerning and rapidly growing threat is audio deepfakes: synthetic speech generated with deep learning models that mimic the voice of a designated victim [19], sometimes with levels of realism so convincing that make them difficult — if not impossible — to distinguish from genuine speech recordings [20]. This concern is aggravated by the increasing availability of commercial services and even open source models [21, 22] that offer voice cloning to a wide audience, even those without specific, relevant expertise. As previously discussed, these threats call for both *legal* and *technological* mitigation.

One *legal* mitigation is the European Union's General Data Protection Regulation (GDPR), which provides a framework that places requirements on how voice data may be collected, processed, and stored [23, 24]: because speech signals are so information-rich, they may fall under the category of *biometric data* when used for identification, which is considered a special category subject to particularly strict regulations. The recent EU Artificial Intelligence (AI) Act also tackles the issue. It explicitly defines voice-based recognition systems as "biometric identification systems", which are classified as high-risk and prohibited under specific circumstances. Deepfakes are also addressed by the Act, which mandates that synthetic audio content must be clearly disclosed as artificially generated when deployed [25].

The *technological* mitigation can take several forms, including cryptography, Differential Privacy (DP) [26], anti-spoofing [18, 19], and, last but not least, VA. Once anonymized, the voice identity of a speech signal is removed, preventing risks of voice cloning. In addition, anonymized speech cannot be traced back to its original speaker, averting the inference of any sensitive information about them from the recording (e.g., age, gender, health condition).

In this dissertation, we delve further into the topic of VA. Note that, in legal contexts, the term "anonymization" is used only when this objective has been fully achieved. In contrast, we follow the definition of [27], where "anonymization" refers to the task being attempted, even if not successfully.

1.2 Voice anonymization

VA is defined as the task of processing a speech recording to conceal the identity of the speaker while preserving the *utility* of the recording. The notion of "utility" corresponds to the possibility of using the recording in a set of downstream tasks of interest [17]. For example, an individual might want to use a cloud-based speech recognition system without having to share their voice identity with the service provider, which could be made possible by anonymizing their speech. More generally, the usual goal of voice anonymization is to allow the use of voice-based services and the analysis of speech data while protecting the *privacy* of the identity of the involved speakers. This is in line with the principle of *data minimization* as defined in Article 5(1)(c) of the GDPR, which states that disclosure of personal data must be "limited to what is necessary in relation to the purposes for which they are processed" [24].

Meyer et al. proposed a taxonomy of three possible macro-categories of downstream tasks for VA [28]: *human-human* interactions, *human-computer* interactions, and *data storage*. *Human-human* interactions include scenarios in which humans listen to the anonymized voice data, such as call centers, legal recordings, or interviews. *Human-computer* interactions refer to situations in which the speech recordings are processed by machines, such as automatic transcription or interaction with voice assistants. Lastly, *data storage* refers to recordings that are stored for possible future processing, without a predefined downstream task.

We propose one further taxonomic classification that instead regards the privacy adversary. Within the scope of this dissertation, voice anonymization is intended to protect the privacy of the processed speech utterances both in the case of *human intrusion* and *machine-based intrusion*. By "human intrusion" we mean that a malicious privacy adversary attempts to violate the anonymization by simply listening to the anonymized utterance and trying to recognize the original speaker. By "machine-based intrusion" we mean that the privacy adversary makes use of speaker recognition software to retrieve the original speaker identity despite the anonymization. Such software could be specifically tailored to undo the anonymization.

The usual approach to tackle both cases simultaneously is to anonymize the input utterance by changing the voice of the original speaker to that of a different speaker. In the scientific literature, this process is generally referred to as *voice conversion* [29]. A more drastic approach consists in extracting a textual transcription of the utterance, then synthesizing a completely new signal with it in the voice of a different speaker. This practice, known as Automatic Speech Recognition (ASR)+Text-to-Speech (TTS), has its pros and cons in the context of VA, which will be discussed throughout this dissertation.

Note that another line of research instead focuses on protecting speech recordings from voice cloning while leaving them perceptually unchanged to humans [30, 31]. While also providing a form of privacy protection, they are outside the scope of this work, as they do not protect from human intrusion.

A similar argument could be used for the technique of *adversarial attacks*, which can be used to disrupt the functioning of a speaker recognition system by applying a noise perturbation that leaves most of the signal intact [32, 33, 34]. Like VA, they could potentially be used to protect the speech recordings from machine-based intrusion; however, they do not offer protection against human-based intrusion. Moreover, their main goal is usually the disruption of recognition systems, and not the protection of voice identity; as such, they are generally framed as a tool used by a potential adversary for malicious purposes. Nevertheless, they partially intersect with the candidate's work on VA, as will be described in Chapter 5.

1.3 Thesis outline and contributions

We present novel research on the topic of VA. In the following, we provide an outline of the thesis and we summarize the candidate's contributions.

Chapter 2

We describe the task of VA as formalized in the VoicePrivacy Challenge (VPC). We also provide a summary of the VPC editions completed thus far: 2020, 2022, and 2024. During his PhD, the candidate contributed actively to the organization of the VPC and became part of the organizing team. He carried out the post-analysis of the results from the 2022 edition of the challenge, which was published in:

> **Michele Panariello**, Natalia Tomashenko, Xin Wang, Xiaoxiao Miao, Pierre Champion, Hubert Nourtel, Massimiliano Todisco, Nicholas Evans, Emmanuel Vincent, and Junichi Yamagishi. **The VoicePrivacy 2022 Challenge: Progress and Perspectives in Voice Anonymisation**. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:3477–3491, 2024.

Other than the post-analysis, the candidate led the paper's writing, contributing the majority of its content. Figure 2.5 was initially made by one of the co-authors, and later modified by the candidate.

The candidate contributed to the organization of the 2024 edition of the challenge. This involved taking part in the decision process regarding the changes with respect to the 2022 edition, reviewing system descriptions form the participants, and contributing to the official VPC 2024 code base. In particular, the greatest contribution to the code base was the adaptation of the system described in Chapter 4 as a baseline for the challenge. The 2024 challenge description was published in the evaluation plan, which the candidate also co-authored:

> Natalia Tomashenko, Xiaoxiao Miao, Pierre Champion, Sarina Meyer, Xin Wang, Emmanuel Vincent, **Michele Panariello**, Nicholas Evans, Junichi Yamagishi, and Massimiliano Todisco. **The VoicePrivacy 2024 Challenge Evaluation Plan**, 2024.

At the time of writing, the candidate is also involved in the organization of the upcoming 2026 edition of the VPC.

Chapter 3

After a technical introduction to x-vector–based VA, we propose the concept of *vocoder drift*, a measure of how much a vocoder influences an x-vector–based VA pipeline. We find that the vocoder's impact is considerable, and can contribute to the overall privacy protection provided by the system. We then show how an attacker can exploit this behavior to reverse the vocoder drift in what we name a *drift reversal* attack. In the considered VA systems, this kind of attack obtains results on par with current, well-established state-of-the-art attacks. This work was published in:

> **Michele Panariello**, Massimiliano Todisco, and Nicholas Evans. **Vocoder drift in x-vector–based speaker anonymization**. In *Interspeech 2023*, pages 2863–2867, 2023.

Follow-up work investigated the causes of vocoder drift: our results show that it is due to the interaction between the anonymized x-vector and the rest of the features — which are not anonymized — that are input to the vocoder for waveform synthesis. Speaker-specific information is re-entangled in the final signal, causing vocoder drift. We then show how to adjust the input features to mitigate the effect of vocoder drift to allow for more reliable evaluation of the VA system. The results of this work were published in:

> **Michele Panariello**, Massimiliano Todisco, and Nicholas Evans. **Vocoder drift compensation by x-vector alignment in speaker anonymisation**. In *3rd Symposium on Security and Privacy in Speech Communication*, pages 16–20, 2023.

Chapter 4

We propose a novel approach to VA based on the recent technique of Neural Audio Codec (NAC) language modeling. Our model outperforms the best Voice Conversion (VC)-based participant system of the VPC 2022 in terms of privacy protection and voice distinctiveness, albeit at the cost of a marginal reduction in speech content preservation. This work was published in:

> **Michele Panariello**, Francesco Nespoli, Massimiliano Todisco, and Nicholas Evans. **Speaker anonymization using neural audio codec language models.** In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4725–4729, 2024.

The system was later adopted as a baseline for the following 2024 edition of the VPC. In this context, it was named `B4`. Under the new 2024 benchmark, it was the second best baseline in terms of privacy protection, and outperformed other strong baselines in emotion preservation. A minor reduction in content preservation persisted, but was later tackled and solved in follow-up work. This was achieved by integrating into `B4` a novel conditioning mechanism which we named *character-level vocoder conditioning*. With this new setup, the system (later renamed `B4*` in following work) was able to outperform all other baselines in terms of speech content preservation. The work was published in:

> **Michele Panariello**, Massimiliano Todisco, and Nicholas Evans. **Preserving spoken content in voice anonymisation with character-level vocoder conditioning.** In *4th Symposium on Security and Privacy in Speech Communication*, pages 12–16, 2024.

Chapter 5

We shift our investigation to the side of the attacker, examining the reliability of the privacy evaluation. For the first time, we document the issue of privacy overestimation, which takes place when the measured privacy protection of a VA system is due not to the efficacy of the system in erasing personal information, but to a sub-optimal training of the attacker's ASV system. We highlight potential cases of privacy overestimation in the current literature, and directly tackle one of them, correcting the overestimation. We then propose a lightweight technique to detect potential privacy overestimation when evaluating a VA system. Several experiments confirm its reliability. This work was carried out in collaboration with other members of the VPC organizing team, and published in:

> **Michele Panariello***, Sarina Meyer*, Pierre Champion, Xiaoxiao Miao, Massimiliano Todisco, Ngoc Thang Vu, and Nicholas Evans. **The Risks and Detection of Overestimated Privacy Protection in Voice Anonymisation.** In *5th Symposium on Security and Privacy in Speech Communication*, pages 8–12, 2025.

where the symbol * denotes equal first authorship. The candidate's contribution were: the planning of the experiments, all the results relative to `B4` and `B4*`, and most of the paper writing.

Privacy overestimation refers to *inadvertent* sub-optimal training of the attacker's ASV. In subsequent work, we investigated how the defender (that is, the user of the VA system) can *force* the poor performance of any potential attacking ASV system. We do so by applying small adversarial perturbation to some of the data the attacker employs. Experiments performed using one of the VPC 2024 baselines show notable improvements in privacy protection, achieving was is informally known as "perfect privacy" (i.e., the attack is as good as a system trying to guess the speaker identity at random). To the best of our knowledge, this constitutes the first step towards *proactive defense* VA. While current results are encouraging, this investigation is still ongoing and not yet published.

Chapter 6

We summarize the results of our work, and present high-level conclusions that can be drawn from it. We also suggest potential future directions to extend the research described in the dissertation and for the field of VA in general.

Further contributions

The candidate contributed to one more publication on VA concerning the topic of non-timbral cues by suggesting research directions and experiments:

Rayane Bakari, Olivier Le Blouch, Nicholas Evans, Nicolas Gengembre, **Michele Panariello**, and Massimiliano Todisco. **The influence of non-timbral cues in voice anonymisation and evaluation.** In *5th Symposium on Security and Privacy in Speech Communication*, pages 35–42, 2025.

Furthermore, the candidate held a talk for the Security and Privacy in Speech Communication (SPSC) Webinar series[1] entitled "Speaker anonymization: current methods, challenges and perspectives". The talk, held on the 6th of May 2024, illustrated the state of the research on VA, an overview of the VPC 2024, and future research perspectives.

The candidate also carried out further research work outside of the scope of this dissertation. Most notably, he actively investigated the topic of adversarial attacks. He designed *Malafide*, an adversarial attack on speech anti-spoofing countermeasures based on convolutional noise — to the best of our knowledge, the first to adopt this approach. The work was published in:

Michele Panariello, Wanying Ge, Hemlata Tak, Massimiliano Todisco, and Nicholas Evans. **Malafide: a novel adversarial convolutive noise attack against deepfake and spoofing detection systems.** In *Proc. INTERSPEECH 2023*, pages 2868–2872, 2023.

From *Malafide* stemmed a line of research comprising two follow-up works: *Malacopula*, an adaptation of Malafide to target ASV systems; and 2D-Malafide, which tailored the attack to face deepfake detection systems. They were published in:

Massimiliano Todisco, **Michele Panariello**, Xin Wang, Héctor Delgado, Kong Aik Lee, and Nicholas Evans. **Malacopula: adversarial automatic speaker verification attacks using a neural-based generalised Hammerstein model.** In *The Automatic Speaker Verification Spoofing Countermeasures Workshop (ASVspoof 2024)*, pages 94–100, 2024.

Chiara Galdi, **Michele Panariello**, Massimiliano Todisco, and Nicholas Evans. **2d-malafide: Adversarial attacks against face deepfake detection systems.** In *2024 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–7, 2024.

On the topic of adversarial attacks, purification techniques were also touched upon in:

Yibo Bai, Sizhou Chen, **Michele Panariello**, Xiao-Lei Zhang, Massimiliano Todisco, and Nicholas Evans. **MDD: a mask diffusion detector to protect speaker verification systems from adversarial perturbations**. In *2025 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2025.

The topic of gender suppression in speech, both at the embedding level and waveform level, was explored in the following publications:

Oubaïda Chouchane, **Michele Panariello**, Oualid Zari, Ismet Kerenciler, Imen Chihaoui, Massimiliano Todisco, and Melek Önen. **Differentially Private Adversarial Auto-Encoder to Protect Gender in Voice Biometrics.** In *Proceedings of the 2023 ACM Workshop on Information Hiding and Multimedia Security (IH&MMSec '23). Association for Computing Machinery*, pages 127–132, 2023.

[1] https://www.spsc-sig.org/webinar

Oubaïda Chouchane, **Michele Panariello**, Chiara Galdi, Massimiliano Todisco, and Nicholas Evans. **Fairness and Privacy in Voice Biometrics: A Study of Gender Influences Using wav2vec 2.0**. In *2023 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pp. 1-7, 2023.

Yangyang Qu, **Michele Panariello**, Massimiliano Todisco, and Nicholas Evans. **Reference-free Adversarial Sex Obfuscation in Speech**. In *2025 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2025.

Lastly, the candidate contributed to curation of the audio data contained in the following dataset.

Hava Chaptoukaev, Valeriya Strizhkova, **Michele Panariello**, Bianca Dalpaos, Aglind Reka, Valeria Manera, Susanne Thümmler, Esma Ismailova, Nicholas W., Francois Bremond, Massimiliano Todisco, Maria A Zuluaga, and Laura M. Ferrari. **StressID: a multimodal dataset for stress identification.** In *Advances in Neural Information Processing Systems*, volume 36, pages 29798–29811. Curran Associates, Inc., 2023.

Chapter 2

Related works

This chapter provides an overview of the scientific literature on VA. The development of this field has been significantly driven by the VoicePrivacy initiative [17] and its associated VPC series. The VPC is a competition in which participants are invited to design and submit VA systems which are evaluated and ranked under standardized conditions using multiple metrics. At the time of writing, three iterations of the challenge have taken place: in 2020, 2022, and 2024. The following sections focus on the 2022 and 2024 editions, which are most relevant to the work presented in this dissertation.

2.1 Definition of the voice anonymization task

In this section, we describe how the VA task is formalized within the context of the VPC, using a framework that is common to all editions of the challenge held to date. The aspects that distinguish each individual edition are detailed in the subsequent sections.

2.1.1 Definition of a voice anonymization system

A VA system should:

- accept a speech waveform at its input and produce another waveform at its output.
- conceal the identity of the original speaker by altering their voice in the output waveform.
- preserve other speech characteristics such as linguistic content and paralinguistic attributes (e.g., prosody, emotion).

In practice, VA shares many characteristics with VC [29]. However, the two tasks differ in their objectives: VC aims to accurately reproduce the voice of a specific target speaker, whereas the goal of VA is to conceal the identity of the original speaker.

2.1.2 The role of the defender and the attacker

The VA task is formulated as a game between a *defender* and an *attacker*. The **defender** possesses speech data whose privacy has to be protected by means of VA. For example, the user(s) of a voice-powered app sending their own speech recordings over the internet, the guest of a radio program wishing to protect their identity, or the possessor of a speech dataset who intends to share it for research purposes in a privacy-compliant manner. The defender uses a single anonymization system to anonymize and protect the voice data at hand. It is assumed that the data in question contains the recordings of multiple speakers. The **attacker** is a malicious entity that somehow obtains the data the defender is attempting to protect. We assume that the attacker has knowledge of a potential pool of speakers that might appear in the anonymized data. With that knowledge, they attempt to identify the anonymized voices in the protected data by using an ASV system. The process, which we now detail more formally, is illustrated in Figure 2.1.

We denote the defender's data as D. It is composed of a set of waveform utterances $d_1 \dots d_{N_d}$ spoken by a set of speakers $S = \{s_1 \dots s_{N_S}\}$. This data is "unprotected", i.e. it has not yet undergone any anonymization. The defender trains a VA system *VA* and produces the anonymized data $D^{VA} = \left\{d_1^{VA} \dots d_{N_d}^{VA}\right\}$. Each anonymized utterance d_i^{VA} corresponds to original unprotected data d_i processed by anonymization system *VA*. When anonymizing an utterance d, the voice identity of the original speaker is changed to that of a different *pseudo-speaker*. The pseudo-speaker's voice can be synthesized from the recording of a real individual who agreed to donate their voice for this purpose, or can be artificially generated (i.e. the pseudo-speaker is not a real person). Hence, when designing a VA system, it is necessary to define a function which, given an utterance d_i, returns the pseudo-speaker $\hat{s}_i$ whose voice the utterance will be converted to. We refer to this function as a *pseudo-speaker selection strategy*. In general, the literature related to the VPC defines two approaches by which a pseudo-speaker selection strategy can be applied: at the *speaker level* and at the *utterance level* [27, 35]. In speaker-level anonymization, the pseudo-speaker selection function is applied once for a single speaker s_k to select a pseudo-speaker $\hat{s}_k$; all utterances belonging to s_k are then converted to the voice of $\hat{s}_k$. In utterance-level anonymization, the pseudo-speaker selection function is applied to each utterance independently, potentially resulting in different pseudo-speakers even for utterances belonging to the same original speaker.

The attacker's goal is to recover the original speaker identity associated with the utterances in D^{VA}. Importantly, this does NOT imply an attempt to reconstruct the original waveforms D; rather, the attacker seeks to infer the categorical speaker identity — that is, the speaker label. To do so, the attacker makes use of an ASV system. We assume that the attacker has access to a further speech dataset A. It is composed of a set of utterances $a_1 \dots a_{N_A}$ which will be used as *enrollment* utterances to the attacking ASV system. These utterances are assumed to have been spoken by the same speakers in D, or subset of them. Hence, the most notable initial hypothesis is that the attacker knows in advance a list of his potential victims: in that sense, the task can be seen as the attacker "probing" the protected data for certain speakers of interest.

Various kinds of attack have been formalized based on the degree of knowledge that the attacker has of the VA system used by the defender. The simplest case is the one of an **ignorant attacker**, who uses the utterances in A as enrollment data without modification. Such a weak attack was shown to be of little efficacy [36] because of the domain mismatch between D^{VA} (anonymized) and A (unprotected). This issue is circumvented when the attacker is *informed*, i.e. when they know what VA system the defender used to generate D^{VA}. A **lazy-informed attacker** is assumed to have access to that system, and uses it to anonymize A, obtaining A^{VA}. This reduces the domain mismatch between enrollment and trial utterances, and results in a stronger attack. In an even more favorable setting, the attacker might have sufficient data and the computational capability to train a new ASV system that is specifically adapted to the anonymized data and can extract speaker-specific information from it: we refer to this scenario as a **semi-informed** attacker. More specifically, the attacker is assumed to possess a dataset of utterances T whose speakers do not overlap with either A or D. Using the same system as the defender, they anonymize T to T^{VA} and then use the latter to train the ASV model, obtaining ASV^{VA}. A^{VA} and D^{VA} are then used for speaker verification. This attack is the most powerful of the three and, at the time of writing, is the de facto standard to evaluate the privacy protection provided by a VA system [37]. Note that, even in the last case, the attacker is still not completely "informed" because, while they are assumed to know what anonymization system was used to produce D^{VA}, they are still not aware of which pseudo-speaker was chosen for each d_i^{VA}. Hence, T^{VA} is anonymized with a different utterance–to–pseudo-speaker mapping than D^{VA} and A^{VA}. This constitutes a disadvantage for the attacker. Having complete knowledge of the utterance–to–pseudo-speaker mapping would result in a **fully informed attack**; while this scenario has been explored in the literature [38, 39, 40], it is not widely adopted for benchmarking. There are several reasons behind such a choice. The author of [40] argued that a fully-informed attack falls into a *security* problem rather than a privacy problem, and therefore decided not to conduct thorough experimentation with it. The authors of [38, 39] investigated a specific version of the fully informed attack where only a single pseudo-speaker for all utterances in A^{VA}, D^{VA} and T^{VA} was used. Since such a scenario is not widely applicable in practical contexts, it was not considered for further study.

It is worth noting that the focus of all VPC editions was on the defender side: participants were asked to develop their own VA system, while the attacking ASV system was fixed and provided by the organizers. After the 2024 edition of the VPC, the VoicePrivacy Attacker Challenge [41] proposed a more attack-centered competition.

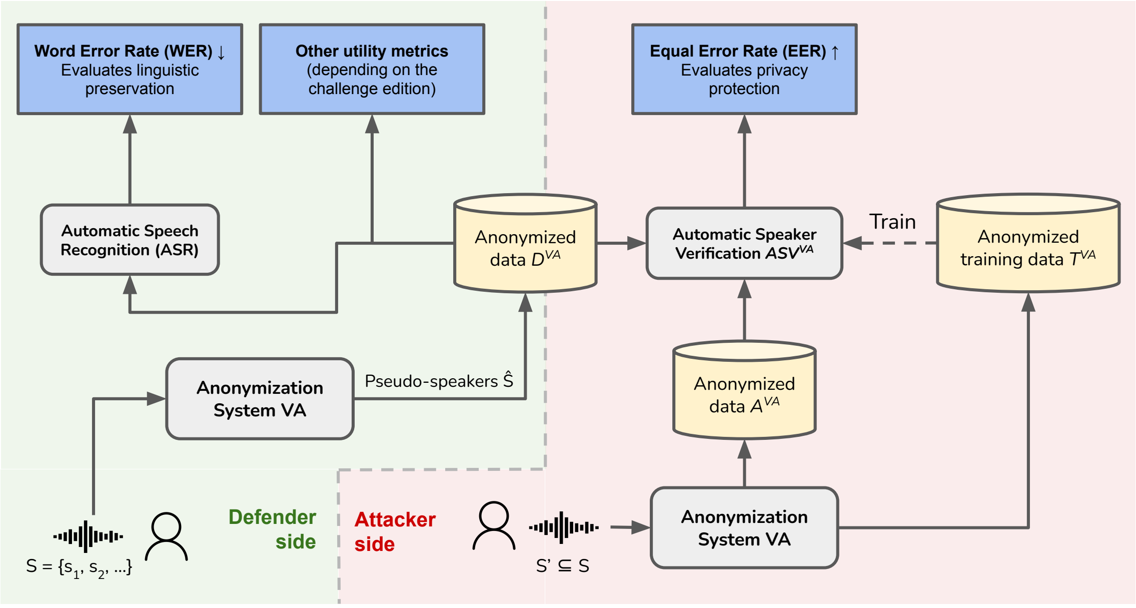


Figure 2.1: General evaluation framework for the VA task in the specific case of the *semi-informed* attack, in which the attacker trains ASV^{VA} on anonymized training data T^{VA}.

2.1.3 Privacy and utility metrics

VA systems aim to protect the privacy of the speaker while preserving the utility of the anonymized signal: hence, the metrics used for evaluation are grouped in *privacy* metrics and *utility* metrics.

Across all editions of the VPC, the primary metric for privacy evaluation has been the Equal Error Rate (EER) achieved by the attacking ASV system [35,37,42]. If the EER is high, it is challenging for the attacker to infer the identity of the original speaker from the anonymized utterances. Hence, a high EER indicates a high level of privacy protection provided by the VA system. A higher EER corresponds to a higher level of privacy: this means that it is challenging for the attacker to infer the identity of the original speaker from the anonymized utterances. This privacy metric is *objective* — i.e., it measures the amount of personally identifiable vocal information that can be retrieved systematically via an algorithm.[1] The 2020 [35] and 2022 [27, 42] editions of the VPC also included *subjective* privacy metrics, whereby human listeners were made to listen to anonymized utterances to assess how similar they sounded to other utterances of the same, original speaker or of a different speaker. Refer to the subsections of the individual VPC editions for more details.

[1] Note that we only refer to *vocal* information: other personally identifiable information might be disclosed by different means, e.g., the speaker verbally stating their name and the like.

A wide variety of utility metrics have been used throughout the various editions of the VPC, reflecting the diversity of potential VA applications. The only utility metric consistently adopted is the Word Error Rate (WER) achieved by an ASR system applied to the anonymized utterances [27, 35, 37]. It is used to provide an estimation of the intelligibility of the signal after it has been anonymized. Like the EER, the WER is an objective metric, since it is based on a well defined algorithm rather than human judgment.

Intuitively, the estimates of privacy and utility are expected to follow an inversely proportional behavior — that is, VA systems that provide high privacy protection will naturally tend to achieve low utility, and vice versa. To understand why, let us consider the trivial example of a VA system that, given any input utterance, always outputs a waveform containing random Gaussian noise. Such a system provides high privacy protection, since it completely destroys any personally identifiable vocal information found in the input utterance; however, it also destroys *any other* information in the utterance, resulting in zero utility. Conversely, a VA system that simply outputs the unchanged input utterance achieves perfect utility (all input information is preserved) but zero privacy (there is no anonymity as the voice identity is also preserved). In general, one of the main goals of the research field of VA is to break free from this *privacy-utility tradeoff*, attempting to design systems that provide high privacy protection while also preserving utility as much as possible, at least for a set of defined downstream tasks.

2.2 VoicePrivacy Challenge 2020

The VoicePrivacy Challenge 2020 [35] was the first of the VPC iterations. In the following, we provide only a high-level description of this challenge edition for introductory purposes, since our work concerns mostly the 2022 and 2024 editions. For the 2020 edition of VPC, all anonymization is performed at the speaker level.

2.2.1 Datasets

The defender is only allowed to use certain datasets to train their VA system. The training datasets include the full VoxCeleb-1,2 dataset [43, 44, 45], and 600h subsets of the training partitions of the LibriSpeech [46] and LibriTTS [47] corpora.

The defender can use a development set to evaluate their VA system. It includes *LibriSpeech dev-clean* [46] and a subset of the VTCK corpus [48], denoted as *VCTK-dev*. Both are split into trial and enrollment subsets to simulate the defender's and attacker's data respectively. In the case of LibriSpeech dev-clean, the speakers in the attacker's set (enrollment) are a subset of the speakers found in the defender's set (trials): in other words, the attacker is attempting to infer the identity of only a subset of the speakers protected by the defender.

Table 2.1: Number of speakers and utterances in the datasets of the VPC 2020. Datasets marked with the symbols ● and ▲ are assumed to be controlled and processed by the defender and the attacker respectively.

Subset			**Female**	**Male**	**Total**	**#Utterances**
VA training	VoxCeleb-1,2 ●		2 912	4 451	7 363	1 281 762
	LibriSpeech train-clean-100 ●		125	126	251	28 539
	LibriSpeech train-other-500 ●		564	602	1 166	148 688
	LibriTTS train-clean-100 ●		123	124	247	33 236
	LibriTTS train-other-500 ●		560	600	1 160	205 044
Dev.	LibriSpeech dev-clean	Enrollment ▲	15	14	29	343
		Trial ●	20	20	40	1 978
	VCTK-dev	Enrollment ▲	15	15	30	600
		Trial (different) ●				10 677
		Trial (common) ●				695
Eval.	LibriSpeech test-clean	Enrollment ▲	16	13	29	438
		Trial ●	20	20	40	1 496
	VCTK-test	Enrollment ▲	15	15	30	600
		Trial (different) ●				10 748
		Trial (common) ●				700
ASV training	LibriSpeech train-clean-360 ▲		439	482	921	104 014

In the case of VCTK, attacker and defender utterances contain the same speakers. VCTK-dev is split into two groups: *common* and *different*. For the former, the speakers pronounce a shared set of utterances, so that the same spoken content is available for multiple speakers. The latter comprises different utterances for all speakers. The evaluation set is constructed similarly to the development set, comprising *LibriSpeech test-clean* and a subset of VCTK denoted as *VCTK-test*, in which the utterances are also divided between *common* and *different*.

A summary of the datasets, as well as the number of speakers and utterances of each, can be found in Table 2.1. The number of same-speaker and different-speaker trials in the development and evaluation datasets is given in Table 2.2.

2.2.2 Evaluation

Privacy evaluation is performed with ignorant, lazy-informed, and semi-informed attacks. The ASV system is based on x-vector embeddings paired with a Probabilistic Linear Discriminant Analysis (PLDA) backend [49, 50] that produces Log-Likelihood Ratio (LLR) scores between enrollment and trial utterances. The x-vector extractor is trained on *LibriSpeech train-clean-360* [46]. As detailed in Section 2.1.2, the dataset is anonymized before training in the case of the semi-informed attack. The LLR scores are used to compute the final EER.

Table 2.2: Number of speaker verification trials in the 2020 VPC.

Subset		Trials	Female	Male	Total
Dev.	LibriSpeech dev-clean	Same-speaker	704	644	1 348
		Different-speaker	14 566	12 796	27 362
	VCTK-dev	Same-speaker	2 125	2 366	4 491
		Different-speaker	18 029	17 896	35 925
Eval.	LibriSpeech test-clean	Same-speaker	548	449	997
		Different-speaker	11 196	9 457	20 653
	VCTK-test	Same-speaker	2 290	2 096	4 386
		Different-speaker	17 894	18 210	36 104

The primary utility evaluation metric is the WER computed with a Factorized Time-Delay Neural Network (TDNN-F) [51] and a trigram language model trained on *LibriSpeech train-clean-360*. The WER is computed separately on the evaluation partitions of both LibriSpeech and VCTK (in the case of LibriSpeech, attacker and defender data are merged together).

A pair of secondary utility metrics are also used: De-Identification (DeID) and Gain of Voice Distinctiveness (G_{VD}), both based on speaker similarity matrices [52]. DeID quantifies the overall loss in similarity between utterances of the same speaker following anonymization, whereas G_{VD} assesses the ability of the VA system to generate distinct-sounding voices. Among these two metrics, only G_{VD} was retained for the 2022 challenge edition; its detailed description is provided in Section 2.3.1.

The evaluation protocol also includes four subjective metrics: two that assess privacy (*speaker verifiability* and *speaker linkability*) and two for utility (*speech naturalness* and *speech intelligibility*). As three of these metrics were also employed in the 2022 edition, their descriptions are likewise deferred to Section 2.3.1.

2.3 VoicePrivacy Challenge 2022

The second edition of the VPC, held in 2022, introduced new metrics and updated evaluation protocols. We present the challenge rules, a summary of the submitted VA systems, and an analysis of the results [42].

We omit the description of the training and evaluation datasets, as they are the same as those of the 2020 edition (see Section 2.2).

2.3.1 Metrics

Evaluation metrics are categorized as primary and secondary. Only primary metrics were used for system ranking.

Primary objective assessment

Privacy metric – EER. Anonymization performance is evaluated objectively using an ASV system based on x-vector speaker embeddings and PLDA scoring [49], as in the VPC 2020. In that edition, semi-informed attacks were found to consistently outperform both ignorant and lazy-informed attacks [35]. Consequently, only semi-informed attacks were considered in the 2022 edition, in order to ensure a strong, “worst-case-scenario” evaluation setting.

A further change introduced in the 2022 edition concerns the anonymization of the attacker’s training data T^{VA}. While this was performed at the speaker level in 2020 (see Section 2.2), it was instead carried out at the utterance level in 2022. This modification was motivated by findings indicating that training ASV^{VA} on utterance-level anonymized data leads to a more effective attack [53, 54]. In addition, results were reported for the *unprotected* scenario, where no anonymization is applied by either the defender or the attacker. This corresponds to a standard speaker verification setup and serves solely as a baseline reference for the EER.

For a given speaker and for both evaluation scenarios, all enrollment utterances available to the attacker are used to compute an average enrollment x-vector. The number of same-speaker and different-speaker trials in the development and evaluation datasets is given in Table 2.2. The final privacy level estimation is given by the ASV^{VA} EER: the higher, the better the privacy preservation.

Utility metric – WER. The preservation of linguistic information is assessed objectively using an ASR system based on the Kaldi toolkit [55]. The ASR model architecture is once again the same as for the 2020 edition: a TDNN-F with a trigram acoustic model. As for the approach to gauge privacy, we consider two ASR evaluation scenarios: *unprotected* and *anonymized.* In the unprotected scenario, no anonymization is used. Unprotected defender data (i.e., trial utterances) is decoded using the ASR model trained using the original *LibriSpeech-train-clean-360* dataset. In the anonymized scenario, anonymized defender data is decoded using an ASR model trained using the *LibriSpeech-train-clean-360* dataset after it is treated with *utterance-level* anonymization, using the same anonymization system as the defender data. The system is denoted ASR^{VA}. The first scenario again serves as a baseline. The lower the WER for the second, the better the utility preservation.

Privacy-utility tradeoff. New to the 2022 edition of the VPC is the use of multiple evaluation conditions. These are introduced in recognition of the practical demand for different privacy-utility trade-offs and solutions which can be configured to operate at different operating points, as well as to provide *common* optimization criteria; for the 2020 challenge, participants had to select an appropriate privacy-utility trade-off themselves, resulting in each team essentially choosing *different* optimization criteria. Evaluation conditions take the form of increasingly demanding minimum privacy requirements. For each condition, systems which meet the corresponding minimum privacy condition are then ranked according to utility preservation. The primary privacy and utility metrics (EER and WER) are used for this purpose.

To stimulate progress, the 4 evaluation conditions are specified by a range of modest-to-ambitious minimum target EERs: 15%, 20%, 25% and 30%. Participants were encouraged to submit solutions to as many conditions as possible, with submissions to any one condition being required to achieve a weighted average EER for the evaluation set greater than the minimum target. In the final evaluation, EERs and WERs are equally-weighted averages computed from those for the *LibriSpeech-test-clean* and *VCTK-test* datasets (with the *different* and *common* partitions having the same weight within VCTK).

Secondary objective assessment

Also new to the VPC 2022 was the introduction of a pair of secondary utility metrics, namely estimates of the gain of voice distinctiveness G_{VD} and the pitch correlation $\boldsymbol{\rho}^{F_0}$.

Voice distinctiveness – G_{VD} As in the previous challenge edition, the gain of voice distinctiveness was adopted to help observe the consistency in pseudo-voices for speaker-level anonymization. G_{VD} is estimated using voice similarity matrices [52, 56]. A voice similarity matrix $M = (M(i,j))_{1\leq i\leq N, 1\leq j\leq N}$ is defined for a set of N speakers. $M(i,j)$ reflects the similarity between the voices of speakers i and j:

$$M(i,j) = \mathrm{sigmoid}\left(\frac{1}{n_i n_j} \sum_{\substack{1\leq k\leq n_i \text{ and } 1\leq l\leq n_j \\ k\neq l \text{ if } i=j}} \mathrm{LLR}\left(x_k^{(i)}, x_l^{(j)}\right)\right) \tag{2.1}$$

where $\mathrm{LLR}\left(x_k^{(i)}, x_l^{(j)}\right)$ is the log-likelihood-ratio obtained by comparing the k-th utterance from the i-th speaker with the l-th utterance from the j-th speaker, and where n_i and n_j are the numbers of utterance for each speaker. LLRs are estimated using the ASV model trained using unprotected data. Two matrices are computed: M_{oo}, computed using unprotected utterances; M_{aa}, computed using anonymised utterances. The diagonal dominance $D_{\mathrm{diag}}(M)$ is then computed for both. $D_{\mathrm{diag}}(M)$ is the absolute difference between the mean values of diagonal and off-diagonal elements:

$$D_{\mathrm{diag}}(M) = \left| \sum_{1\leq i\leq N} \frac{M(i,i)}{N} - \sum_{\substack{1\leq j\leq N \text{ and } 1\leq k\leq N \\ j\neq k}} \frac{M(j,k)}{N(N-1)} \right|. \tag{2.2}$$

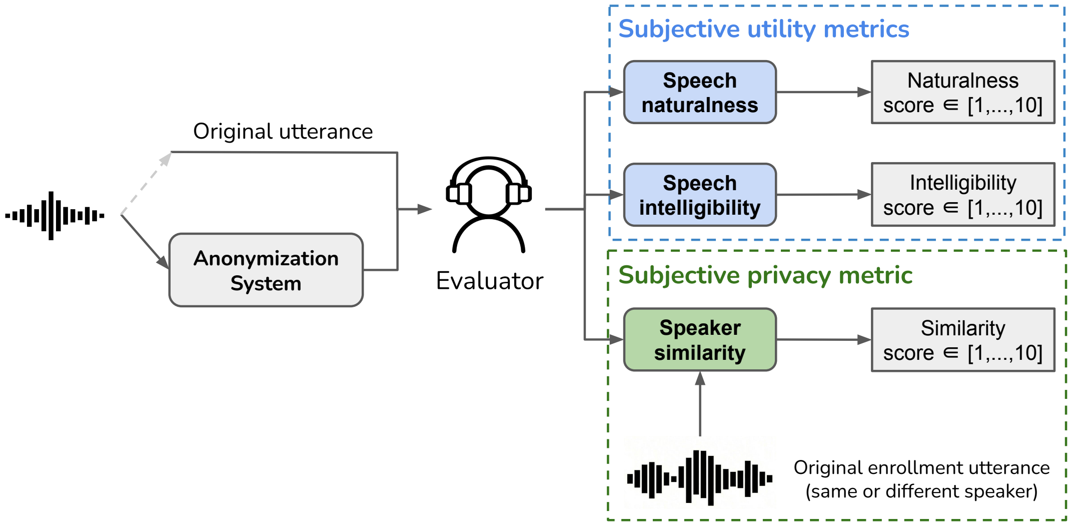

Subjective utility metrics
Speech naturalness
Naturalness score ∈ [1,...,10]
Speech intelligibility
Intelligibility score ∈ [1,...,10]
Original utterance
Anonymization System
Evaluator
Subjective privacy metric
Speaker similarity
Similarity score ∈ [1,...,10]
Original enrollment utterance (same or different speaker)

Figure 2.2: Subjective assessment pipeline for speech naturalness, intelligibility, and speaker verifiability in the 2022 VPC.

G_{VD} is then computed as the ratio of diagonal dominance for each of the two matrices [52]:

$$G_{\text{VD}} = 10 \log_{10} \frac{D_{\text{diag}}(M_{\text{aa}})}{D_{\text{diag}}(M_{\text{oo}})}. \tag{2.3}$$

A gain of $G_{\text{VD}} = 0$ implies the preservation of voice distinctiveness. Positive and negative gains correspond respectively to an average increase or decrease in voice distinctiveness.

Pitch correlation – $\boldsymbol{\rho}^{F_0}$ Estimates of pitch correlation are used to approximate the degree to which an anonymization system preserves intonation. Following [57], the pitch correlation metric $\boldsymbol{\rho}^{F_0}$ is the Pearson correlation between the pitch contours of original and anonymized utterances. The shortest of the two sequences is linearly interpolated so that its length matches that of the longest sequence. The temporal lag between original and anonymized utterances is then adjusted in order to maximize the Pearson cross-correlation when estimated using only segments during which both original and anonymized utterances are voiced. Estimates of $\boldsymbol{\rho}^{F_0}$ are averaged across the full set of utterances in a given data set. While a secondary metric, all submissions were required to achieve an average pitch correlation of $\boldsymbol{\rho}^{F_0} > 0.3$ for each dataset and for each evaluation condition to which a submission was made.

Subjective assessment

As illustrated in Figure 2.2, subjective metrics include estimates of speaker verifiability, speech intelligibility and speech naturalness. Subjective evaluation tests were conducted by the challenge organizers. For naturalness and intelligibility assessments, evaluators were asked to rate a *single* original or anonymized trial utterance at a time. For naturalness, evaluators assigned a

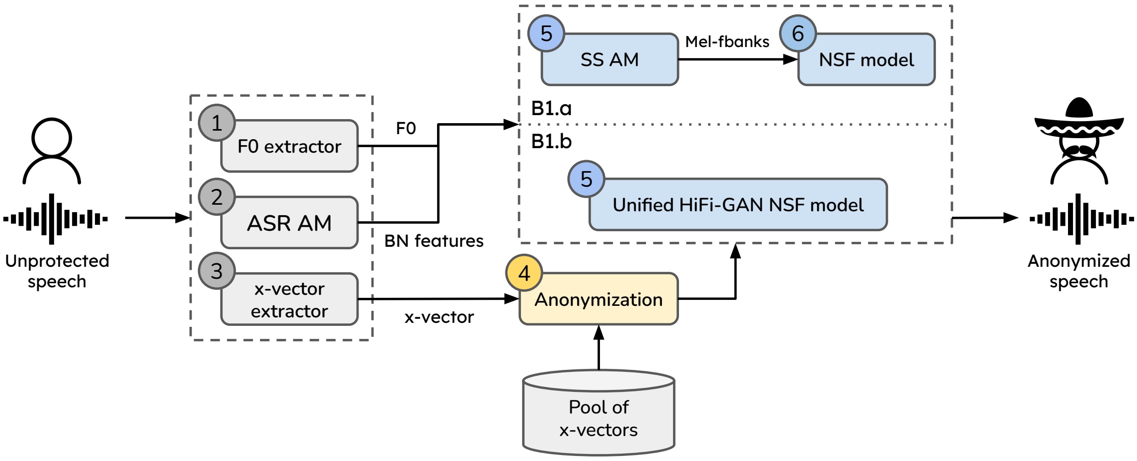


Figure 2.3: Baseline anonymization systems B1.a and B1.b.

score from 1 ('totally unnatural') to 10 ('totally natural'). For intelligibility assessments, evaluators assigned a score from 1 ('totally unintelligible') to 10 ('totally intelligible'). Both naturalness and intelligibility scores were normalised to within a range between 0 and 1 using rank normalization [58], with 0 representing the value of lowest naturalness/intelligibility, and 1 representing the highest. Assessments of speaker verifiability were performed with *pairs* of utterances, namely an unprotected enrollment utterance, and either an unprotected or anonymized trial utterance collected from the same or a different speaker. Evaluators assigned a speaker similarity score between 1 ('the trial and enrollment speakers are surely different') and 10 ('the trial and enrollment speaker are surely the same'). Similarity scores were normalized in the same way as the naturalness and intelligibility scores.

The evaluation trials are taken from the *LibriSpeech-test-clean* dataset and include 1 352 unprotected utterances and 104 anonymized utterances per anonymization system. Each subset of anonymized utterances contains 1 target trial and 1 non-target trial for each of 52 different speakers, evenly split between female and male. The evaluation is performed by 52 native English speakers aged 18 to 70, of whom 40 where male, 11 were female, and 1 of undisclosed gender. Each evaluator rated 52 trials, 26 of which were unprotected, with the remaining utterance being anonymized either by a baseline or by one of the submitted systems. With this configuration, all trial-enrollment pairs used for subjective evaluation were rated by at least one evaluator.

2.3.2 Anonymization systems

We provide a description of the three VPC 2022 baseline systems along with those developed by challenge participants.

Three different anonymization systems were provided as challenge baselines, denoted `B1.a`, `B1.b`, and `B2`. Baselines `B1.a` and `B1.b` are shown in Figure 2.3. Inspired by [59], `B1.a` uses x-vectors and neural waveform models, and comprises three steps: first, x-vector [49], pitch (F0) and Bottleneck (BN) features [51] which encode linguistic content are extracted from the input utterance (blocks ①, ②, ③); second, the x-vector is anonymized (block ④); third, speech is synthesized using the anonymized x-vector and the original F0 and BN features (blocks ⑤ and ⑥). Pitch is estimated using YAAPT [60]. BN features are 256-dimensional vectors extracted using a TDNN-F ASR Acoustic Model (AM) [51]. Speaker embeddings are 512-dimensional x-vectors extracted using a Time-Delay Neural Network (TDNN) [49]. The *anonymization function* (yellow block in Figure 2.3) converts the original x-vector to an anonymized substitute. Anonymized x-vectors are generated by averaging a set of N^* x-vectors. The latter are selected from a larger set of the N farthest x-vectors from the original x-vector selected at random using a PLDA [50] distance metric. A Speech Synthesis (SS) AM generates Mel-filterbank features from the anonymized x-vector and F0+BN features. The speech synthesis module is a Neural Source-Filter (NSF) waveform model [61]. Full details are available in [62]. Baseline `B1.b` is the same as `B1.a`, except that the SS AM is removed and the NSF waveform model is fed directly with BN features. Moreover, the NSF is trained with an additional discriminator loss inspired by the HiFi-GAN system [63].

Baseline `B2` is the technique presented in [64], and is based purely on signal processing techniques. The method utilizes a coefficient α which is referred to as the *McAdams* coefficient. Each pseudo-speaker is associated to a value of α randomly sampled from a uniform distribution within the range $(\alpha_{\min}, \alpha_{\max})$. Linear predictive coding (LPC) is used to decompose the input utterance into a set of pole positions and an excitation signal. The poles positions are rotated within the complex plane to adjust the phase ϕ to ϕ^{α}, hence shifting the formant positions of the input signal. An anonymized utterance is then synthesized using the modified pole positions and the original excitation signal.

The challenge saw five teams submitting their anonymization system. Systems by teams `T04`, `T11`, `T18` and `T40` were based on deep learning models. More specifically, `T04` [65] proposed a system based on the combination of an ASR [66] and a TTS [67] model: a phoneme-level transcription is inferred from the input, then converted into articulatory feature vectors [68] which are used along with a GAN-generated speaker embedding to generate a new waveform with the TTS engine. Remaining systems were variations of `B1.a` and `B1.b`. `T11` replaced the speaker embeddings with one-hot representations, F0 curves with Yingrams [69], and the ASR AM with U2 [70]. `T18` used adversarial noise to anonymize the speaker embeddings. `T40` estimates the F0 curve from the BN features and the speaker embedding instead of extracting it from the original utterance. Team `T32` proposed a purely signal processing-based anonymization system which

Table 2.3: Summary of the anonymization systems submitted to the the 2022 VPC.

Team	Feature extraction	Anonymization	Resynthesis	Results summary
T04	x-vectors and ECAPA [73] vectors are concatenated to create speaker embeddings. Linguistic content is transcribed to phonemes. Removed F0 extraction.	Pseudo-speaker embeddings generated with GAN [74].	TTS model generates Mel-spectrograms that are converted to waveform by HiFi-GAN [63].	TTS-based approach provides excellent levels of privacy and utility, but barely passes the $\boldsymbol{\rho}^{F_0} > 0.3$ requirement.
T11	Yingram [69] for F0 extraction, U2++ [70] for linguistic features. Speaker embeddings are one-hot representations of the speaker IDs encountered during training. One "pseudo-speaker ID" not corresponding to any real speaker is also stored.	Final pseudo-speaker embedding created by means of a weighted average between K random speaker embeddings and the one-hot representation of the "pseudo-speaker ID".	HiFi-GAN [63] is used to synthesize waveforms from AM-generated Mel-spectrograms.	T11-p4 has one of the best results in terms of privacy and utility, but very low G_{VD}: all pseudo-speakers are "similar".
T18	As for B1.a and B1.b, except for their second system where x-vectors are replaced with Transformer-based ASR embeddings.	Two anonymization strategies are proposed. The first uses adversarial noise to anonymize the speaker embedding. The second replaces the x-vector embedding with an ASR-based embedding.	As in B1.a.	Both approaches offer modest privacy improvement over B1.a and B1.b at the cost of reduced WER.
T40	F0 curve not extracted from the intput signal directly, but estimated from x-vector and BN features.	As in B1.a and B1.b.	As in B1.b.	Modest improvement over B1 in terms of privacy.
T32	Signal processing-based approach: pitch shift with TD-PSOLA [71] and PV-TSM [72].			Performance mostly on par with B2 except for a higher $\boldsymbol{\rho}^{F_0}$ and subjective intelligibility.

employs the pitch-shifting techniques TD-PSOLA [71] and PV-TSM [72] to anonymize the input utterance. A summary of the submitted systems can be found in Table 2.3. For further details, refer to [42].

2.3.3 Results

Objective evaluation results

Primary objective assessment results for baselines and all submitted systems for the evaluation set are illustrated in Figure 2.4. The plot to the left depicts the privacy-utility trade-off for unprotected data (gray points), for each baseline system (black points) and for each submission

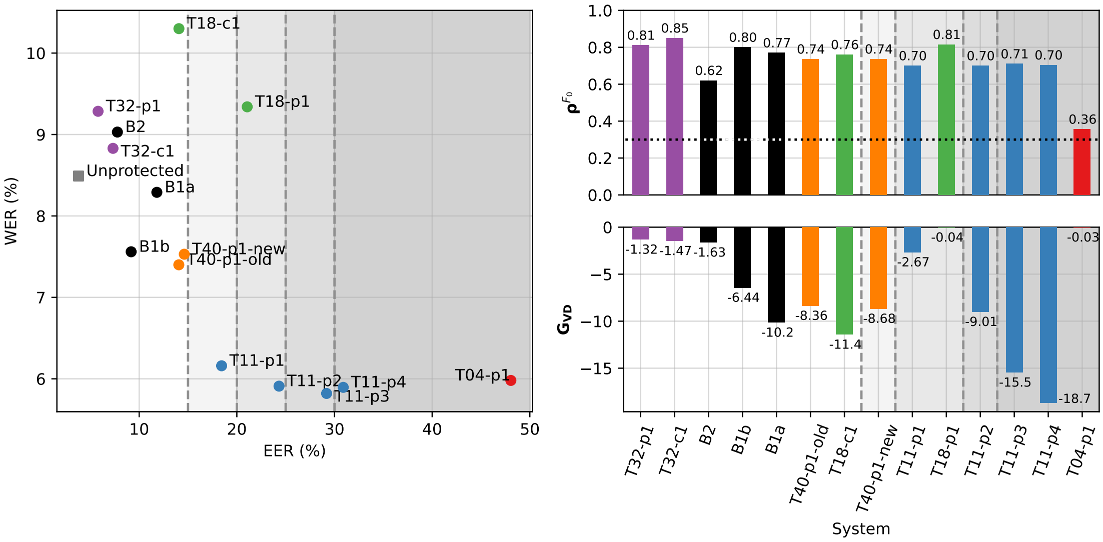


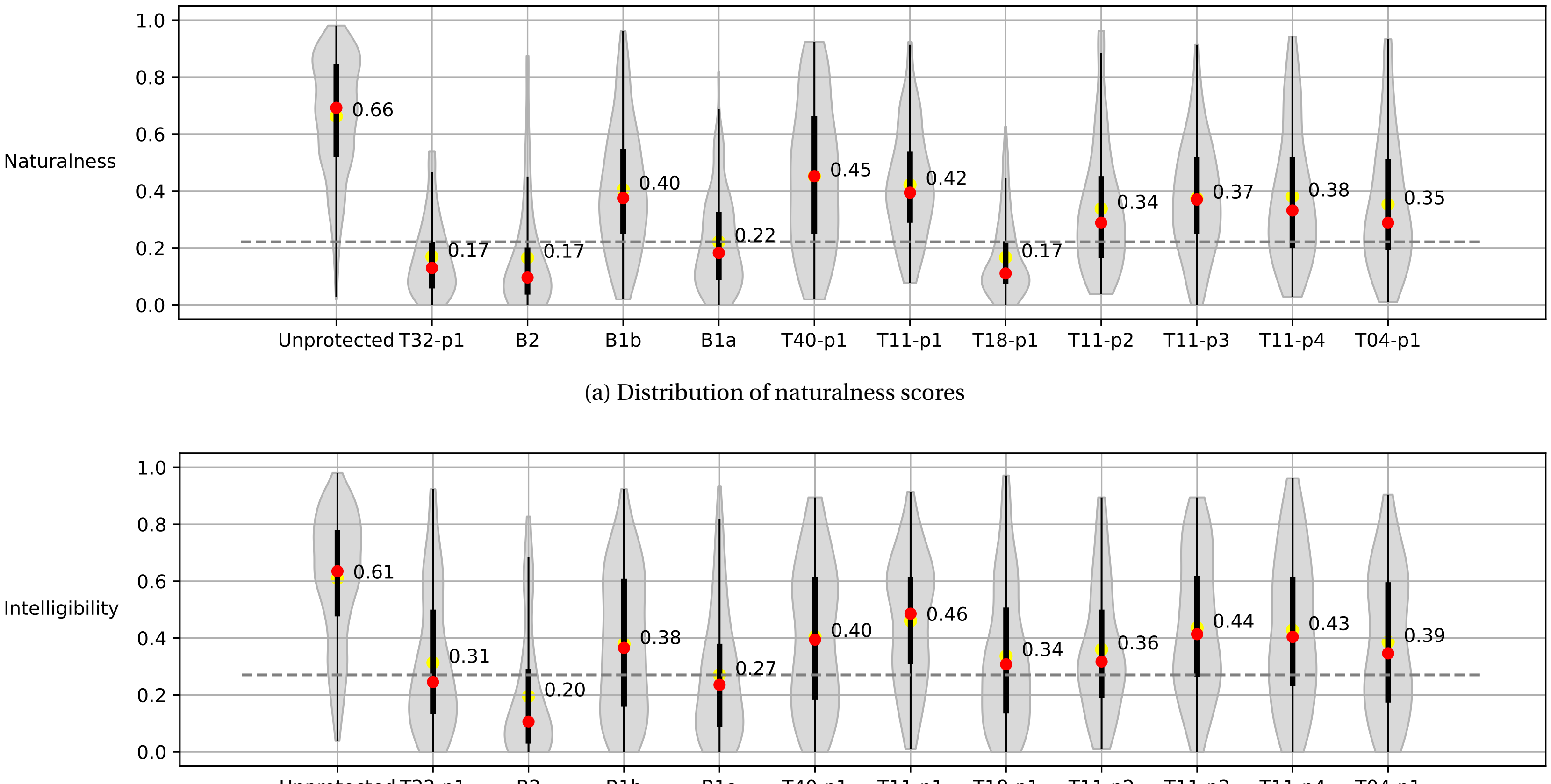
Figure 2.4: Objective evaluation results for the 2022 VPC test set. Unprotected data was evaluated with ASV and ASR systems trained on unprotected data, while anonymized data was evaluated with ASV^{VA} and ASR^{VA}. Vertical dashed lines indicate the separation between different evaluation conditions. The horizontal dotted line in the pitch correlation plot shows the minimum pitch correlation threshold $\boldsymbol{\rho}^{F_0} = 0.3$.

(colored points). All systems submitted by a given team are depicted by points of the same color. The vertical dashed bars depict the set of minimum target EERs of the four evaluation conditions defined in Section 2.3.1.

To the right of Figure 2.4 are results for secondary metrics $\boldsymbol{\rho}^{F_0}$ and G_{VD}. The set of systems is sorted according to the EER (low to high), and each bar has the same colour as corresponding points in the privacy-utility plots. The dashed bars again depict the juncture between evaluation conditions. The horizontal dotted line in the upper plot of pitch correlation results indicates the minimum threshold of $\boldsymbol{\rho}^{F_0} = 0.3$.

Signal processing approaches (B2, T32) are weakest in terms of both primary metrics ($\approx$ 8% EER, $\approx$ 9% WER), but achieve good performance in terms of $\boldsymbol{\rho}^{F_0}$ (0.8) and G_{VD} -1.3. T11 provided a system that offers good privacy level (30% EER) at the cost of a diminished G_{VD} (-18.7). Perhaps the most notable result is T04, which achieved nearly 50% EER, but barely satisfies the requirement of having $\boldsymbol{\rho}^{F_0} > 0.3$.

It is worth noting that, for all evaluated systems, the EER computed on the test partition of VCTK was always higher than for LibriSpeech. This could reasonably be due to the fact that the attacker trains ASV^{VA} on a training partition of LibriSpeech: the ASV task is therefore in-domain in the case of LibriSpeech-test, and out-of-domain in the case of VCTK-test, resulting in a stronger attack case of the former.



(a) Distribution of naturalness scores

(b) Distribution of intelligibility scores

Figure 2.5: Subjective assessment results of baselines and submitted systems in terms of (a) naturalness and (b) intelligibility. Red and yellow dots represent respectively the median and the mean of each set of scores. The dashed gray line corresponds to the mean score of baseline B1.a, provided to facilitate the comparison with the other systems. Figure reproduced from [42].

Subjective evaluation results

Results of subjective naturalness assessment are shown in Figure 2.5a. The first observation is a universal and substantial degradation in naturalness stemming from anonymization. Naturalness scores for the B2 baseline and T32-p1 systems, both signal-processing based, are among the lowest. The relatively higher scores for Team T11 systems and lower scores for B1.a and derived T18-p1 systems suggest that adversarially trained vocoders (HiFi-GAN, NSF) produce more natural speech. The T04-p1 system, albeit ASR+TTS-like, is also competitive.

The trends for intelligibility scores shown in Figure 2.5b reflect those for naturalness; anonymization also universally degrades intelligibility. The B1.b baseline, the T40-p1 and the set of T11 systems are all competitive, as is the ASR+TTS-like T04-p1 system. Scores for signal processing based solutions and the B1.a baseline are the lowest though, in contrast to results for objective utility assessment, the T18-p1 system fares better. We address the correlation between objective and subjective results later in Section 2.3.4.

2.3.4 Post-evaluation analysis

Further examination of the challenge results, later included in the official post-evaluation analysis of the VPC 2022 [42], were performed by the candidate as an organizing member of the VPC team. They are reported in this section.

EER of verifiability according to human listeners

Verifiability score histograms shown in Figure 2.6 depict the distribution of verifiability scores for target trials (the speaker of both enrollment and trial utterances is the same) and non-target trials (the speaker of both enrollment and trial utterances is different). The top-left-most histogram shows distributions for unprotected enrollment and trial utterances (no anonymization) and that listeners can determine with reasonable reliability when the speaker of each utterance is the same or different. All other histograms depict distributions when trial utterances are anonymized with one of 11 anonymization baselines and submitted systems. Enrollment utterances remain unprotected. The number of histogram bins is reduced compared to the plot for unprotected data because of the smaller number of trials. In almost all cases, the distributions for target and non-target trials are highly overlapping, indicating that listeners have greater difficulty to determine when the speaker of each utterance is the same or different. This includes distributions for the B2 and T32-p1 systems indicating that signal processing and deep learning based approaches to anonymization are equally effective in the case of human listeners. EERs estimated from the scores provided by human evaluators are almost all above 40%. These estimates are, however, particularly noisy given the low number of trials, hence the reporting of histograms.

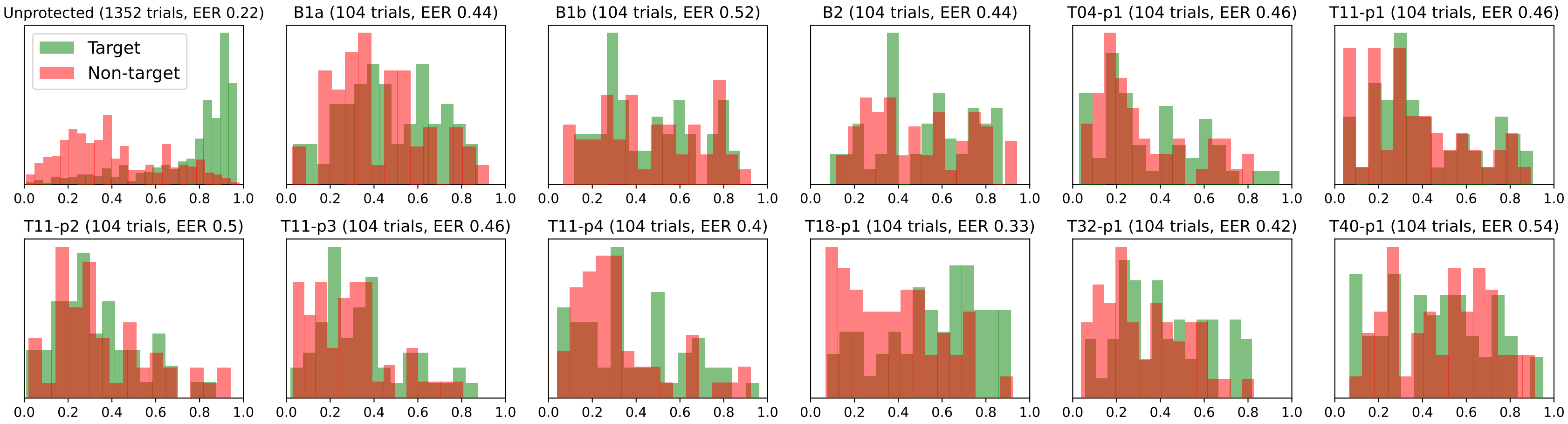


Figure 2.6: Distribution of target and non-target scores according to human listeners. For the top-left histogram, all data is unprotected. For all others, the title above each histogram identifies the system used to anonymize trial utterances. Enrollment utterances remain unprotected. Plots reproduced from [42].

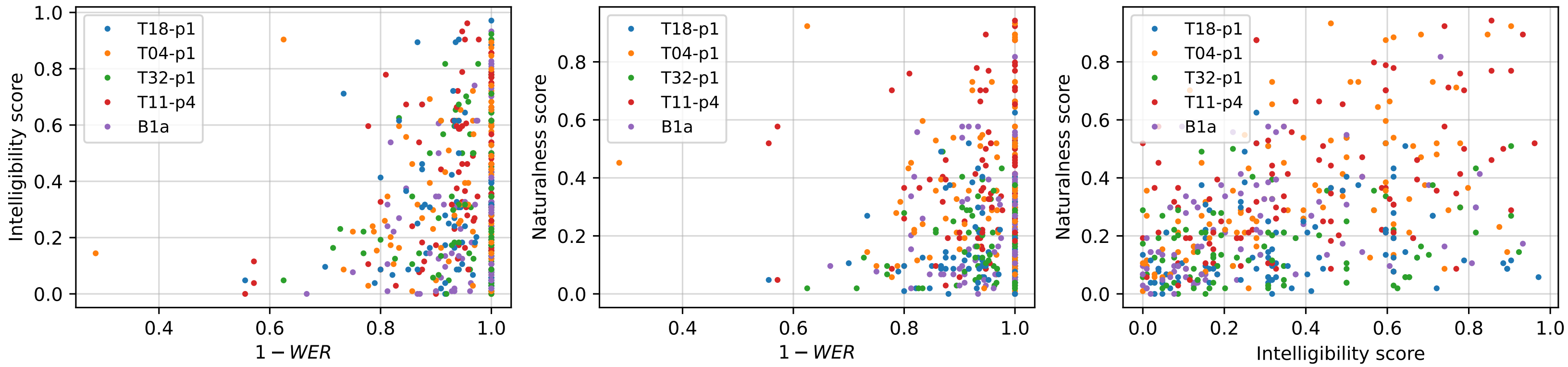


Figure 2.7: Scatterplots showing intelligibility score, naturalness score and the score corresponding to $1 - \text{WER}$ for utterances of different systems.

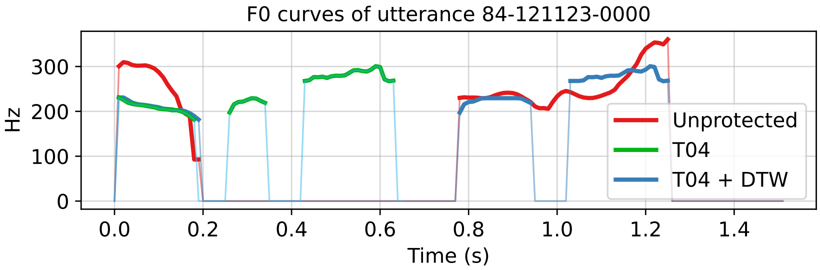


Figure 2.8: F0 contours for utterance 84-121123-0000 (from the *LibriSpeech-test-clean* set). Red – unprotected; green – anonymized with T04 system; blue – anonymized with T04 system and then aligned to the original utterance using DTW.

Subjective versus objective estimates of intelligibility

The results in Figure 2.4 show that WER estimates for some submissions are even lower than for unprotected data, implying that anonymization actually *improves objective estimates* of intelligibility (see the following sections for a specific discussion of this issue). However, results in Figure 2.5 show that anonymization *degrades subjective estimates* of intelligibility. It is hence of interest to explore these contradicting observations further.

Figure 2.7 shows a set of scatter plots which depict the correlation between *utterance-level* subjective intelligibility scores (left) and subjective naturalness scores (middle) against 1 – WER, for four different submissions and the baseline B1.a system. In both cases, and for all five anonymization systems, the correlation is low; the Pearson correlation with 1 – WER is 0.14 for intelligibility and 0.05 for naturalness. Curiously, for some utterances, 1 – WER = 1 (they are perfectly transcribed) but the corresponding subjective scores are near zero. These observations suggest that objective and subjective measures are *not functionally equivalent*, even though the WER is defined in [27] as a proxy for intelligibility. Based on this observation, it is clear that objective measures should not be considered as a proxy for subjective measures. Even so, both are still of interest; they are indicators of anonymization performance for different use cases, one involving the automatic treatment of anonymized utterances by machines and, for the other, consumption by human listeners.

As shown in the scatterplot to the right of Figure 2.7, there appears nevertheless to be some degree of correlation between the two subjective measures, with their Pearson correlation coefficient being 0.58. This might indicate that, from a perceptual perspective, the concept of ‘naturalness’ is intrinsically linked to intelligibility. In the future, subjective metrics which better distinguish between the two aspects should be considered, e.g. assessments of intelligibility might be made by asking listeners to transcribe the spoken content of both anonymized and unprocessed utterances, and comparing the results.

Table 2.4: Pitch correlation values of different anonymization systems with and without the application of Dynamic Time Warping (DTW).

	$\boldsymbol{\rho}^{F_0}$ (normal)	$\boldsymbol{\rho}^{F_0}$ (with DTW)
T04-p1	0.36	0.83
T11-p4	0.70	0.91
T18-p1	0.70	0.93
T32-p1	0.81	0.94
T40-p1 (old)	0.74	0.90
T40-p1 (new)	0.74	0.90

ASR+TTS-based anonymization

Since its inception, VoicePrivacy was designed to encourage the development of anonymization systems which preserve linguistic and para-linguistic speech attributes. Despite it being a design goal, a means to gauge the preservation of para-linguistic attributes was missing in 2020. This was an obvious weakness since ASR systems could be used to generate an intermediate transcription of the input before the application of TTS to generate perfectly voice-anonymized and intelligible utterances, albeit without the para-linguistic attributes of the input. While the submission of ASR+TTS systems was not prohibited, the pitch correlation metric $\boldsymbol{\rho}^{F_0}$ and minimum threshold were hence introduced for the VPC 2022 edition in order to favor solutions (e.g. those based on voice conversion) which offer better potential for anonymization while also preserving para-linguistic attributes.

The majority of teams pursued voice conversion-based solutions. Team T04 was alone in exploring an ASR+TTS-like solution which, perhaps unsurprisingly, delivers near-to-perfect objective anonymization results and among the lowest WERs for test data (see Figure 2.4), as well as competitive subjective naturalness and intelligibility assessment results (see Figure 2.5). The gain in voice distinctiveness G_{VD} is the best of all and, interestingly, the pitch correlation $\boldsymbol{\rho}^{F_0}$ also exceeds the minimum threshold. Whether ASR+TTS solutions are appropriate, whether the minimum pitch correlation threshold for the VPC 2022 was perhaps too low, or even whether pitch correlation is sufficient on its own as a measure of para-linguistic attribute preservation, are all matters of opinion. Ultimately, all are also dependent upon the specific use case scenario. Given the promising EER, WER and G_{VD}, ASR+TTS systems warrant continued attention, especially given that various techniques could be applied readily to boost the pitch correlation or the preservation of para-linguistic attributes in order to improve their competitiveness with voice conversion-based solutions. In the following, we present some initial work that was carried out designed to gauge the potential.

Pitch realignment

We applied a trivial form of pitch realignment using DTW to determine whether the pitch correlation for ASR+TTS anonymized utterances can be improved, and hence whether even higher minimum thresholds might also be met with ASR+TTS approaches. Whereas the pitch correlation $\boldsymbol{\rho}^{F_0}$ is estimated by compensating for any misalignment between anonymized and unprotected utterances using linear warping functions, DTW supports more flexible non-linear warping functions. This greater flexibility should result in improved pitch correlation. We set the DTW local continuity constraints to warp the pitch correlation of anonymized utterances onto those of corresponding unprotected utterances.

Figure 2.8 shows three pitch contours for the same expressive utterance *"Go! Do you hear?"*, contained within the *LibriSpeech* test set. They correspond to: the unprotected utterance (red); the `T04`-anonymised utterance (green); the corresponding DTW-aligned utterance (blue). Alignment is shown to improve greatly with the application of DTW.

We applied the same pitch realignment process to the full test set and utterances treated with one of the six systems shown in Table 2.4. Pitch correlation results, shown in all cases with and without the application of DTW alignment, improve markedly for all systems, and for the `T04-p1` system the most. The improvement in this case is substantial, giving values of $\boldsymbol{\rho}^{F_0}$ higher than those for all original systems without DTW alignment. Still, the result of $\boldsymbol{\rho}^{F_0} = 0.83$ is lower than that for all systems with DTW alignment. This result suggests that high values of $\boldsymbol{\rho}^{F_0}$ can be obtained with minimal additional effort, and either that the threshold of 0.3 is far too low, and/or the metric itself is deficient. We stress that we did not use the warped F0 contours to resynthesize speech signals, the naturalness and intelligibility of which would inevitably degrade. Resynthesis would require further work to warp-adjust articulatory or linguistic features, or an alternative approach to DTW, e.g. by operating upon spectral features.

Utility increase with anonymization

The plot to the left in Figure 2.4 shows that some anonymization systems lead to lower WERs than that for unprotected data. This would suggest that, contrary to intuition, anonymization *improves* intelligibility. This would be a rather favorable and unrealistic interpretation.

The ASR model is trained using unprotected data sourced from the *LibriSpeech-train-clean-360* dataset. Conversely, the ASR^{VA} model is trained using the anonymized version of the same dataset. The anonymization system, however, is trained using a much larger amount of data, namely that sourced from the *LibriSpeech-train-clean-100*, *LibriSpeech-train-other-500*, and *VoxCeleb 1-2* datasets. This likely leads to the leaking of anonymization training data into the ASR^{VA} model, implying that it is exposed to more data during training than the ASR model. While comparisons made between two different anonymization systems involve models trained under identical data conditions, comparisons in utility before and after anonymization do not. Results should then be interpreted with appropriate caution. Fairer comparisons of utility before and after anonymization might be made if the data used for the training of the the ASR model were augmented with the same data used in the training of anonymization systems. It should be noted, though, that this would lead to other undesirable issues concerning data overlap.

2.4 VoicePrivacy Challenge 2024

This section provides an overview of the third edition of the VPC, held in 2024 [37]. For this edition, the candidate was actively involved as organizing member of the VoicePrivacy team.

2.4.1 Datasets

VCTK was removed from the evaluation sets, leaving only the LibriSpeech partitions. As detailed in Section 2.3.3, in the previous challenge edition, the attack by ASV^{VA} was consistently more effective on LibriSpeech than on VCTK, making LibriSpeech a representative "worst-case scenario".[2] To simplify the evaluation pipeline and make the challenge more accessible in terms of computational requirements, the exclusive use of LibriSpeech was deemed sufficient for assessing privacy protection.

Around the time of the 2022 edition, the use of self-supervised speech foundation models in VA pipelines began gaining traction within the research community, demonstrating promising results across several studies [75, 76, 77]. However, these models are typically trained on large volumes of unlabeled speech data and thus did not comply with the dataset restrictions enforced in the 2020 and 2022 editions, which limited training to LibriSpeech, LibriTTS, and VoxCeleb-1,2 (see Section 2.2.1). To align with this emerging research direction, the 2024 edition allowed participants to use any training dataset, provided that they specified the datasets used in their system description and notified the organizers in advance.

As detailed later in Section 2.4.2, the 2024 edition of the challenge introduced a new utility metric: speech emotion preservation. To perform evaluation of this task, the IEMOCAP dataset [78] was adopted as a test set. IEMOCAP is an audio-visual emotion recognition dataset comprising motion capture and audio recordings of both improvised and scripted dialogues performed by five female and five male English-speaking actors. The data is annotated with nine emotion classes. For the purposes of the VPC, only the audio modality was used, and only utterances

[2] While no thorough investigation was conducted among the VPC organizers as to why, it is reasonable to believe that the greater effectiveness of ASV^{VA} on LibriSpeech was due to T^{VA} being also taken from LibriSpeech. As a consequence, evaluating on Libri-dev/test is an in-domain task, while VCTK-dev/test are out-of-domain datasets.

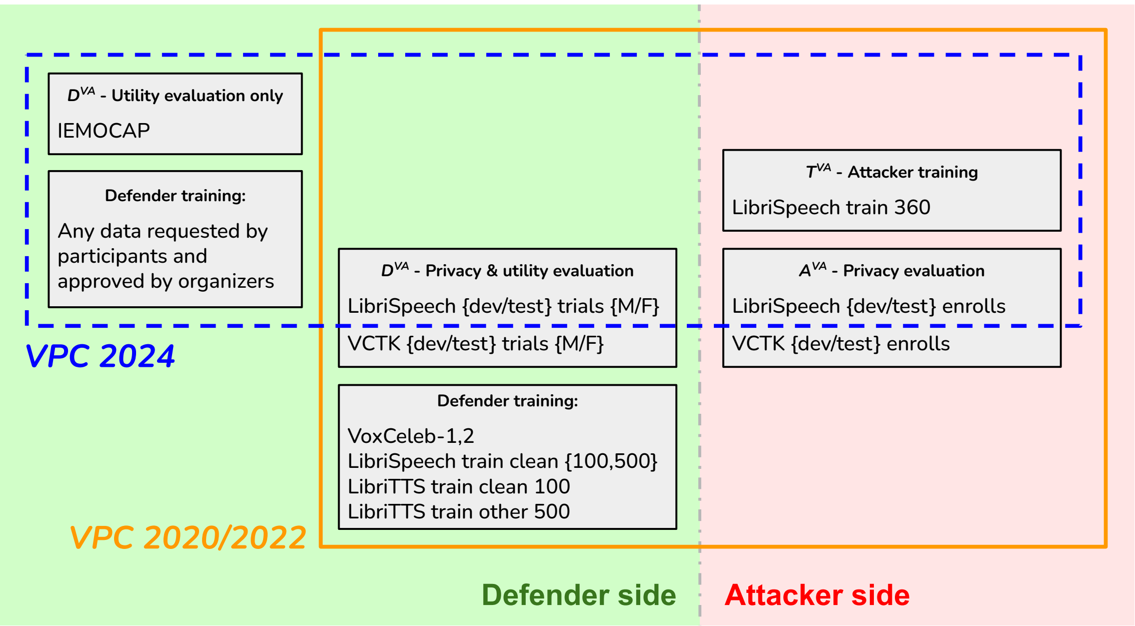

D^{VA} - Utility evaluation only
IEMOCAP
Defender training:
Any data requested by participants and approved by organizers
VPC 2024
D^{VA} - Privacy & utility evaluation
LibriSpeech {dev/test} trials {M/F}
VCTK {dev/test} trials {M/F}
Defender training:
VoxCeleb-1,2
LibriSpeech train clean {100,500}
LibriTTS train clean 100
LibriTTS train other 500
VPC 2020/2022
T^{VA} - Attacker training
LibriSpeech train 360
A^{VA} - Privacy evaluation
LibriSpeech {dev/test} enrolls
VCTK {dev/test} enrolls
Defender side
Attacker side

Figure 2.9: Summary of datasets used in the 2020, 2022, and 2024 VPC editions. The solid orange line encloses the datasets used in the 2020 and 2022 editions; the dashed blue line encloses the datasets of the 2024 edition.

labeled as *neutral*, *happiness*, *sadness*, or *anger* were retained. As IEMOCAP contains relatively few speakers and limited data compared to the other evaluation datasets, a five-fold cross-validation strategy was employed. Refer to [37] for further details.

Figure 2.9 summarizes the use of the various data partitions throughout the three editions of the VPC.

2.4.2 Metrics

Only objective metrics were employed for the 2024 edition. As with the previous editions, the privacy metric is the EER achieved by the model ASV^{VA} trained by a semi-informed attacker. In the 2024 edition, the ASV model is changed to an ECAPA-TDNN with 5 convolutional blocks of with 512 channels and kernel sizes `[5,3,3,3,1]`, as implemented in the *SpeechBrain* package [79]. This change was motivated by the necessity to provide a more updated, PyTorch-based ASV system as backbone for the attack, which was easier to integrate in modern pipelines with respect to the 2022 edition attacker implemented in Kaldi.

Moreover, all data is now anonymized at the utterance level, including A^{VA} and D^{VA}. This configuration was deemed more straightforward and less likely to induce technical errors in the anonymization process, since attacker and defender are now meant to anonymize data using essentially the same strategy.

The main utility metric is the WER computed by an ASR model on D^{VA}. The model is again taken from SpeechBrain and is based on a wav2vec2.0 backbone [80] with a Connectionist Temporal Classification (CTC) head [81]. Contrary to the previous edition, the ASR model weights are directly loaded from a checkpoint provided by SpeechBrain[3] trained on LibriSpeech and Libri-Light [82]; therefore, the model is not trained on anonymized data, but clear, unprotected data.

The only secondary utility metric is based on the ability of the VA system to preserve the emotional content of the anonymized speech. This is done by using a Speech Emotion Recognition (SER) model based on a pretrained and frozen wav2vec2 feature extractor[4] completed with an average pooling layer and a final linear layer. The classification head is trained 5 times in a cross-validation style by optimizing it on 4 folds, testing it on the remaining one, and repeating this process until all folds have been tested on. The evaluation metric is the Unweighted Average Recall (UAR) metric, defined as:

$$\mathrm{UAR} = \frac{1}{N_{\mathrm{class}}} \sum_{i=1}^{N_{\mathrm{class}}} \frac{\mathrm{TP}_i}{\mathrm{TP}_i + \mathrm{FN}_i} \tag{2.4}$$

That is, the mean of the class-wise recalls, where each recall value is weighted according to the population of each class.

As in the 2022 edition, systems are evaluated according to the EER intervals $[10,20), [20,30), [30,40), [40,100)$. For each interval, systems are ranked separately in order of increasing WER and decreasing UAR.

2.4.3 Baselines

Several new baselines were introduced in this challenge edition. Baselines `B1.b` (now renamed `B1`) and `B2` were kept from the previous edition. Baseline `B3` was initially introduced in [83] as a follow-up work of system `T04` [65] from the VPC 2022. Similarly to `T04`, it is ASR+TTS hybrid: phonetic features are extracted from the input signal with a Branchformer-based ASR model [84], then converted again to audio with FastSpeech2 [67], which is conditioned on further inputs: an F0 curve, the phonemes' fundamental frquency (F0), energy and duration, and a speaker embedding. F0 and energy values are perturbed by multiplying them by a random scalar uniformly sampled from the range $[0.6, 1.4]$. The original speaker embedding is extracted with the Global Style Token (GST) model [85]; then, a Generative Adversarial Network (GAN) [86] generates another speaker embedding from scratch. If the two embeddings are required to have cosine distance greater than or equal to 0.3 to guarantee dissimilarity with the original speaker voice; if not, then the GAN keeps generating new embeddings until a suitable one is

[3] https://huggingface.co/speechbrain/asr-wav2vec2-librispeech
[4] https://huggingface.co/facebook/wav2vec2-base

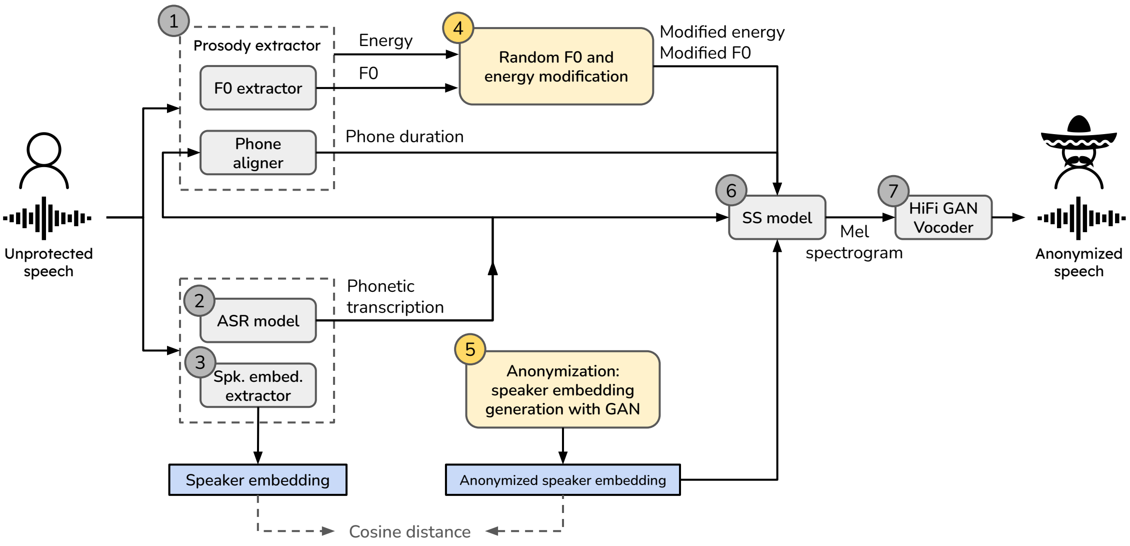


Figure 2.10: Baseline anonymization system B3.

found. The selected synthetic speaker embedding is used to condition the FastSpeech2 model, which generates a Mel spectrogram of the anonymized utterance. As a final step, a HiFi-GAN converts the Mel spectrogram to waveform.

Baseline B4 is based on autoregressive modeling of NAC tokens [87]. As it the candidate's work and constitutes one of the contributions of this dissertation, it is described in Section 4.2.

Baselines B5 and B6 were introduced in [38]. They extract the F0 curve and features of an ASR AM from the input signal. The latter are processed with a Vector Quantization (VQ) bottleneck layer to reduce the amount of residual speaker information they encode. The resulting quantized features are used to synthesize the final anonymized utterance along with the F0 curve and a one-hot representation of the target speaker's voice. The synthesis model is a HiFi-GAN. B5 uses an AM based on a pretrained wav2vec2.0 backbone with three additional TDNN-F layers; B6 simply uses 12 TDNN-F layers.

2.4.4 Submitted systems

The 2024 VPC saw a notable increase in the number of participating teams compared to previous editions, with most teams submitting more than one system variation. Not all submitted systems are directly relevant to this dissertation, since much of the research presented here was carried out before the challenge results were made public — therefore, most of the benchmarking relies on comparisons with the systems that were available prior to the release of the 2024 VPC evaluation plan [37]. This is the case for the work presented in Chapters 3 and 4, with Section 4.2 describing what would later become baseline B4 for the 2024 VPC. Chapter 5 does not propose

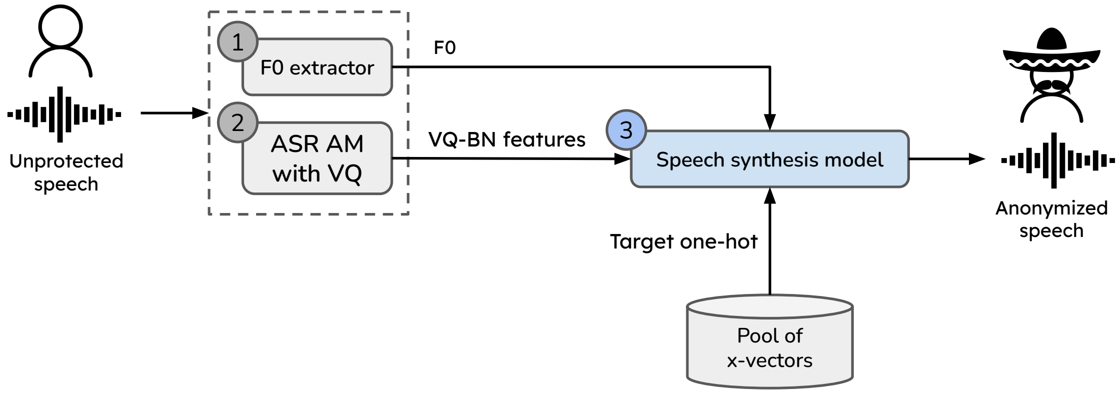


Figure 2.11: Baseline anonymization systems B5 and B6.

new anonymization techniques, but rather investigates the role of the attacker in the task of VA; for reasons that will become apparent later, comparisons with newer VA systems were either irrelevant or unfeasible.

Nevertheless, a number of VPC participants' systems are referenced throughout this dissertation, either as illustrative examples or as supporting evidence for specific conclusions. Therefore, for completeness, we summarize the principal design trends that characterized systems submitted to the 2024 VPC. Notably, it is possible to identify three groups of submitted systems that somehow align with, or are derived from, the three best performing baselines in terms of privacy protection: those that employed a cascade of ASR and TTS systems (similar to B3), those based on NACs (similar to B4), and those that applied VQ to content representations to suppress speaker-specific voice cues (similar to B5). In the following, we provide an overview.

Systems based on ASR + TTS

The following systems are based on transcribing the input utterance either at the word or phoneme level with an ASR model, then synthesizing the anonymized utterance anew using TTS. This approach was first adopted by system T04 in the 2022 VPC [65], and later popularized by baseline B3 [83] in the following edition.

System T08-1 [88] is the concatenation of an ASR system that produces a transcription and a TTS system that re-synthesizes a speech waveform — with no interaction between the two. **Systems T12-2,3** [89] directly modify B3 by conditioning the TTS module on an emotion embedding to improve emotion preservation. A set of speaker embedding extractors are compared, and the GAN-based anonymization function is replaced with the pool-based function of B1. **System T30** [90] uses F0 curves to compute "templates" for each of the four emotions to classify in the SER task. Textual features are extracted from the input utterance and are integrated with one of the emotion templates before being re-synthesized using a TTS model.

Systems based on neural audio codecs

These systems achieve anonymization by manipulating the tokens generated by a NAC. This approach was first used in baseline B4, proposed by the candidate in [87] and described in Section 4.2.

System T10 [91] employs NAC-style residual quantization (see Section 4.1.2). At the first quantizer stage, speaker information is suppressed by subtracting the speaker embedding, while linguistic information is preserved via distillation from a linguistic encoder. Subsequent quantizers are aligned with the F0 curve of the original utterance. **System T12-1** [89] is based on FACodec [92]. The speaker embedding is replaced with an anonymized version obtained through pool-based averaging (as in B1) and Gaussian noise addition. **System T17** [93] also builds upon FACodec, adding an autoencoder-based prosody anonymization module conditioned on a pseudo-speaker embedding. **Systems T38-1,2,3,4,5,6** [94] are all variants of the DISSC [95] architecture. Although not strictly based on NACs, the system performs VC by modeling discrete speech units, following a conceptually similar principle.

Systems based on vector quantization

The following systems employ VQ to discretize the features representing the spoken content of the input utterance, with the aim of suppressing residual speaker-specific cues that may still be present in the content representation. This approach was first proposed with baseline B5 in [76].

System T14 [96] can be seen as an enhanced version of the typical x-vector–based VA pipeline on which B1 is based (see Section 3.2 for further details). A convolutional encoder extracts content-related features, which are then processed by a VQ layer. The F0 curve of the original utterance is also estimated and processed by a Gated Recurrent Unit (GRU), while the speaker embedding is computed using an ECAPA-TDNN [73] model. **Systems T12-5,6** [89] modify B5 by interpolating the F0 curve with its moving-average estimate and adding Gaussian noise to it. **System T19-3** [97] is based on VQMIVC [98], a VC model that disentangles speaker-specific information from content embeddings by minimizing the mutual information between them and the pitch and speaker representations. **Systems T25-1,2** [99] extend B5 by adding a SER module that identifies the emotional state of the original utterance and uses a pool of reference utterances to encode that emotion into a “non-content embedding.” This embedding is then used to condition the synthesis of the HiFi-GAN vocoder.

Table 2.5: Performance metrics (%): EER, WER, and UAR on data processed by the baselines. Performance on original data (Orig.) is also shown.

Metric	Dataset	System						
		Orig.	**B1**	**B2**	**B3**	**B4**	**B5**	**B6**
EER	LibriSpeech-dev (f)	10.51	10.94	12.91	28.43	34.37	35.82	25.14
	LibriSpeech-dev (m)	0.93	7.45	2.05	22.04	31.06	32.92	20.96
	Average dev	5.72	9.20	7.48	25.24	32.71	34.37	23.05
	LibriSpeech-test (f)	8.76	7.47	7.48	27.92	29.37	33.95	21.15
	LibriSpeech-test (m)	0.42	4.68	1.56	26.72	31.16	34.73	21.14
	Average test	4.59	6.07	4.52	27.32	30.26	34.34	21.14
WER	LibriSpeech-dev	1.80	3.07	10.44	4.29	6.15	4.73	9.69
	LibriSpeech-test	1.85	2.91	9.95	4.35	5.90	4.37	9.09
UAR	IEMOCAP-dev	69.08	42.71	55.61	38.09	41.97	38.08	36.39
	IEMOCAP-test	71.06	42.78	53.49	37.57	42.78	38.17	36.13

Other systems

Some participants experimented with methods that do not fall into the previous categories; we now summarize them. **Systems** `T08-2` [88] and `T19-2` [97] are both inspired by the k-Nearest Neighbors (k-NN)-VC framework originally proposed in [100]. In this technique, two sets of speech embeddings are extracted using WavLM [101]: one from the source utterance (source set) and one from a target speaker (target set). Each source embedding is replaced with the average of its k-nearest neighbors from the target set, and the resulting sequence is converted into a waveform through a vocoder. Both systems adapted this approach for the VA task. **System** `T08-2` introduces multiple target sets to combine several pseudo-speakers within a single utterance. **System** `T19-2` extracts feature vectors from different layers of WavLM to capture a broader range of speech cues. **System** `T09` [102] can be viewed as an extension of the k-NN-VC approach: target sets from multiple speakers are extracted and used to estimate a Gaussian Mixture Model (GMM), whose centroids are perturbed to increase anonymization. The resulting GMM is then used to generate embeddings that compose a new target set for conversion.

System `T07` [103] trains multiple VC models, each on a single speaker voice. A pseudo-speaker voice is obtained by fusing these models at the parameter level. An emotion embedding extracted from the input utterance is used to condition the synthesis. **System** `T33` [104] builds upon the FreeVC [105] system. The original feature extractor is replaced with an ASR model and an F0 curve estimator; the input to the VC model consists of the concatenation of the F0 values and both the hidden features and logits of the ASR model. **System** `T8-5` [88] mixes test data from `T08-1` and `T08-2` with a 40%-60% proportion.

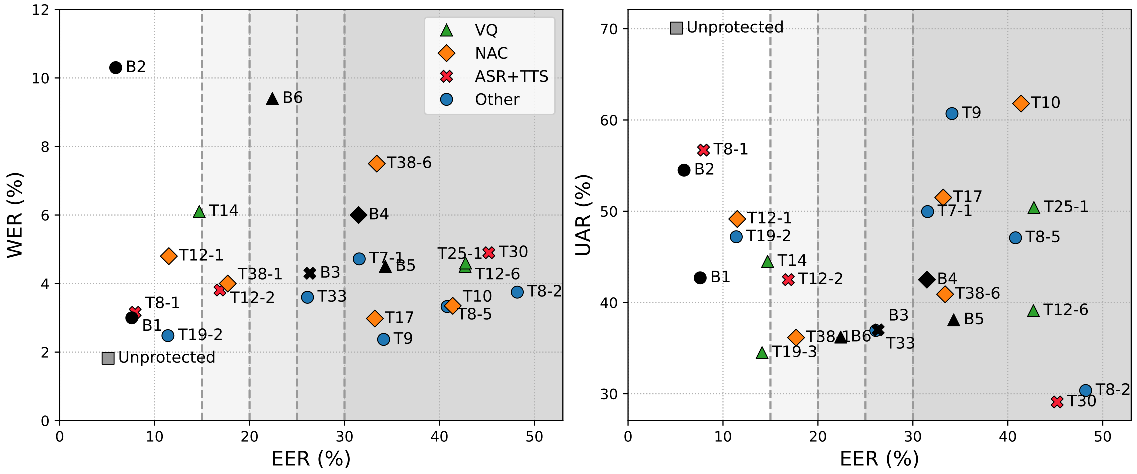


Figure 2.12: Overview of the results of the systems submitted to the 2024 VPC. Also shown are the results of the baselines B1 through B6 and the unprotected data.

2.4.5 Results

In terms of privacy protection, a clear gap is observed between the baselines carried over from the 2022 VPC (B1 and B2, both with EER below 10%) and the newly introduced systems, which represented the state of the art at the time of the challenge (EER above 20%). Among these, B5 achieves the highest EER (≈ 34%), followed by B4 (≈ 31%) and B3 (≈ 26%). Although B1 does not offer strong privacy protection, it achieves the best performance in preserving speech content, with a WER of 2.91%. In contrast, the strongest anonymization systems (B3, B4, and B5) obtain WER scores ranging from approximately 4.5% to 6%. With a UAR of approximately 54%, B2 demonstrates the best emotion preservation performance — likely due to its signal processing-based approach, which modifies the signal at the formant level while leaving prosody untouched. B1 and B4 follow with UAR scores around 42%, while the remaining systems fall below 38%. A summary of these results is provided in Table 2.5.

As we remark in Section 2.4.4, the participants' systems are not central to this dissertation; therefore, we do not report their results in detail. Nevertheless, for completeness, their performance is summarized in Figure 2.12.[5]

[5] For improved visual clarity, the WER value of system T19-3 is omitted from the left plot, as its considerably higher value (19.8%) would disproportionately expand the y-axis scale and obscure differences among the remaining systems.

Chapter 3

Vocoder drift

As we described in Chapter 2, VA pipelines based on x-vector manipulation had a relevant role in the field, starting from original work in [59] and subsequently being employed for several participant systems in the VPC 2022. With the work presented in this chapter, we seek to better understand the behavior of such systems. We introduce the concept of *vocoder drift*, a phenomenon that quantifies the intrinsic contribution of the vocoder to the privacy protection provided by an x-vector–based anonymization pipeline.

3.1 Motivation

Most state-of-the-art VA systems released around the time of the VPC 2022 (including baselines `B1.a` and `B1.b`) followed the blueprint introduced in [59], relying on the extraction and processing of three distinct speech representations:

- a set of linguistic features produced by an ASR model;
- a representation of intonation and prosody, usually in the form of a fundamental frequency (F0) curve;
- an *x-vector*, namely a neural embedding which encodes the voice identity [49].

To conceal the voice identity, the x-vector is typically perturbed by means of an *anonymization function*, thereby obtaining a new *pseudo-speaker* embedding. The three components are then fed into a vocoder to produce an utterance in the voice of the pseudo-speaker. The anonymization function used by two of the three VoicePrivacy baselines utilizes a *pool* of external x-vectors; the pseudo-speaker x-vector is derived from a subset of the furthest vectors in the pool from the input x-vector.

In the 2020 edition of the VPC, 13 out of 16 valid submissions were derived from baseline `B1` [35], which employed the same pool-based x-vector anonymization technique. Among these systems, 10 focused exclusively on modifying the anonymization function in the attempt to achieve more effective anonymization [106, 107, 108, 109]. This trend was also observed in the 2022 edition, where participants focused predominantly upon improving the anonymization function [65,110,111]. X-vector-based anonymization approaches continued to receive attention in the literature even beyond the VPC 2022 [75, 77, 112].

Focusing on the anonymization function to improve VA performance makes intuitive sense: assuming that the feature extraction and VC blocks of the VA system provide perfect disentanglement, all the voice-specific information related to the original speaker should reside in the x-vector extracted from the input utterance; then, perturbing such an x-vector into a new one should result in a different voice identity in the output waveform. However, the VC process cannot be assumed to be perfect. First, the voice identity may not be entirely disentangled from the linguistic and prosodic features of the utterance, voice-specific cues can still leak into the supposedly anonymized waveform. Second, the vocoder may fail to reconstruct a waveform whose resulting x-vector fully matches the intended target x-vector given as input, introducing residual differences between the desired and the synthesized voice identity. In this chapter, we focus on the latter case: specifically, we observe and compare the relative impacts of the vocoder module on the performance of an x-vector–based VA system. We show that the contribution of the vocoder to the final waveform, which we refer to as the *vocoder drift*, can be equal or in some cases even greater than that of the anonymization function. We demonstrate that this phenomenon is also common to many popular vocoders. Collectively, they fail to provide the level of fine-grained control over the input/output x-vector space that would otherwise justify the focus within the community upon the anonymization function.

First, we provide a more technical overview of a typical x-vector–based VA system. Next, we describe vocoder drift, and how it can potentially be exploited by a privacy adversary to reduce the privacy protection provided by the anonymization. We then investigate the causes of vocoder drift and show that it is linked to the interaction between the anonymized x-vector processed by a vocoder and the remaining input features — which are not anonymized, and could therefore still contain speaker-specific cues. Lastly, we present a technique to reduce vocoder drift.

3.2 X-vector–based voice anonymization

We now describe in detail the typical x-vector–based anonymization pipeline outlined in Section 3.1, also illustrated in Figure 3.1. Let $\mathbf{u} \in \mathbb{R}^L$ be an input speech utterance of L samples. The input is first frame-blocked into a sequence of N frames and then decomposed into three separate representations comprising: an F0 curve $\mathbf{f} \in \mathbb{R}^N$ which is intended to encode intonation and prosody; a set of c-dimensional linguistic features $\mathbf{G} \in \mathbb{R}^{c \times N}$ which encode the spoken content (the text); an x-vector $\mathbf{x}_o \in \mathbb{R}^m$ which encodes the speaker identity, where subscript o denotes extraction from an *original* input utterance.

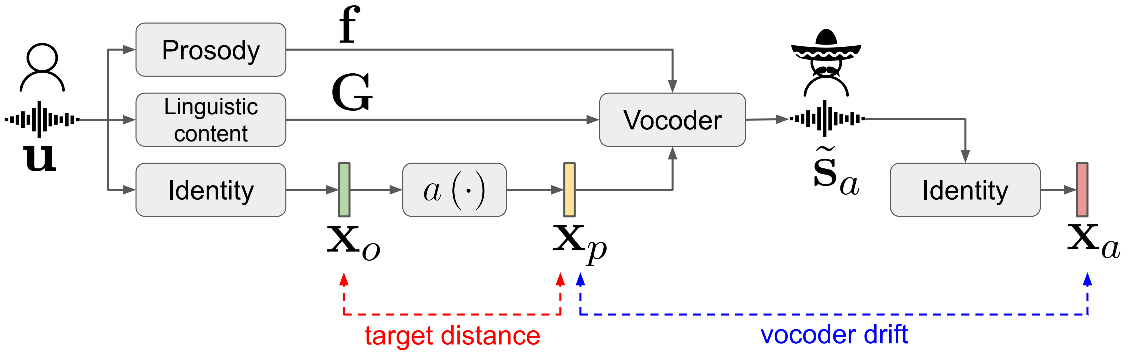


Figure 3.1: Overview of a conventional speaker anonymization system and the different x-vector domains. An intuitive representation of the concepts of *target distance* and *vocoder drift* is shown in the lower part of the diagram.

A vocoder model $V(\mathbf{f},\mathbf{G},\mathbf{x}_o)$ is trained to reconstruct input waveforms from the decomposition. Anonymization is achieved by replacing $\mathbf{x}_o$ with a substitute so as to conceal the speaker identity, but by using the other components unchanged in order to preserve remaining speech attributes. The substitution is performed using an anonymization function $a(\mathbf{x}_o) = \mathbf{x}_p \in \mathbb{R}^m$ to perturb the original x-vector.

An anonymized utterance $\tilde{\mathbf{u}}$ in the voice of a fictitious pseudo-speaker determined by the anonymized x-vector $\mathbf{x}_p$ is then synthesized according to $\tilde{\mathbf{u}} = V(\mathbf{f},\mathbf{G},\mathbf{x}_p)$. The anonymized utterance should maintain the same linguistic and paralinguistic content as the original input signal. As discussed later, an additional x-vector $\mathbf{x}_a$ can be extracted from $\tilde{\mathbf{u}}$ in order to measure privacy.

By convention, $a(\cdot)$ acts to create a new pseudo-speaker using speaker embeddings drawn from an external pool of x-vectors [27, 35, 59, 62, 75, 113]. Given an input $\mathbf{x}_o$, the K vectors within the pool that are furthest from $\mathbf{x}_o$ according to some distance metric are selected and then, from among them, K^* vectors are chosen randomly and averaged to obtain $\mathbf{x}_p$. The design of this function has received considerable attention, with numerous works having investigated how its configuration, the choice of distance metric [114] and the strategy by which x-vectors are selected from the pool [39, 114] influence performance. The participants of the two VoicePrivacy Challenges held in 2020 and 2022 proposed different enhancements to $a(\cdot)$. They include the generation of pseudo-speaker embeddings using a generative adversarial network [65] and adversarial noise [110], among others [111, 115]. None of the participants reported the influence of the vocoder; we show that it too contributes to anonymization and that it can be responsible for a great deal of the privacy protection.

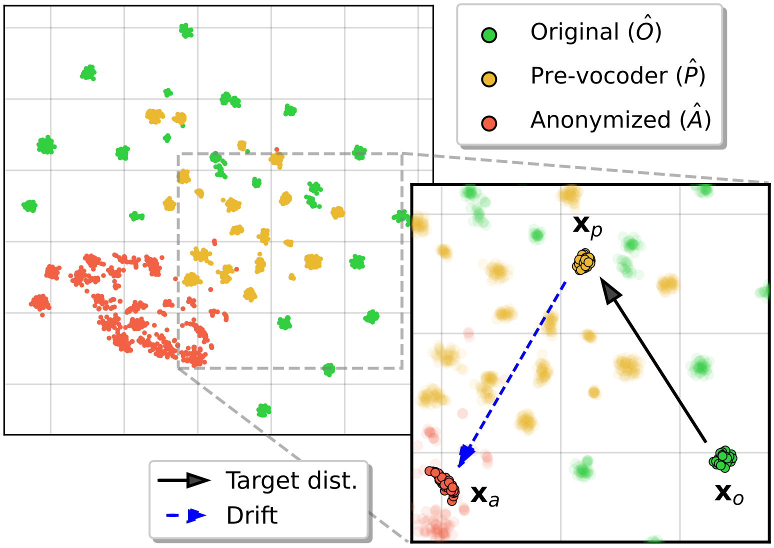


Figure 3.2: t-SNE visualization of the x-vector trajectory for LibriSpeech trial utterances (M) across the three x-vector domains (left). Focus on the trajectory of a single speaker (right). Best viewed in color.

3.3 Vocoder drift

In this section, we introduce the notion of *vocoder drift* [116] and report an investigation of its impact upon x-vector perturbation and privacy.

3.3.1 Definition

Figure 3.1 shows the three x-vectors used in this work. The first $\mathbf{x}_o$ is extracted from the original utterance $\mathbf{u}$ (left in Figure 3.1). A second x-vector $\mathbf{x}_a$ can be extracted from the anonymized utterance $\tilde{\mathbf{u}}$ (right). The third x-vector $\mathbf{x}_p$ is the output of the anonymization function (middle). We denote the separate domains of $\mathbf{x}_o$, $\mathbf{x}_a$ and $\mathbf{x}_p$ as $\hat{O}$ (original), $\hat{A}$ (anonymized) and $\hat{P}$ (pseudo-speaker), respectively.

As described in Section 3.2, a significant portion of research has focused on improving the anonymization function $a(\cdot)$, the general hypothesis being that this component is primarily responsible for ensuring privacy. Intuitively, privacy is improved by increasing the difference between $\mathbf{x}_o$ and $\mathbf{x}_p$, e.g. according to the cosine distance. With the focus being upon the anonymization function, there is an inherent, perhaps unrealistic assumption that the vocoder preserves this distance such that the difference, which we term as the *drift*, between the x-vectors at the input ($\mathbf{x}_p$) and that which can be extracted from the output ($\mathbf{x}_a$) is only modest. We seek to test this assumption.

Table 3.1: Average target distance and drift for each vocoder and each test set of LibriSpeech and VCTK, separated by speaker sex. All cosine distances have a standard deviation between 0.05 and 0.10.

	target	**drift**		
		HiFi-GAN	NSF	HiFi-NSF
LibriSpeech (F)	1.3	0.62	0.91	0.97
LibriSpeech (M)	1.2	0.56	0.80	0.94
VCTK (F)	1.3	0.67	0.92	0.94
VCTK (M)	1.3	0.59	0.90	0.90

We model the relationship between the $\hat{P}$ and $\hat{A}$ domains with a function $v\left(\mathbf{x}_p\right) = \mathbf{x}_a$. It allows us to define the trajectory of an x-vector through the whole anonymization system as $v \circ a : \mathbf{x}_o \mapsto \mathbf{x}_a$, where $\circ$ denotes function composition.

In seeking to quantify the impact of $v\left(\cdot\right)$ on the x-vector trajectory, we define two metrics. Let d be some distance measure over $\mathbb{R}^m$. We then define:

- $d(\mathbf{x}_o,\mathbf{x}_p)$ as the *target* distance, a measure of how far $\mathbf{x}_o$ is perturbed away from its original position according to $a\left(\cdot\right)$;
- $d(\mathbf{x}_p,\mathbf{x}_a)$ as the vocoder *drift*, a measure of the shift between the input x-vector $\mathbf{x}_p$ and that which can be extracted from the vocoder output $\mathbf{x}_a$, introduced by means of $v\left(\cdot\right)$.

A graphical depiction of vocoder *drift* and *target* distance is shown in the right panel of 3.2. Intuitively, it is desirable that *drift* $\ll$ *target*, which means the anonymization system provides fine-grained control over the final position of $\mathbf{x}_a$: it is close to the targeted pseudo-speaker embedding $\mathbf{x}_p$. If this is not the case, then the x-vector trajectory is determined in considerable part by $v\left(\cdot\right)$; the x-vector extracted from the output $\mathbf{x}_a$ is far from the target and the system does not provide fine-grained control over the x-vector space in $\hat{A}$.

We first compute and compare the average values of vocoder drift and target distance over the VPC 2022 datasets for 3 different vocoders used in a VA system. Then, we perform ASV experiments in the domains of $\hat{O}$, $\hat{P}$ and $\hat{A}$ to quantify the contribution of each step of the VA pipeline to the overall privacy protection of the anonymized utterance.

3.3.2 Experimental setup and results

Our approach is based on the pipeline described in [75]. The F0 curve is estimated using YAAPT [60]. The linguistic feature extractor is a HuBERT-based soft content encoder [117] and x-vectors are extracted using ECAPA-TDNN [73]. We experimented with three vocoders: the HiFi-GAN [63], originally used in [75]; the NSF model [61] as used by baseline B1.a of the VPC held in 2022; a variation of the HiFi-GAN which uses a NSF model as generator, as used by baseline `B1.b` of the same VPC edition [27]. We use the conventional pool-based anonymization function $a(\cdot)$ described in Section 3.2 with $K = 200$, $K^* = 100$, and a cosine distance metric.

Evaluation is performed using the VPC 2022 database and standard protocols as described in Section 2.3. The *LibriTTS-train-clean-100* dataset is used for vocoder training. The *LibriSpeech-test-clean* and *VCTK* datasets (decomposed into male and female subsets) are used for evaluation. The external pool of x-vectors is derived using the *LibriTTS-train-other-500* [47] dataset. Privacy is evaluated using ASV experiments comprising a set of enrollment utterances that an attacker attempts to match to a set of protected (anonymized) trial utterances. ASV is performed by scoring x-vectors with the cosine distance and without any additional backend processing [73, 75]. We follow the VoicePrivacy-defined approach to measure privacy impacts. We report a set of ASV experiments using different combinations of x-vectors. In all cases, privacy is measured using estimates of the EER. Enrollment and trial utterances are as defined by the VPC protocol (see Section 2.3). There are several enrollment utterances per speaker. Individual x-vectors are extracted from each, averaged, and compared to a number of trial utterances. For each utterance, we extract the set of $\mathbf{x}_o$, $\mathbf{x}_p$ and $\mathbf{x}_a$ x-vectors. Each set of experiments is conducted three times, with each iteration using one of the three different x-vectors. Results using the set containing $\mathbf{x}_o$ x-vectors ($\hat{O}$ domain) provide a baseline. Those derived from the set of $\mathbf{x}_p$ x-vectors ($\hat{P}$ domain) provide an indication of the contribution to privacy of the anonymization function $a(\cdot)$. Results using final set containing $\mathbf{x}_a$ x-vectors ($\hat{A}$ domain) provide an indication of the contribution of the vocoder function $v(\cdot)$. Once again, we report results for the same experiment using all three vocoders.

We compute the average *drift* and *target* for each database subset and each vocoder: results are shown in Table 3.1. The *target* is in the order of 1.3 for all four subsets. The value of these distances lies in their *comparison* to estimates of the *drift* shown in the last three columns. For the HiFi-GAN vocoder, the drift is almost half the target distance. Lying between 0.8 and 0.97, the drift for the NSF and HiFi-NSF vocoders is substantially greater still, with drift distances almost as large as target distances. These results show that the control over the x-vector domain $\hat{A}$ is potentially low and suggest that the x-vector anonymization and vocoder functions have an almost-comparable contribution to x-vector perturbation.

Results on the impact upon *privacy* are shown in Table 3.2, for the same database subsets as in Table 3.1. Baseline results for the $\hat{O}$ domain show EERs of approximately 1%. In the $\hat{P}$ domain, increases in the EER to between 2.5% and 5.6% indicate that the anonymization function delivers only a low level of privacy. In the $\hat{A}$ domain, however, EERs are substantially higher for all three vocoders, if still far from providing perfect privacy (EERs of 50%). The comparison of results for $\hat{P}$ and $\hat{A}$ domains show that the vocoder plays a dominant role; most of the anonymization can be attributed to vocoder drift. We explored this phenomenon with t-SNE visualizations [118] of

Table 3.2: Privacy protection of the x-vector domains at different stages of the anonymization pipeline (EER, %) on test sets of LibriSpeech and VCTK, separated by speaker sex.

	$\hat{O}$ dom.	$\hat{P}$ dom.	$\hat{A}$ dom.		
			HiFi-GAN	NSF	HiFi-NSF
LibriSpeech (F)	0.54	2.51	15.0	17.9	16.2
LibriSpeech (M)	0.88	2.99	14.5	20.3	19.0
VCTK (F)	1.13	5.59	25.3	31.0	28.1
VCTK(M)	0.17	3.04	18.5	16.7	19.1

pooled x-vectors. Results are illustrated in Figure 3.2, which depicts a distribution of x-vectors for the male partition of the LibriSpeech dataset. In the $\hat{P}$ domain, speaker clusters are still clearly distinguishable, while the bulk of the anonymization can be attributed to vocoder drift.

One could claim that these findings are neither surprising, nor cause for concern. There is no guarantee that the vocoder function $v(\cdot)$ is invertible in any way which would allow the recovery of x-vector inputs $\mathbf{x}_p$ in the $\hat{P}$ domain. Since the attacker does not have access to the $\hat{P}$ domain, but only to the $\hat{A}$ domain, whether anonymization is attributed to the anonymization function or the vocoder function is of little consequence. In the next section, we disprove these arguments and show that an attacker *can* learn this function or, more specifically, how to undo it. Armed with the inverse function $v^{-1}(\cdot)$, an attacker can estimate an x-vector in the $\hat{P}$ domain that corresponds to an x-vector in the $\hat{A}$ domain and hence *reverse* the anonymization.

3.4 Drift reversal attacks

At the time when the candidate carried out his research into vocoder drift, relatively little attention had been given to attacks on VA systems. The semi-informed attack defined in the 2022 edition of the VPC was the standard, with [113] being the only work to investigate an alternative approach based on the use of a rotation matrix to estimate speaker embeddings $\mathbf{x}_o$ in the unprotected domain $\hat{O}$ from protected x-vectors $\mathbf{x}_a$ in $\hat{A}$. In the following, we instead propose an attack that aims to revert only the *vocoder drift* to recover an estimate of $\mathbf{x}_p$ in $\hat{P}$.

3.4.1 Definition and implementation

In the case that the bulk of the anonymization performance can be attributed to the vocoder function $v(\cdot)$ instead of the anonymization function $a(\cdot)$, a drift reversal attack can be mounted to undo most of the protection. Let $\mathbf{u}^{(e)}$ be an original (i.e. unprotected) enrollment utterance. An attacker can derive a representation of this signal in the $\hat{P}$ domain by extracting an x-vector

$\mathbf{x}_o^{(e)}$ and then by computing $a\left(\mathbf{x}_o^{(e)}\right) = \mathbf{x}_p^{(e)}$. Now let $\tilde{\mathbf{u}}^{(t)}$ be an anonymized trial utterance with corresponding x-vector $\mathbf{x}_a^{(t)}$ in the $\hat{A}$ domain. The attacker can estimate a representation in the $\hat{P}$ domain $\mathbf{x}_p^{(t)}$ by reversing the vocoder drift, i.e. by computing $v^{-1}\left(\mathbf{x}_a^{(t)}\right)$.

While the inverse function is not analytically tractable, the attacker can attempt to *learn* a function $g_\theta(\cdot) \approx v^{-1}(\cdot)$ using a database of training pairs $\mathbf{x}_{p_i}$ and anonymized utterances $\tilde{\mathbf{u}}_i$. Function g_θ can be learned using a neural network to map an anonymized utterance $\tilde{\mathbf{u}}$ to an approximation of the corresponding x-vector $\mathbf{x}_p$ in $\hat{P}$. This can be achieved by optimizing the objective function

$$\min_\theta d\left(\mathbf{x}_p, g_\theta\left(\tilde{\mathbf{u}}\right)\right) \tag{3.1}$$

where d is the cosine distance. Training pairs $\left\{\left(\mathbf{x}_{p_i}, \tilde{\mathbf{u}}_i\right)\right\}_i$ can be obtained by applying anonymization to *any* appropriate (even unlabeled) speech dataset.

3.4.2 Experimental setup and results

Because function g_θ is effectively an x-vector extraction operation, we fine-tune a pretrained ECAPA-TDNN model to learn it. In line with the VoicePrivacy protocol, the model is trained using the *LibriSpeech-train-clean-360* dataset, although approximately 3% of the data is set aside for validation purposes. Still in line with the VoicePrivacy protocol, anonymization is performed at the *speaker level* (see Section 2.1.2) in deriving $\mathbf{x}_a$ for each enrollment and trial utterance, instead of at the utterance level. The network is fine-tuned for 3 epochs using Adam optimizer [119] with a learning rate of $5 \cdot 10^{-5}$ and a batch size of 8. Validation is performed every 200 iterations. Attacks are performed using the network for which the validation loss is lowest.

We compare the drift reversal attack to related VoicePrivacy *lazy-informed* and *semi-informed* attacks. For the former, the attacker compares enrollment and trial utterances which are both in the $\hat{A}$ domain, but with an ASV model trained using data in the $\hat{O}$ domain; other than by anonymizing the enrollment utterance, there is no compensation for operating upon anonymized data. The *semi-informed* attacker makes greater effort and uses an ASV system that is trained using an independent set of similarly-anonymized data. The latter is the default VoicePrivacy attack model.[1] The *lazy-informed* attack is implemented using the original, pretrained ECAPA-TDNN for x-vector extraction. The *semi-informed* attack is performed using an ECAPA-TDNN model which is fine-tuned using Additive Angular Margin (AAM)-softmax loss [120] and the same training settings as the drift reversal attack model.

[1] It could be argued that drift reversal is also a *semi-informed* attack, since it involves re-training a model on anonymized data (albeit unlabeled). However, for clarity, we use the term *semi-informed* to refer to the attack method used in the VPC 2022, as described in Section 2.1.2.

Table 3.3: Performance of the proposed drift-reversal attack compared to a lazy-informed attack and a supervised semi-informed attack (EER, %) on the LibriSpeech and VCTK test sets.

Vocoder	Dataset	Unprotected	Lazy informed	Semi informed	Drift reversal
HiFi-GAN	LibriSpeech (F)	0.54	11.3	3.21	**3.10**
	LibriSpeech (M)	0.88	10.9	**1.78**	4.45
	VCTK (F)	1.13	19.2	13.3	**7.53**
	VCTK (M)	0.17	11.0	8.14	**3.79**
NSF	LibriSpeech (F)	0.54	15.3	**1.92**	5.05
	LibriSpeech (M)	0.88	13.8	**1.94**	6.04
	VCTK (F)	1.13	24.7	**15.7**	16.5
	VCTK (M)	0.17	**9.81**	12.3	10.7
HiFi-NSF	LibriSpeech (F)	0.54	12.6	**4.01**	4.23
	LibriSpeech (M)	0.88	14.7	**2.23**	4.90
	VCTK (F)	1.13	22.8	18.4	**14.1**
	VCTK (M)	0.17	13.5	11.7	**11.1**

Privacy evaluation results in terms of EER estimates are presented in Table 3.3 for each vocoder and each dataset. EER results for unprotected data (no anonymization) are shown in column 3 and provide a reference against which EERs for protected data can be compared. Results for the *lazy-informed* attack are shown in column 4 and show substantial privacy gains (higher EERs). This setting, however, gives a false sense of protection. Results for the *semi-informed* attack shown in column 5 show considerably lower privacy gains; by retraining the ASV system using similarly anonymized data, the attacker can undo the anonymization to some degree. Results for the drift reversal attack also show universally lower privacy gains compared to the lazy-informed attack. These results add to the evidence that the role played by vocoder drift in anonymization is substantial and is also a potential weakness that can be exploited by an adversary. The most powerful of the 3 attacks for each vocoder and dataset is highlighted in bold face and, for 5 of the 12 cases, the most powerful attack is drift reversal.

3.5 The cause of vocoder drift

In this section, we describe what we believe to be the source of vocoder drift and present a set of experiments which validate our hypothesis [121].

3.5.1 Feature mismatch

As mentioned in Section 3.2, the vocoder model is trained in self-supervised fashion to reconstruct input signals $\mathbf{u}$ at the output. While, ideally, input components $\mathbf{f}$, $\mathbf{G}$ and $\mathbf{x}_o$ should be disentangled from one another – so that none contains any information that is also contained in any other – there is no explicit incentive in the training criterion of any of the three extraction models which would encourage the learning of disentangled representations. Previous work has confirmed that the representations are indeed *entangled* to some extent. For example, results in [53, 76] show that speaker-related information, normally captured in $\mathbf{x}_o$, can leak into linguistic representations $\mathbf{G}$. The vocoder can hence learn to rely on such mutual dependencies between input features in learning how it should reconstruct $\tilde{\mathbf{u}}$.

Through anonymization, original speaker embeddings $\mathbf{x}_o$ are substituted by pseudo-speaker embeddings $\mathbf{x}_p$, and used by the vocoder to reconstruct a speech signal using the F0 curve $\mathbf{f}$ and linguistic features $\mathbf{G}$ extracted from the input speech signal corresponding to x-vector $\mathbf{x}_o$. The new pseudo-speaker embedding will hence not *match* any speaker-related information contained in $\mathbf{f}$ and $\mathbf{G}$. This results in a *mismatch* with the data distribution learned by the vocoder at training time. It is our hypothesis that this mismatch is the source of vocoder drift.
Alternatively, the idea of mismatch could be thought of as different sets of features encoding speaker-specific information about both the original speaker and the pseudo-speaker; these two sources of information get "re-entangled" in the synthesis process in unexpected ways, causing vocoder drift.

We verified our hypothesis with an experiment in which we anonymized a set of utterances $\mathbf{u}$ and computed original x-vectors $\mathbf{x}_o$ and corresponding pseudo-speaker embeddings $a(\mathbf{x}_o) = \mathbf{x}_p$. However, rather than synthesizing new waveforms according to the usual approach $\mathrm{V}\left(\mathbf{f}, \mathbf{G}, \mathbf{X}_p\right)$, we compute instead $\mathrm{V}(\mathbf{f}, \mathbf{G}, \mathbf{X}_i)$, where $\mathbf{x}_i$ is an interpolation between $\mathbf{x}_o$ and $\mathbf{x}_p$:

$$\mathbf{x}_i = \mathbf{x}_o + \lambda(\mathbf{x}_p - \mathbf{x}_o) \tag{3.2}$$

The parameter $\lambda \in [0, 1]$ acts to control the distance between $\mathbf{x}_i$ and either $\mathbf{x}_o$ or $\mathbf{x}_p$. In line with definitions presented in Section 3.3.1, we term $d(\mathbf{x}_o, \mathbf{x}_i)$ the *target distance*. The target distance can be interpreted to reflect the mismatch between the speaker embedding that would naturally complement $\mathbf{f}$ and $\mathbf{G}$ and the embedding received by the vocoder. By adjusting λ, we conducted a set of anonymization experiments with increasing target distances, i.e. higher values of λ, equivalent to increasing feature mismatch. For each experiment, we also measure the resulting vocoder drift. A positive correlation between drift and target distance would then suggest that vocoder drift does indeed have some dependency on the mismatch between vocoder input features.

3.5.2 Experimental setup and results

We conducted experiments with values of $\lambda = \{0, 1/3, 1/2, 1\}$. In the case of $\lambda = 0$, (3.2) reduces to $\mathbf{x}_i = \mathbf{x}_o$, which corresponds to the absence of anonymization (i.e. $a(\cdot)$ is not applied): the system performs copy-synthesis. Conversely, in the case of $\lambda = 1$, (3.2) reduces to $\mathbf{x}_i = \mathbf{x}_p$: the

Table 3.4: Average target distance and drift (without and with compensation) for the four different 2022 VPC data subsets and for four different values of λ.

	$\lambda = 0$ (copy-synthesis)			$\lambda = 1/3$		
	target	drift	drift (comp.)	target	drift	drift (comp.)
LibriSpeech (F)	0	0.29	0.047	0.13	0.38	0.049
LibriSpeech (M)	0	0.27	0.047	0.11	0.35	0.048
VCTK (F)	0	0.29	0.049	0.11	0.36	0.051
VCTK (M)	0	0.29	0.049	0.08	0.35	0.049

	$\lambda = 1/2$			$\lambda = 1$ (normal anon.)		
	target	drift	drift (comp.)	target	drift	drift (comp.)
LibriSpeech (F)	0.35	0.50	0.054	1.0	0.63	0.052
LibriSpeech (M)	0.31	0.48	0.051	1.0	0.65	0.052
VCTK (F)	0.30	0.49	0.084	1.0	0.69	0.082
VCTK (M)	0.26	0.46	0.062	1.1	0.79	0.078

pseudo-speaker embedding is employed during synthesis as with usual anonymization. Values of $\lambda = 1/3$ and $1/2$ correspond to different interpolations between $\mathbf{x}_o$ and $\mathbf{x}_p$. We measured the target distance and vocoder drift for all four configurations.

Results are reported in Table 3.4, which shows the target distance and drift in the first two columns of each set of results for each value of λ. Results are shown separately for LibriSpeech and VCTK datasets and for male and female subsets in both cases. A degree of positive correlation between λ and both the target distance and drift is apparent. For $\lambda = 0$, the target distance is always 0 (since $\mathbf{x}_i = \mathbf{x}_o$) and the drift is consistently in the order of 0.28. For $\lambda = 1/3$, the target distance increases to an average of 0.1 and the drift to an average of 0.36. Both the target distance and drift increase further for higher values of λ: such a correlation is evident when plotting the two metrics against one another for a whole data partition and different values of λ, as in Figure 3.3. These results show that, the greater the degree of mismatch between input features, the greater is the vocoder drift. This in turn implies that greater target distances incur less control over the x-vector space. However, but not surprisingly, for copy-synthesis when $\lambda = 0$, the drift is still substantial. For this configuration, there is no mismatch in the input features; those used for reconstruction are exactly those extracted from the input signal. This suggests that a component of the drift stems from the intrinsic nature of the waveform reconstruction process. In the following section, we report an approach to compensate for the vocoder drift.

3.6 Drift compensation

Vocoder drift, while advantageous in terms of anonymization [116], can be undesirable in that it prevents fine-grained control over the x-vector space. Because the impact of the vocoder upon the x-vector space can dominate that of the anonymization function, this lack of control impedes the design of better anonymization functions. Hence, even if lower vocoder drift might initially degrade anonymization performance, it may deliver better control over the x-vector

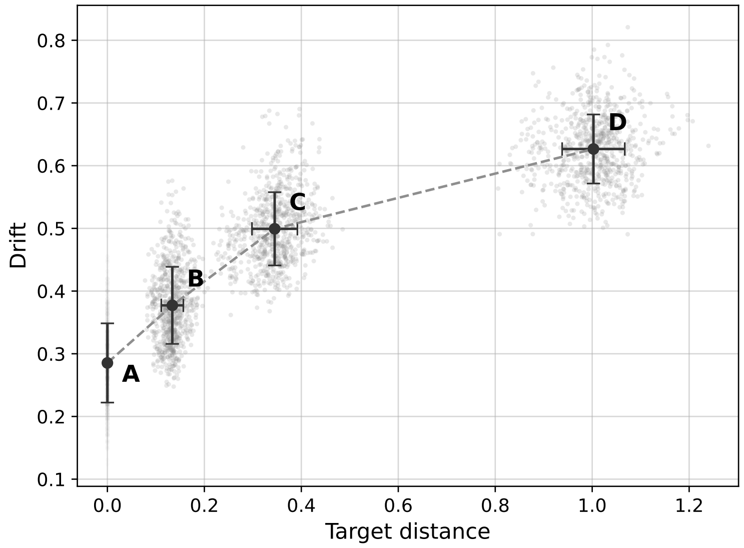


Figure 3.3: Vocoder drift plotted against target distance for the LibriSpeech dataset and for female speakers. Dots and bars represent mean and standard deviation of each of the following experimental setups: (A) $\lambda = 0$; (B) $\lambda = 1/3$; (C) $\lambda = 1/2$; (D) $\lambda = 1$.

space and then be beneficial to the future development of better anonymization functions. In this section, we introduce a new technique for vocoder drift compensation [121]. It is based upon the iterative *alignment* of $\mathbf{x}_a$ to $\mathbf{x}_p$ at inference time.

3.6.1 X-vector alignment

Our goal is to adjust the matrix $\mathbf{X}_i$ so as to reduce the mismatch to $\mathbf{G}$ and $\mathbf{f}$ in order then to reduce vocoder drift. This adjustment can be formulated as an optimization problem:

$$\mathbf{X}_i^* = \operatorname*{argmin}_{\mathbf{X}_i} \; d\Big(\underbrace{f\big(\mathrm{V}(\mathbf{f},\mathbf{G},\mathbf{X}_i)\big)}_{\mathbf{x}_a}, \mathbf{x}_p\Big) \tag{3.3}$$

where d is again the cosine distance. In essence, we seek to adjust $\mathbf{X}_i$ so as to minimize the cosine distance between $\mathbf{x}_p$ (the x-vector vocoder input) and $\mathbf{x}_a$ (the x-vector extracted from its output). The resulting, optimized matrix $\mathbf{X}_i^*$ is then used to synthesize an anonymized utterance $\tilde{\mathbf{u}}_a^*$, whose drift-compensated x-vector we denote as $\mathbf{x}_a^*$. We optimize the objective function directly at inference time via gradient descent. With this approach, the drift can be arbitrarily reduced by any desired amount, at the cost of proportionately increasing the computation time required to synthesize the anonymized waveform.

Table 3.5: ASV results (EER, %) for the 2022 VPC test sets, using the same set of different x-vector speaker embeddings as in Figure 3.4.

	$\mathbf{x}_o$	$\mathbf{x}_p$	$\mathbf{x}_a$	$\mathbf{x}_a^*$ (comp.)
LibriSpeech (F)	0.54	2.51	15.0	2.75
LibriSpeech (M)	0.88	2.99	14.5	3.34
VCTK (F)	1.13	5.59	25.3	9.20
VCTK(M)	0.17	3.04	18.5	5.23

3.6.2 Experimental setup and results

We optimise (3.3) at the utterance level using Adam [119] with a learning rate of 5e−3. Optimisation runs for a maximum of 150 steps, but stops earlier if the drift falls below 0.05 (set arbitrarily to reduce processing time). The impact of drift compensation is then observed by repeating the experiments described in Section 3.5.1 but with $\mathbf{X}_i$ replaced by drift compensated versions $\mathbf{X}_i^*$ and by observing the reduction in vocoder drift.

Results are shown in the third columns of each block in Table 3.1. Drift compensation reduces the vocoder drift for all values of λ. For $\lambda = \{0, 1/3\}$, 150 optimisation steps are generally sufficient for the drift to reach the lower bound of 0.05, for all datasets. This is also the case for the LibriSpeech dataset for $\lambda = \{1/2, 1\}$. For the VCTK dataset, we obtain drift values of approximately 0.07 — slightly higher than LibriSpeech, yet still considerably lower than the initial vocoder drift. Informal listening tests show that drift compensation introduces no discernible degradation to speech quality — any differences are negligible to the point that signals generated with and without drift compensation are difficult to tell apart.

If the vocoder drift is responsible for the bulk of anonymization performance, and if drift compensation performs as intended, then the application of drift compensation is expected to result in degraded anonymization performance. We performed a set of ASV experiments to observe the impact. Experiments were conducted according to the protocol described in the VPC 2022 evaluation plan [27]. For each dataset, the experiment is run four times, each time using one of the set of x-vectors ($\mathbf{x}_o$, $\mathbf{x}_p$, $\mathbf{x}_a$, $\mathbf{x}_a^*$) for each utterance. The results are reported in Table 3.5.

As expected, low EERs for x-vectors $\mathbf{x}_o$ increase for $\mathbf{x}_p$ and even more noticeably for $\mathbf{x}_a$, indicating the dominant impact of the vocoder upon anonymization. This is especially evident in VCTK partitions, likely because of a domain mismatch with the HiFi-GAN vocoder which, in accordance with the VPC 2022 protocol, is trained on *LibriTTS-train-clean-100*. EERs for x-vectors $\mathbf{x}_a^*$ are close to those of $\mathbf{x}_p$, indicating successful vocoder drift compensation. This result can also be observed visually in Figure 3.4, which shows t-SNE visualizations [118] of all four x-vector embeddings for the LibriSpeech dataset and female speakers (both trial and enrollment

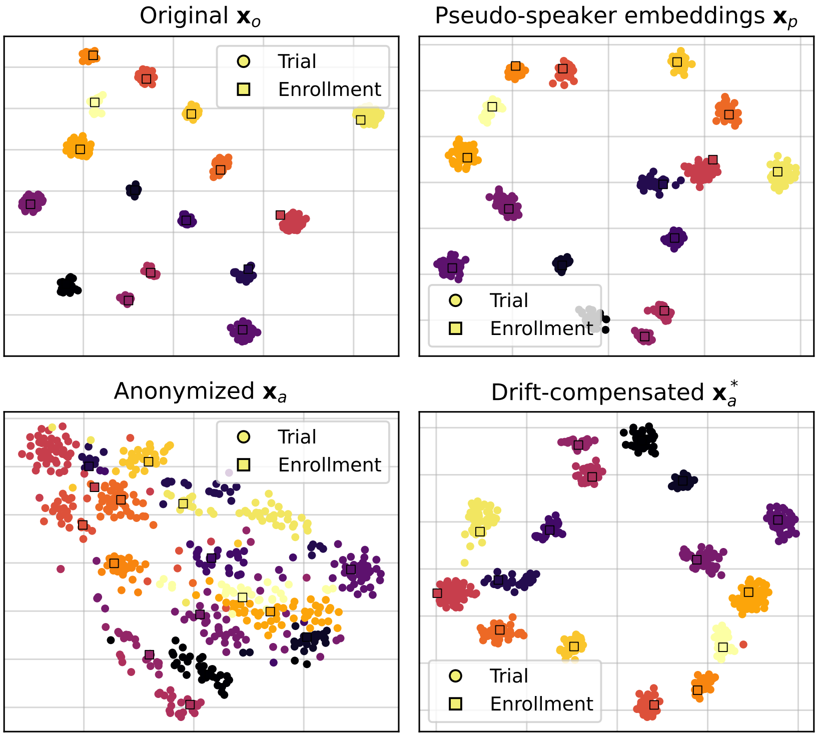


Figure 3.4: t-SNE visualizations of four different x-vector spaces and embeddings for the enrollment and trial utterances of the LibriSpeech dataset and female speakers. Different colors correspond to different speakers. From left to right: original x-vectors $\mathbf{x}_o$, pseudo-speaker embeddings $\mathbf{x}_p$, anonimised embeddings $\mathbf{x}_a$, anonymized and drift-compensated embeddings $\mathbf{x}_a^*$.

utterances). The effect of drift is clearly visible upon the comparison of the visualizations for $\mathbf{x}_p$ and $\mathbf{x}_a$: in the latter, embeddings are notably more dispersed. The visualization for $\mathbf{x}_a^*$ shows that drift compensation reduces the dispersion, giving compact clusters once more.

3.7 Conclusions

In this chapter, we introduced the concept of vocoder drift, defined as the distance between the x-vector provided as input to the vocoder during the VA task and the x-vector extracted from the resulting anonymized utterance. Within the context of x-vector–based VA, this can be interpreted as the degree to which the vocoder influences the overall anonymization process. Our analysis demonstrated that, for several vocoders commonly used in the literature, this effect is substantial and contributes significantly to anonymization. This creates a false sense of privacy protection and exposes a vulnerability: as we have shown, an attacker can learn to reverse the effects of vocoder drift, effectively bypassing the anonymization it provides.

We also observed that the magnitude of vocoder drift appears to be proportional to the mismatch between the linguistic features extracted from the original signal and the anonymized x-vector provided to the vocoder. This suggests that vocoder drift may be caused by an unexpected “re-entanglement” of conflicting speaker information between x-vector and linguistic features. Finally, we proposed a technique that adjusts the x-vector input to the vocoder at inference time to reduce vocoder drift. As expected, eliminating this drift also reduces the level of privacy protection. However, the technique remains valuable for the development and evaluation of anonymization functions, as it allows for isolating their contribution to privacy protection from those introduced by vocoder drift.

In light of the findings presented in this chapter, two fundamental conclusions can be drawn regarding the task of VA:

- **Prefer simple pseudo-speaker selection strategies.** In x-vector–based speaker anonymization, the pseudo-speaker x-vector and the x-vector extracted from the final anonymized utterance differ considerably, as shown by the phenomenon of vocoder drift. Consequently, the research effort dedicated to designing complex anonymization functions [65, 75, 77, 106, 107, 108, 109, 110, 111, 112, 115] might be misplaced. The final position of the pseudo-speaker’s x-vector is changed by the vocoder, likely due to the re-entanglement of personal information coming from the linguistic features, diminishing the relevance of the anonymization function itself. The author of [38] recommends the use of simple pseudo-speaker selection strategies with well-defined behavior, such as a random pick among a set of pre-defined voices, as they allow the most precise estimation of privacy protection (see the reference for details). We support this recommendation for a further reason: without more well-behaved vocoders, i.e. vocoders that do not present drift, complex anonymization functions are simply unnecessary.

- **Focus on effective VC.** The correlation between vocoder drift and vocoder input mismatch suggests that the linguistic features used in the x-vector–based VA pipelines presented throughout this chapter still contain speaker-specific information. The assumption made in [59] and in many derived works is that the vocoder is capable of changing the voice identity of the original speaker solely based on in the input x-vector, selectively disentagling and ignoring the speaker-specific information found in the linguistic features. However, given that a semi-informed attack is still able to recover the voice identity of the original speaker (sometimes with very low EER), this assumption appears to be incorrect. A possible solution to the problem might be the use of different, more effective VC techniques in VA systems, which might result in stronger privacy protection. Ideally, such techniques should attempt to either purge or ignore speaker-specific information in the linguistic features (and any further features) extracted from the original signal, and create a new voice identity based exclusively on the provided pseudo-speaker representation. The following chapter reports an investigation in this direction.

Chapter 4

Voice anonymization with neural audio codecs

The work on vocoder drift, presented in Chapter 3, revealed a clear trend: most VA systems at the time followed the pipeline blueprint introduced in [59]. We identified some limitations of such models, emphasizing the need to shift focus from designing x-vector anonymization functions to developing feature extraction and VC methods that can effectively erase speaker-specific voice cues from the anonymized utterance. The work described in the first part of this chapter represents an effort to move beyond the conventional x-vector–based VA framework and explore alternative, more recent speech synthesis techniques. Specifically, we focus on the use of Neural Audio Codecs (NACs) paired with language modeling, which we show to achieve stronger privacy protection than x-vector–based baselines. This comes at the cost of a marginal decrease in content preservation capabilities: in the second part of the chapter, we propose a technique to tackle the issue.

4.1 Related works

4.1.1 Linguistic feature discretization in voice anonymization

In the VPC 2022, team `T04` proposed the use of a TTS-based VA system, achieving excellent results in terms of privacy protection, at the cost of reduced pitch correlation (see Section 2.3.2). The work on `T04` first highlighted a possibly intuitive, yet important idea: strong anonymization can be achieved by extracting the spoken content of the input utterance to anonymize in the form of text and using it to synthesize a completely new speech waveform with a TTS engine. This effectively erases all speaker-specific information from the original waveform. The downside of this approach is that one could question whether the anonymized signal is even related at all to the input utterance: assuming that VA can be carried out via pure TTS implies that the only relevant information that anonymization should preserve is the spoken content, i.e. the textual information. Then, it could be argued that producing an anonymized waveform is useless, when simply storing the transcription of the input utterance would preserve the same amount of information for a much lower storage cost.

While a VA system purely based on TTS seems too extreme and not useful in practice, it does embody an important idea: before performing voice conversion, the input utterance should be converted to a more coarse-grained representation in order to isolate the essential parts of the signal and remove speaker-specific clues. This concept was further explored in [76], which introduced the foundations of what would become baseline B5 in the 2024 edition of VPC. Using the typical x-vector–based VA pipeline [59] as a starting point, the authors of [76] proposed to sanitize the linguistic features extracted from the input signal via a classic vector quantization scheme [122]. Given a sequence of linguistic feature vectors $\mathbf{H} = (\mathbf{h}_1, \mathbf{h}_2 \dots \mathbf{h}_T)$, each vector $\mathbf{h}_i$ is quantized to another vector $\mathbf{e}_j$ of the same size by a VQ layer; $\mathbf{e}_j$ is chosen from a codebook $E = \left\{\mathbf{e}_1, \mathbf{e}_2 \dots \mathbf{e}_J\right\}$, according to the rule

$$\mathrm{VQ}\left(\mathbf{h}_t\right) = \tilde{\mathbf{h}}_t = \underset{e_j \in E}{\operatorname{argmin}} \|\mathbf{h}_t - \mathbf{e}_j\|_2^2 \tag{4.1}$$

The new, “sanitized” set of linguistic features then becomes $\tilde{\mathbf{H}} = \left(\tilde{\mathbf{h}}_1, \tilde{\mathbf{h}}_2 \dots \tilde{\mathbf{h}}_T\right)$. For each step $\mathbf{h}_t$ of the original sequence, a loss term optimizes its assigned codebook embedding $\mathbf{e}_k$ to become closer to $\mathbf{h}_t$:

$$\mathscr{L}_{vq} = \|sg\left[\mathbf{h}_t\right] - \mathbf{e}_k\|_2^2 \tag{4.2}$$

where $sg\left[\cdot\right]$ is the stop-gradient function, which prevents gradients from flowing into the encoder. In this way, the embedding codebook is learned during training along with the rest of the model parameters.

In [76], two main model configurations were presented. They differ in the way linguistic feature vectors $\mathbf{H}$ are produced. The first configuration uses a TDNN-F [51] trained on LibriSpeech-train-clean-100 and LibriSpeech-train-other-500 with a ASR objective for triphone classification. The second configuration uses a pretrained wav2vec 2.0 feature extractor [80]. The latter achieved the best performance on the VPC 2022 benchmark, scoring 17.5% EER on an informed attack scenario [38] while keeping a moderate WER of 4.5% and a pitch correlation $\boldsymbol{\rho}^{F_0}$ of 0.67.

This work confirmed the potential of audio discretization techniques for VA: mapping continuous feature vectors into a set of codewords via VQ favors the suppression of speaker-specific information from the final anonymized waveform. In the following section, we describe a different, emerging audio processing technique based on the quantization of audio features, which the candidate explored in the context of VA.

Algorithm 1: NAC encoding

```
Input: utterance $\mathbf{u} \in \mathbb{R}^L$
Output: NAC tokens $\mathbf{a} \in \mathbb{N}^{Q \times T}$
$\mathbf{H} \leftarrow$ encoder($\mathbf{u}$) // H has shape $d \times T$
$\mathbf{a} \leftarrow \mathbf{0}$
for $t \leftarrow 1$ to $T$ do
    $\tilde{\mathbf{h}}_t \leftarrow \mathbf{0}$
    $\mathbf{h}_t \leftarrow \mathbf{H}[:, t]$
    residual $\leftarrow \mathbf{h}_t$
    for $q \leftarrow 1$ to $Q$ do
        $\tilde{\mathbf{h}}_t$, idx $\leftarrow$ quantize(residual)
        $\mathbf{a}[q, t] \leftarrow$ idx
        residual $- = \tilde{\mathbf{h}}_t$
return a
```

Algorithm 2: NAC decoding

```
Input: NAC tokens $\mathbf{a} \in \mathbb{N}^{Q \times T}$
Output: utterance $\mathbf{u} \in \mathbb{R}^L$
$\mathbf{H} \leftarrow \mathbf{0}$
for $t \leftarrow 1$ to $T$ do
    for $q \leftarrow 1$ to $Q$ do
        $\mathbf{H}[:, t] += \text{codebook}_q(\mathbf{a}[q, t])$
$\mathbf{u} \leftarrow$ decoder($\mathbf{H}$)
return u
```

4.1.2 Neural audio codecs

A Neural Audio Codec (NAC) is an audio compression algorithm based on deep neural networks. While the first notable work on the topic was SoundStream [123], NACs really began to gain momentum in the speech research community with EnCodec [124], around the same time as the VPC 2022 [27]. Both SoundStream and EnCodec are neural encoder-decoder architectures: the encoder compresses audio input at a low bitrate, while the decoder reconstructs the original waveform with high fidelity. Essential to both SoundStream and EnCodec is the use of vector quantization in the latent space: the embeddings produced by the encoder are quantized using a learned codebook shared between encoder and decoder. This allows the encoder to send integer indices instead of continuous embeddings, reducing the bitrate needed to compress the input audio. To ensure a high reconstruction quality, both SoundStream and Encodec make use of *residual* vector quantization, whereby the initial embedding produced by the encoder is discretized with a hierarchical sequence of quantizers. The first quantizer operates as described in the previous section, assigning an encoded audio frame $\mathbf{h}_t$ to the closest codebook entry $\mathbf{e}_k$. Codebooks from the second onward instead quantize the quantization error of the previous codebooks, referred to as the *residual*. This allows for a more fine-grained discretization of the original embedding and a consequent high-quality reconstruction. The output of the encoder for each timestep is the set of indices produced by all codebook: this way, an audio frame can be encoded as a list of Q integers, where Q is the total number of codebooks, rather than a d-dimensional embedding in floating point format. The encoding and decoding processes are more formally detailed in Algorithm 1 and Algorithm 2 respectively.

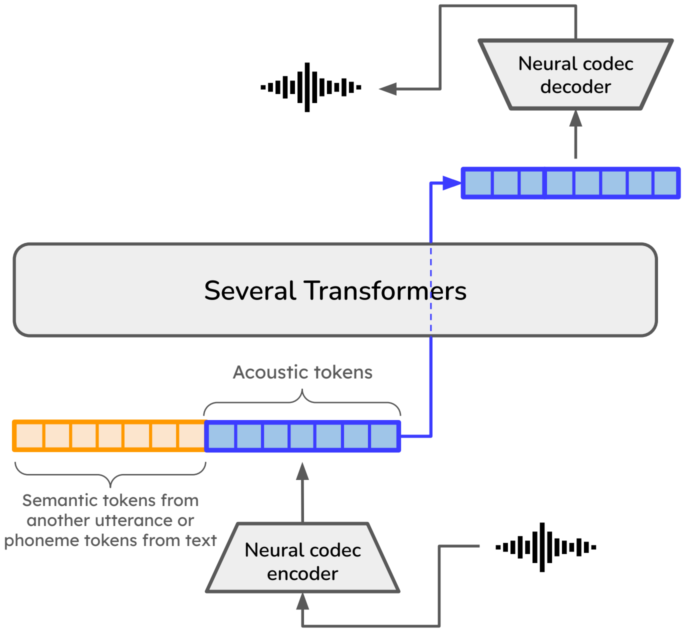


Figure 4.1: High-level overview of the functioning of a speech synthesis system based on NAC language modeling, such as AudioLM or VALL-E.

4.1.3 Neural audio codec language modeling for text-to-speech

The core idea behind NACs is that audio can be compressed to low bitrates by encoding the signal as a sequence of integers. Beyond compression, this approach also enables the application of modern language modeling techniques to audio. In Natural Language Processing (NLP), text is typically represented as discrete tokens, each mapped to an integer and processed by models such as Transformers [125]. Similarly, with NACs, audio can be tokenized into integer sequences and processed in the same way.

The first step in this direction was AudioLM [126], which applied Transformer based autoregressive language modeling to NAC tokens. More specifically, AudioLM consists in a three-stage pipeline:

1. A tokenization stage that converts the input audio into a sequence of tokens;
2. A decoder-only Transformer model [127] trained to maximize the sequence likelihood in an autoregressive way and predicts new tokens autoregressively at inference time;
3. A de-tokenization stage that converts the generated tokens back to waveform.

The model popularized the idea of combining two distinct tokenization schemes: *acoustic* and *semantic* tokenization. The acoustic tokens are NAC tokens generated by SoundStream: they capture fine-grained audio details that allow for high-quality waveform synthesis. Semantic tokens are obtained by clustering the feature representations from an intermediate layer of wav2vec-BERT [128] to create a codebook. These tokens capture long-term structural information, including the speech content of the utterance. The two token sets are concatenated into a single sequence of integers and processed by the decoder-only Transformer autoregressively. At inference time, the Transformer generates new acoustic tokens that are decoded to waveform by the SoundStream decoder. The authors demonstrated that this dual token representation enables AudioLM to, among other tasks, generate a continuation of a 3-second utterance that preserves the original speaker's timbre, suggesting the potential of NAC-based language modeling for zero-shot speech synthesis in a target speaker's voice.

This potential was later harnessed by VALL-E [129], a TTS system based on NAC-based language modeling. Inspired by AudioLM, VALL-E replaced semantic tokens with *phoneme* tokens derived from the input text. Like semantic tokens, phoneme tokens represent the speech content to be synthesized. To capture the target speaker's vocal characteristics, acoustic tokens are extracted from a reference utterance. These two token sets are then used to condition a Transformer-based model that predicts new acoustic tokens corresponding to the input text spoken in the target speaker's voice. A high-level representation of the functioning of AudioLM and VALL-E is shown in Figure 4.1. Further architectural details are deferred to Section 4.2.1.

At the time of writing, neither AudioLM nor VALL-E have been publicly released. Because these systems are based on large Transformer models, re-implementing and training them from scratch would be prohibitively demanding in terms of computational resources. However, similar foundation models have been open-sourced. One such model is Bark,[1] a TTS system proposed by Suno AI, heavily inspired by VALL-E. We leveraged Bark's modules to explore the application of NAC language modeling to VA.

4.2 Voice anonymization with neural audio codec language modeling

In this section, we present a VA approach based on NAC language modeling. As outlined in the previous sections, this investigation was motivated by two main goals: to move beyond the conventional x-vector–based pipeline (see Section 3.7), and to further explore the potential of audio discretization.

[1] The original souce code is available at https://github.com/suno-ai/bark, though we built our system from the port included in the CoquiTTS library available at https://github.com/coqui-ai/TTS. At the time of writing, the original CoquiTTS library is no longer maintained; however, a community-maintained fork can be found at https://github.com/idiap/coqui-ai-TTS.

4.2.1 Proposed approach

In line with the typical structure of a VA system, the proposed model takes as input an audio utterance to anonymize along with a pseudo-speaker identity, and outputs a new anonymized utterance that reflects the vocal characteristics of the pseudo-speaker. The model consists of:

- A **semantic encoder**, which processes the input utterance to produce the *semantic* tokens;
- A **NAC** encoder-decoder couple: the encoder takes as input an utterance of the pseudo-speaker, while the decoder is used to decode the set of generated *acoustic* tokens to waveform;
- A pair of **Transformers** that process the combination of semantic and acoustic tokens;
- A pool of *speaker prompts*.

In the following, we describe them in detail. An illustration of the system can be found in Figure 4.2.

The semantic encoder is a neural network that produces high-level semantic representations of the input signal using a codebook of N_S quantized embeddings. The output is a sequence of integers $\mathbf{s} \in \{1,\ldots,N_S\}^{T_S}$, where T_S is the number of frames and where each integer is a codeword index. We refer to $\mathbf{s}$ as *semantic* tokens. They are assumed to capture the long-term semantical structure of the input speech, i.e. the spoken content.

The NAC is an encoder-decoder architecture that operates as detailed in Section 4.1.2. The encoder maps input waveforms to a quantized, compressed representation from which the decoder reconstructs a high-fidelity waveform. Efficient compression is achieved with a set of Q hierarchical codebooks. Lower level codebooks capture coarser waveform characteristics, while finer details are captured by higher level codebooks. Following [126, 129], we refer to the first Q_C codebooks as “coarse codebooks”, and to the last $Q - Q_C$ codebooks as “fine codebooks”. All have N_A codewords so that the output of the encoder is $\tilde{\mathbf{a}} \in \{1,\ldots,N_A\}^{Q\times T_A}$, where T_A is the number of frames into which the input is divided. We refer to $\tilde{\mathbf{a}}$ as *acoustic* tokens. Within the context of this model, the acoustic tokens are assumed to capture the vocal characteristics of the pseudo-speaker: conceptually, one could say they take on the role of the speaker embedding in x-vector–based architectures. However, unlike a speaker embedding, a different set of acoustic tokens generated by the Transformers is decoded to waveform to obtain the final anonymized utterance.

The coarse and fine transformers estimate a set of acoustic tokens $\mathbf{a}$ from a prompt of input semantic tokens $\mathbf{s}$ and acoustic tokens $\tilde{\mathbf{a}}$. Essentially, the transformers attempt to predict what semantic information contained in $\mathbf{s}$ should ‘sound like’ in the domain of quantized acoustic tokens.

The coarse transformer is a typical "GPT-style" transformer decoder [130]: it autoregressively predicts coarse acoustic tokens, i.e. the codewords belonging to the coarse codebooks. More specifically, for frame t, the transformer predicts the probability distribution of token $\mathbf{a}_{q,t}$ conditioned on the following elements: the semantic prompt $\mathbf{s}$, the coarse tokens from the acoustic prompt $\tilde{\mathbf{a}}_{<Q_C,:}$, and all previous predictions. The modeled distribution is therefore

$$p\left(\mathbf{a}_{q,t} \middle| \mathbf{s}, \tilde{\mathbf{a}}_{<Q_C,:}, \mathbf{a}_{<Q_C,<t}, \mathbf{a}_{<q,t}\right) \tag{4.3}$$

for $q \in [1, Q_C]$. In practice, the sequence upon to which perform regression is flattened to

$$(\mathbf{s}, \underbrace{\tilde{\mathbf{a}}_{1,1} \ldots \tilde{\mathbf{a}}_{2,1} \ldots \tilde{\mathbf{a}}_{Q_C,1} \ldots \tilde{\mathbf{a}}_{1,2} \ldots \tilde{\mathbf{a}}_{Q_C,2} \ldots \tilde{\mathbf{a}}_{Q_C,T_A}}_{\text{Acoustic prompt, flattened column-wise}} \cdots \underbrace{\mathbf{a}_{1,1}, \mathbf{a}_{2,1} \ldots \mathbf{a}_{Q_C,1}}_{\substack{\text{Predicted coarse} \\ \text{codes of time step } t=1}}, \underbrace{\mathbf{a}_{1,2}, \mathbf{a}_{2,2} \ldots \mathbf{a}_{Q_C,2}}_{\substack{\text{Predicted coarse} \\ \text{codes of time step } t=2}} \ldots) \tag{4.4}$$

In other words, the acoustic prompt $\tilde{\mathbf{a}}$ is flattened column-wise and concatenated to the semantic tokens $\mathbf{s}$ as a single sequence, the continuation of which is autoregressively estimated by the coarse Transformer. The token indices are re-mapped to a common embedding dictionary that includes the entirety of the semantic tokens and the acoustic tokens of all codebook layers: thus, the size of the input embedding dictionary of the coarse Transformer is $N_S + (Q_C \times N_A)$. The fine transformer is instead non-autoregressive. It estimates the tokens of codebook q using all tokens belonging to codebooks $< q$ and all tokens of the acoustic prompt $\tilde{\mathbf{a}}$, thus modeling the distribution

$$p\left(\mathbf{a}_{q,:} \middle| \tilde{\mathbf{a}}_{<q,:}, \mathbf{a}_{<q,:}\right) \tag{4.5}$$

for every $q \in [Q_C + 1, Q]$. In practice, the fine Transformer uses Q different embedding layers, one for each level of acoustic tokens; moreover, it has $Q - Q_C$ different classification heads, one for each level of fine acoustic tokens to predict. The generation of the fine tokens of level q proceeds as follows. For each codebook layer $i < q$, the tokens of the acoustic prompt and the already predicted acoustic tokens are concatenated into a single integer sequence $\left(\tilde{\mathbf{a}}_{i,:}, \mathbf{a}_{i,:}\right)$.[2] Embeddings $\mathbf{e}_i \in \mathbb{R}^{d \times (T_A + T_{\mathbf{a}})}$ are extracted from the i^{th} embedding dictionary ($T_{\mathbf{a}}$ being the length of the sequence of the predicted coarse tokens). The input $\tilde{\mathbf{e}}_q$ given to the Transformer is the sum of all embeddings up to layer $q - 1$:

$$\tilde{\mathbf{e}}_q = \sum_{i=1}^{q-1} \mathbf{e}_i \tag{4.6}$$

[2] When q is the first layer of fine acoustic tokens, i.e. when $q = Q_C + 1$, the variable i loops over the indices of coarse tokens.

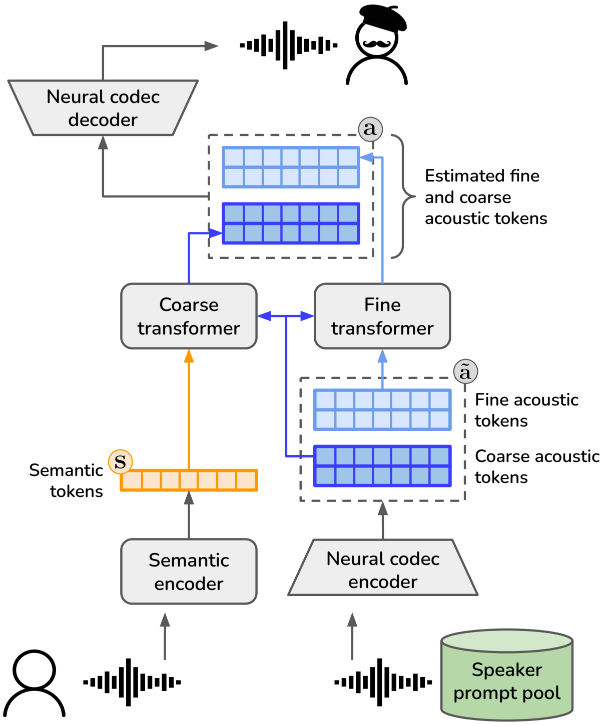


Figure 4.2: Diagram of the proposed VA system.

The sequence is processed non-autoregressively, i.e. the Transformer has visibility to all timesteps when processing each individual timestep t. This allows the prediction of the output for a codebook layer q in a single shot.[3] Once the final hidden states for codebook q have been computed, the q^{th} classification head projects them to a probability distribution over the acoustic tokens for each timestep. Once the acoustic tokens $\mathbf{a}$ have been estimated for all codebooks $q \in [1, Q]$, they can be input into the NAC decoder to synthesize an anonymized waveform.

The pool of speaker prompts is a set of acoustic tokens extracted by the NAC encoder from utterances belonging to a set of external speakers. Those speakers are referred to as *pseudo-speakers*, since they replace the original speaker in the anonymized utterance. As suggested in [126], acoustic tokens, especially the coarse tokens, can capture information related to the speaker identity. We use them to perform VC.

4.2.2 Anonymization technique

A set of semantic tokens $\mathbf{s}$ is first extracted from the input utterance. These tokens encode the high-level spoken content. Their quantization helps to suppress speaker-related information.

[3] In practice, the fine Transformer has a maximum input length L_{max}. If the input sequence exceeds that length, it is processed in sliding windows.

A pseudo-speaker is chosen by randomly selecting an acoustic prompt $\tilde{\mathbf{a}}$ from the speaker prompt pool. Anonymization can be performed at either speaker or utterance levels. At the *speaker level*, anonymization is performed using the same speaker prompt for each utterance corresponding to any one specific speaker. In contrast, for *utterance level* anonymization, a speaker prompt is selected at random for each utterance. While several anonymization systems include techniques to synthesize fictitious voices [65, 76, 77, 110], here we use real voices as pseudo-speakers to focus our analysis on the intrinsic anonymization capability of the NAC language model.

Prompted with $\mathbf{s}$ and $\tilde{\mathbf{a}}$, the coarse and fine transformers generate a set of acoustic tokens $\mathbf{a}$ which reflect the semantic information of the original utterance, but the acoustic characteristics of the pseudo-speaker. Acoustic tokens $\mathbf{a}$ are fed to the NAC decoder which synthesizes the anonymized output waveform.

4.2.3 Implementation details

The coarse and fine Transformers are taken from Bark. They are 12-layer decoder-only models [130] with a hidden embedding size of 1024 and 16 heads in each attention layer. They collectively generate acoustic tokens of $Q = 8$ different codebooks with $N_A = 1024$ codewords. The first $Q_C = 2$ codebooks are considered coarse. $L_{\max} = 1024$. Details about the data used to train the models were not disclosed by Suno AI. The NAC is EnCodec [124], whose values of Q and N_A match those of the coarse and fine Transformers.

The semantic encoder is taken from the CoquiTTS library.[4] Bark does not normally have a semantic encoder; instead, it uses a *textual* Transformer that takes text tokens as input and predicts a set of semantic tokens that encode the given textual information in speech form. The textual Transformer must be conditioned on a prompt of semantic tokens extracted from a target speaker whose speaking style should be mimicked. Suno AI only provided semantic prompts for a limited number of speakers.[5]

To allow the processing of arbitrary voices, contributors of CoquiTTS trained a semantic encoder to enable the generation of semantic prompts from arbitrary speakers. The encoder has a HuBERT backbone [117] and a Long-Short Term Memory (LSTM) network [131] back-end which predicts the semantic token associated to the HuBERT feature vector output at each frame. The semantic dictionary is of size $N_S = 10048$. The semantic encoder was trained as follows. A corpus of 9 public domain books from project Gutemberg[6] was chunked into sentences. Using Bark TTS, those sentences were turned into two different data formats: semantic tokens, using the textual Transformer and pre-defined semantic prompts; and waveforms, using the full Bark pipeline. A

[4] https://github.com/coqui-ai/TTS/tree/main/TTS/tts/layers/bark/hubert

[5] https://github.com/suno-ai/bark/tree/main/bark/assets/prompts/v2

[6] https://www.gutenberg.org/

Table 4.1: Results of the analyzed systems on the 2022 VPC test subsets.

System	*LibriSpeech*				*VCTK*			
	EER (%) ↑	WER (%) ↓	G_{VD} ↑	$\boldsymbol{\rho}^{F_0}$ ↑	EER (%) ↑	WER (%) ↓	G_{VD} ↑	$\boldsymbol{\rho}^{F_0}$ ↑
Original	4.4	4.2	0	1	3.2	12.8	0	1
Original (eval. pipeline of [132])	1.5	2.5			1.1	7.6		
B1.b [42]	8.6	4.4	-5.8	0.78	9.7	10.7	-7.1	0.81
T11 [110]	20.6	3.9	-19.0	0.68	39.7	7.9	-18.4	0.73
Ours	28.5	7.5	-1.5	0.68	45.5	18.9	-2.1	0.74
Ours (eval. pipeline of [132])	34.1	4.6	n.a.		36.6	15.5	n.a.	

new semantic encoder was then trained to learn the mapping from waveform to semantic tokens. The HuBERT backbone was kept frozen, while the LSTM back-end was trained to perform the discrete prediction of the token associated to each time frame with a cross-entropy loss.

While the intended function of the semantic encoder was to allow the usage of arbitrary target speakers in Bark TTS, we used it to turn the system into a VC system. We remove the textual Transformer and adapt the semantic encoder to produce ground truth semantic tokens $\mathbf{s}$ of the source utterance, as illustrated in the previous section. The rest of the Bark pipeline can then handle these tokens as if they were generated from text; when paired with an acoustic prompt $\tilde{\mathbf{a}}$ extracted from a target speaker, the system can effectively perform VC.

4.2.4 Experimental setup and results on VPC 2022 protocol

We adopt the VPC 2022 protocol [27] for evaluation (see Section 2.4). The test set comprises subsets of the LibriSpeech [46] and VCTK [48] databases. The pool of speaker prompts is taken from the Bark voice library. It consists of 130 utterances collected from speakers of different gender and nationality.[5] The threat model is the *semi-informed* attack described in [27]. See Sections 2.1.2 and 2.3.1 for further details.

We adopt the B1.b and T11 participant system from the VPC 2022 as baselines. System T11 is the non-TTS system that achieved the highest privacy level in the 2022 challenge [42].[7] Results are shown separately for LibriSpeech and VCTK test sets in Table 4.1. Our system achieves the highest privacy levels: 28.5% EER for LibriSpeech; 45.5% EER for VCTK, both substantially higher than B1.b. We also outperform T11 in terms of privacy protection by 8% and 5% EER for LibriSpeech and VCTK test sets respectively. With a $G_{\mathrm{VD}} \approx -2$, our systems also offers better voice distinctiveness than both baseline B1.b and system T11. Note that the gain in voice distinctiveness for T11 system are very low, meaning that the system maps all speakers to similar pseudo-speakers.

[7] The overall highest privacy level was in fact achieved by a TTS-based system [65] that barely passed the prosody preservation requirement of scoring $\boldsymbol{\rho}^{F_0} > 0.3$. In general, TTS-based systems are known to almost completely erase speaker information, at the cost of a severe loss of intonation and prosody. Therefore, we do not include [65] in our comparative analysis.

However, utility estimates for our model are lower than that of other systems. The WERs increase from 4.2% (original data) to 7.5% for the LibriSpeech subset and from 12.8% to 18.9% for the VCTK subset. Similar issues have also been reported in the literature. The authors of [126] show that the NAC copy-synthesis of LibriSpeech test-clean dataset causes an increase in the WER of its own ASR system from 2.5% to 6%, with similar results being reported in [129]. Nevertheless, informal listening tests on our data do not reveal any notable artifacts or degradation to intelligibility.

In an attempt to shed light on the cause for this phenomenon, we repeated similar experiments using a different ASR architecture, namely that reported in [132], which is retrained according to the same setup described in Section 2.3.1. The issue persists. The WER increases from 2.5% to 4.6% for the LibriSpeech subset and from 7.6% to 15.5% for the VCTK subset. These findings suggest that the degradation to utility is more dependent on the NAC language model than on the ASR system. As suggested in [126], this could be due to the quality of some pseudo-speaker prompts, since the extracted fine acoustic tokens tend also to capture aspects of the (potentially poor) *recording conditions*, the characteristics of which are then transferred to anonymized outputs.

4.2.5 Results for VPC 2024 protocol

Following our proposal of NAC-based anonymization in [87], the described system was selected as one of the baselines for the 2024 edition of the VPC, where it was referred to as baseline B4. Its results, compared with those of the other challenge baselines, have already been presented in Table 2.5 of Section 2.4.5. In what follows, we revise these results with a specific focus on B4.

Within the VA community, B4 was informally considered among the "top 3" baselines of the VPC 2024, together with B5 and B3. This perception stemmed from their strong performance, as they achieved the highest EER values among the baselines: approximately 26%, 31%, and 34% for B3, B4, and B5, respectively. Interestingly, each relied on a different underlying principle: B3 on TTS, B4 on NAC modeling, and B5 on VQ combined with a more traditional VC approach. This diversity contributed to a particularly heterogeneous set of competitive baselines for the 2024 VPC.

Among the three systems, B4 can be regarded as a "middle ground": it provides slightly lower privacy protection than the strongest system (B5), while offering marginally better emotion preservation (~ 42% UAR compared to ~ 38% for B5). Consistent with the VPC 2022 evaluation setting, B4 also shows a higher WER than both B3 and B5, though still lower than B2 and B6. In the 2024 setup, however, this gap is less pronounced — likely because the ASR system was not adapted to the anonymized domain and was instead used off-the-shelf. As a result, the differences in effectiveness across anonymization systems are reduced.

In any case, we sought to investigate why B4 slightly underperforms in terms of objective intelligibility and to explore a possible solution. The findings of this investigation are presented in the next section.

4.3 Preserving spoken content with character-level vocoder conditioning

In this section, we describe a technique designed to improve the speech content preservation capabilities of B4, which we proposed in [133]. It is based on the idea of *re-injecting* the textual information present in the original signal into the vocoder during the synthesis process, and training the vocoder with explicit supervision to preserve that information within the final anonymized waveform. While we test it on B4, the technique is sufficiently generic that it could also be applied to the neural vocoder model of any comparable voice anonymization system.

Prior work has focused almost exclusively on techniques to better sanitize the voice characteristics of the original speaker. Even if the preservation of other attributes is still measured, little-to-no work in the literature has focused on the design of techniques to *expressly* preserve them. This approach has a fundamentally different emphasis, casting the problem of privacy preservation as that of preserving *desired* attributes, rather than or in addition to sanitizing the *undesired* attributes. Privacy is, after all, preserved perfectly if the speech content in a recording is entirely suppressed, even if the desired attributes needed to accomplish some downstream task are also entirely lost. The task is then more to retain only those attributes that are necessary in order to accomplish the downstream task, while sanitizing *everything* else. We take a step in this direction. With the proposed modification, B4 still aims to substitute the voice of an original speaker with that of a pseudo-speaker, but also to provide better spoken content preservation.

Results show that the reported technique is successful in better preserving spoken content, to the point that it outperforms all other baseline systems in terms of the WER, albeit with a modest reduction in voice anonymization performance. While not a design goal, results show that emotion cues are also better preserved.

4.3.1 Previous approaches to content preservation

To the best of our knowledge, few modeling approaches for voice anonymization were designed to explicitly minimize the WER metric. The closest work in the literature is [134]. A CTC loss [81] is used during training to encourage preservation of the spoken content. However, the evaluation method does not align with that of the VPC, most notably in terms of the attack model. The privacy adversary is uninformed and so the ASV system is not retrained using anonymized data. As previously noted, this approach is known to result in a substantially weaker attack, and hence gives a potentially unreliable assessment of anonymization performance. Other key differences relate to the use of an additional loss term during training, whereas our approach involves a more explicit inductive bias in the model architecture to better preserve spoken content.

Other works have explored a different approach whereby an anonymized speech waveform is synthesized anew from the transcription of the original utterance, therefore erasing the original voice characteristics, and by injecting certain desired attributes into the generated output. The most notable example is baseline B3 of the 2024 VPC [83], which re-synthesizes the signal to anonymize with a TTS model conditioned on the pitch and energy values estimated from the input utterance (see Section 2.4.3 for more details). This results in a more faithful reproduction of the prosody at the cost of a lower EER. In the following, we propose a hybrid approach that still relies on VC applied to the input utterance, therefore preserving prosody and paralinguistic attributes, while injecting spoken content information back into the generated output during vocoding. This encourages the preservation of spoken content with a lower sacrifice in the EER.

As we illustrated in Section 2.3.4, other work reported in the context of the 2022 VPC, including two baselines [42, 65, 77], seemingly already shows that spoken content can be preserved to the point that the WER even *improves*. Decreases in the WER for ASV^{VA} might be interpreted to mean that anonymization enhances intelligibility of the spoken content, even without specific optimizations to do so. This is not the correct interpretation. Decreases in the WER are instead attributed to the fact that, in the VPC 2022 evaluation pipeline, the ASV^{VA} model is retrained using a set of anonymized data, when the anonymization system itself is trained using a database far larger than that used to train the initial ASR model. This additional data produces a model which is strongly adapted to the anonymised domain. In any case, WERs estimated using different ASR systems *and* different evaluation data are not comparable. WER estimates made using the same ASR model are comparable and reveal *increases* in the WER. For the 2024 challenge, the ASR system is trained only once, and using unprotected data only [37]. Under this scenario, anonymization *always* results in a higher WER. We propose a technique to reduce the gap in the WER estimated from unprotected and anonymized data and report its application to system B4.

4.3.2 Character-level vocoder conditioning

The approach is illustrated in Figure 4.3 and is built around the NAC-based approach to voice anonymization described above.

Description of the conditioning method

Informal listening tests reveal that utterances anonymized using the NAC-based approach to anonymization contain occasional mispronunciations. For example, we found cases in which the word ‘snack’ was altered to ‘stack’, or ‘thick’ was altered to ‘flick’. We assume that

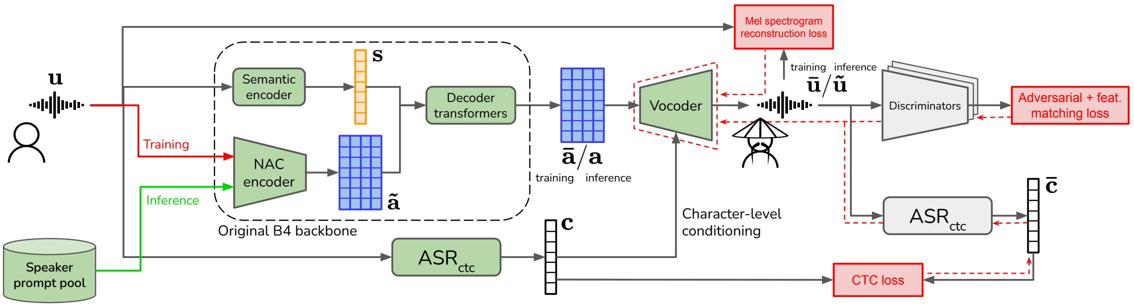


Figure 4.3: Overview of the training and inference procedures. Red dashed lines indicate the gradient flow. At training time, the system operates in a copy-synthesis fashion, extracting the pseudo-speaker identity from the original input utterance $\mathbf{u}_i$. At inference time, only the green blocks are kept, and the pseudo-speaker is selected at random from the pool of pseudo-speakers as in Section 4.2.2.

mispronunciations are caused by the vocoder and that they might be tackled by training the vocoder to use time-aligned, character-level annotations in addition to acoustic tokens $\mathbf{a}$. To avoid mispronunciation errors, the vocoder can be conditioned to use learnable time-aligned, frame-level embeddings which represent the character spoken in the input utterance. Ideally, embeddings should reflect exclusively spoken content. So as to avoid privacy-leakage, the voice characteristics should be contained exclusively in the acoustic tokens $\mathbf{a}$.

The vocoder model is conditioned at every layer with the output of an auxiliary CTC-based ASR system. We use the Vocos [135] vocoder architecture, which generates a waveform by processing a set of NAC acoustic tokens with a chain of *ConvNeXt* blocks [136] which keep the shape of the intermediate features fixed to $D \times T$ for every layer, where D is the channel dimension and T is the number of time steps. This property eases the application of a conditioning mechanism to intermediate vocoder layers. The last layer of the vocoder produces a set of frame-level features which are used by a fully-connected layer to estimate the complex Short-time Fourier Transform (STFT) of each frame. A waveform is then synthesized using the inverse STFT. We employ the version of Vocos designed to operate upon acoustic tokens generated using EnCodec [124]. Since the acoustic tokens $\tilde{\mathbf{a}}$ and hence also $\mathbf{a}$ are in the same EnCodec token format, substitution of the NAC decoder (vocoder) with Vocos is straightforward.

We adjust the vocoder architecture to facilitate its conditioning on character-level annotations extracted from the input $\mathbf{u}$. Between each Vocos ConvNeXt layer, we introduce a *character conditioning* layer which we now describe. As illustrated to the lower left of Figure 4.3, we use a pretrained CTC-based [81] ASR model denoted ASR_{ctc} to extract a character sequence

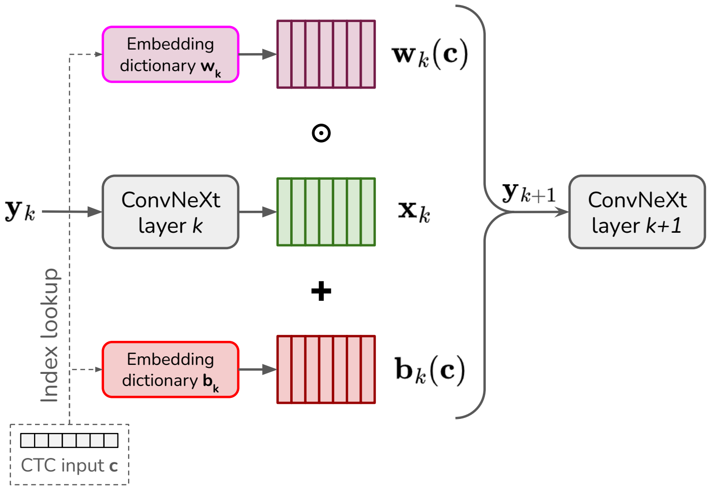

Embedding dictionary $\mathbf{w}_k$
$\mathbf{w}_k(\mathbf{c})$
⊙
$\mathbf{y}_k$
ConvNeXt layer k
$\mathbf{x}_k$
$\mathbf{y}_{k+1}$
ConvNeXt layer $k+1$
+
Index lookup
Embedding dictionary $\mathbf{b}_k$
$\mathbf{b}_k(\mathbf{c})$
CTC input $\mathbf{c}$

Figure 4.4: Character conditioning layer k. Starting from an array of character indexes $\mathbf{c}$, two matrices $\mathbf{w}_k(\mathbf{c})$ and $\mathbf{b}_k(\mathbf{c})$ are constructed from embedding dictionaries $\mathbf{w}_k$ and $\mathbf{b}_k$. The intermediate vocoder representation $\mathbf{x}_k$ is pointwise-multiplied by $\mathbf{w}_k(\mathbf{c})$ then summed to $\mathbf{b}_k(\mathbf{c})$. The result $\mathbf{y}_{k+1}$ is passed to the next layer.

$\mathbf{c} \in \{0, 1 \ldots 30\}^T$ from the input $\mathbf{u}$. Each element of $\mathbf{c}$ is an index associated to a set of CTC characters including the 26 letters of the English alphabet, the white space character, the apostrophe character, the CTC *null* token, and the *beginning-of-sentence* and *end-of-sentence* tokens.[8]

As illustrated in Figure 4.4, the k-th character conditioning layer (placed after the k-th ConvNeXt layer) takes $\mathbf{c}$ as input and contains two dictionaries of learnable embeddings both of dimension D, denoted $\mathbf{w}_k$ and $\mathbf{b}_k$. Each dictionary has 31 entries, one for each CTC character. We denote by $\mathbf{w}_k(\mathbf{c}) \in \mathbb{R}^{D\times T}$ the matrix of embeddings constructed by concatenating entries from $\mathbf{w}_k$ according to the indexes in $\mathbf{c}$. Likewise, $\mathbf{b}_k(\mathbf{c}) \in \mathbb{R}^{D\times T}$ refers to the same operation applied to dictionary $\mathbf{b}_k$. Inspired by FiLM [137], and given the output of the k-th ConvNeXt layer $\mathbf{x}_k$, we condition $\mathbf{x}_k$ with the affine transformation

$$\mathbf{y}_{k+1} = \mathbf{w}_k(\mathbf{c}) \odot \mathbf{x}_k + \mathbf{b}_k(\mathbf{c}) \tag{4.7}$$

where ⊙ represents element-wise multiplication. Each intermediate feature matrix in the vocoder is conditioned upon the time-aligned input character. Output $\mathbf{y}_{k+1}$ is then used as the input to the following $(k+1)$-th ConvNeXt layer. Note that, while the vocoder is conditioned by $\mathbf{c}$, it is still driven by the acoustic tokens $\mathbf{a}$, meaning that the approach is still closer to VC than TTS.

[8] The length of $\mathbf{c}$ could be different to T because of a setting mismatch between ASR$_{\text{ctc}}$ and the anonymisation system (e.g. different hop size). If so, $\mathbf{c}$ is resized to have length T using nearest-neighbour interpolation.

Vocoder training

The character embeddings in $\mathbf{w}_k$ and $\mathbf{b}_k$ are optimized jointly with the vocoder. The goal is to ensure high quality waveform synthesis and to preserve the spoken content found within the input utterance $\mathbf{u}$. The first goal is addressed using the training loss for the original Vocos vocoder. It comprises three components: a mel-spectrogram reconstruction loss (ℓ_{mel}); an adversarial loss against a set of multi-period and multi-resolution discriminators [63, 138] (ℓ_{gan}) in the style of aGAN [74]; a feature-matching loss between the discriminators (ℓ_{fm}). These are represented by the two upper-most, red-colored boxes in Figure 4.3. To encourage the preservation of spoken content, we add a new CTC loss component ℓ_{ctc} shown in the lower red box. The training dataset is composed of triplets $(\mathbf{u}_i, \mathbf{c}_i, \bar{\mathbf{a}}_i)$, where $\mathbf{u}_i$ is an unprotected utterance, $\mathbf{c}_i = \text{ASR}_{\text{ctc}}(\mathbf{u}_i)$ is the sequence of CTC symbols extracted from $\mathbf{u}_i$ using ASR_{ctc} and $\bar{\mathbf{a}}_i$ is the set of acoustic tokens produced by the pair of transformers in the special case where the speaker prompt is that of the original speaker (no anonymization).

During training, the vocoder receives as input $\bar{\mathbf{a}}_i$ (for copy-synthesis in the usual way) and $\mathbf{c}_i$ (for character-level conditioning) and generates the waveform $\bar{\mathbf{u}}_i$. The CTC symbol sequence $\bar{\mathbf{c}}_i = \text{ASR}_{\text{ctc}}(\bar{\mathbf{u}}_i)$ is then inferred. The reconstruction objectives ℓ_{mel}, ℓ_{gan} and ℓ_{fm} are computed by comparing $\mathbf{u}_i$ and $\bar{\mathbf{u}}_i$, while ℓ_{ctc} is computed between $\mathbf{c}_i$ and $\bar{\mathbf{c}}_i$. The generator training loss is then

$$\ell_{\text{gen}} = \lambda_{\text{mel}}\ell_{\text{mel}} + \lambda_{\text{gan}}\ell_{\text{gan}} + \lambda_{\text{fm}}\ell_{\text{fm}} + \lambda_{\text{ctc}}\ell_{\text{ctc}} \tag{4.8}$$

where hyperparameters λ are weights assigned to each term. The discriminator loss is the same as in [135]. The generator (vocoder) and the discriminators are optimized with AdamW [139].

4.3.3 Experimental setup

The vocoder weights are initialized to those of the Vocos checkpoint trained using EnCodec 24 kHz tokens.[9] Both character embedding dictionaries $\mathbf{w}_k$ and $\mathbf{b}_k$ are initialized to $\mathbf{1}$ and $\mathbf{0}$ respectively, with the addition of a small zero-centered gaussian noise, so that Eq. (4.7) is almost equivalent to an identity operation. The weights of the generator loss are $\lambda_{\text{mel}} = \lambda_{\text{fm}} = 1$, $\lambda_{\text{gan}} = 0.5$, $\lambda_{\text{ctc}} = 1.5$. The full system is trained for 300k steps with an initial learning rate of 5e-4 and with a single-cycle cosine annealing decreasing to 0. Since the generator (the vocoder) weights are initialized to those of a pretrained checkpoint, we do the same for the discriminator since it was found to increase adversarial training stability. This is done by fixing the weights of the pretrained generator while optimizing the discriminators for 6000 steps with an initial learning rate of 1e-3 and cosine annealing. Both the generator and the discriminators are trained

[9] https://huggingface.co/charactr/vocos-encodec-24khz

Table 4.2: Results on the evaluation sets of the VPC 2024 of the strongest baselines in terms of privacy protection (B3, B4, B5), the one with the best speech content preservation (B1), and our proposed system (highlighted in gray). The NAC system with character-level conditioning achieves the best results in terms of spoken content preservation (WER) and second best emotion preservation (UAR) while also maintaining a strong privacy protection level (EER).

Dataset	**Unprotected**	**B1**	**B3**	**B4** **(original NAC)**	**B5**	**NAC + char. conditioning**
EER Libri-dev (%)	5.72	9.20	25.24	32.71	34.37	30.76
EER Libri-test (%)	4.59	6.07	27.32	30.26	34.34	28.69
WER Libri-dev (%)	1.80	3.07	4.29	6.15	4.74	2.66
WER Libri-test (%)	1.85	2.91	4.35	5.90	4.37	2.54
UAR IEMOCAP-dev (%)	69.08	42.71	38.09	41.97	38.08	44.91
UAR IEMOCAP-test (%)	71.06	42.78	37.57	42.78	38.17	43.59

using the *LibriSpeech-train-clean-100* [46] database. The ASR$_{\text{ctc}}$ system used to extract $\mathbf{c}_i$ and $\bar{\mathbf{c}}_i$ is a pretrained SpeechBrain [79] model[10] and is a wav2vec 2.0 backbone with a classification head, both jointly fine-tuned using the LibriSpeech database.

Evaluation is performed using the 2024 VPC protocol [37]. Anonymization is performed at the *utterance level* using the voice of a pseudo-speaker selected at random from the speaker prompt pool.[11] We use a semi-informed attacker scenario: as usual, ASV^{VA} is an ECAPA-TDNN model [73] trained using the *LibriSpeech-train-clean-360* database similarly anonymised at the utterance level. The ASR and SER models used for evaluation are pretrained using the original (unprotected) *LibriSpeech-train-960* and *IEMOCAP* [78] databases, respectively (see Section 2.4.2). EER, WER and UAR metrics are computed from anonymized data using the 2024 VPC evaluation pipeline.

4.3.4 Results

Results for the evaluation partition of the 2024 VPC database are reported in Table 4.2. Also shown are results for the relevant challenge baselines, reproduced from Section 2.4.5: B1, the x-vector–based anonymization baseline for which the WER is the lowest among competing baselines [37]; B3, which provides moderately high EER and good WER [83]; B4, the original NAC-based anonymization system described in Section 4.2.1, for which both the EER and UAR are more competitive than B3, but for which the WER is relatively high; B5, the VQ-based system that achieves the highest EER [38].

[10] https://huggingface.co/speechbrain/asr-wav2vec2-librispeech

[11] Inference code is available at https://github.com/eurecom-asp/spk_anon_nac_lm/tree/char_cond_nac.

With a WER of ~2.5%, the proposed system outperforms all competing baselines in terms of preserving speech content, a relative improvement of almost 60% over the original B4 system. As an additional benefit, character-level conditioning also brings modest improvements to emotion preservation. UARs of 45% and 44% are also higher than those for all competing baselines. This improvement might be due to the encoding of some implicit prosody information in **c**. The information contained in **c** is not exclusively 'textual'. Since **c** associates a character to every time step, it can also encode nuances of the speaking rate which are pertinent to the SER task.

The benefits come at the expense of a modest loss in anonymization performance. The EER falls from ~31% for the original B4 baseline to ~29% with character conditioning. The reason is likely the same as for the improvement in SER: nuances of the speaking rate are also pertinent to the ASV task. Nonetheless, anonymization performance of the character-conditioned system is still competitive. Moreover, the relative decrease in the EER of 6% might be a small price to pay given the substantially greater relative improvement in the WER of 56%.

4.4 Conclusions

In this chapter, we presented a novel approach to speaker anonymization based on NAC language modeling. Our system performs voice conversion by extracting a set of semantic tokens from an input signal and using them to estimate a set of acoustic tokens belonging to a different speaker, which in turn are used to synthesize an anonymized speech signal with a NAC decoder. The quantized nature of the semantic and acoustic tokens successfully bottlenecks speaker-related information delivering strong anonymization performance for both the 2022 and 2024 versions of the VPC benchmark. The system demonstrates effective preservation of prosody, speaker distinctiveness, and emotional content. However, while informal listening tests suggest that the anonymized signals remain high in quality and intelligibility, automatic transcription with a speech recognition system revealed a modest reduction in utility, most notably in the 2022 benchmark.

We addressed this reduction by designing a conditioning mechanism for the vocoder that allows the model to better preserve spoken content information. An auxiliary CTC based ASR model is used to extract a symbol sequence from the unprotected utterance. This is used to condition the vocoder module of the anonymization system via a set of learnable embedding dictionaries in order to encourage the preservation of spoken content. Relative to the baseline approach, and for only a modest cost in anonymization performance, the technique is successful in decreasing the word error rate computed from anonymized utterances by almost 60%. The resulting system outperforms all six of the 2024 VPC baselines in terms of preserving spoken content. So long as gradients can be backpropagated, alternative approaches to the extraction of symbol sequences with appropriate temporal resolution could also be used. The technique could equally be applied to other approaches to anonymization which employ a neural-based vocoder model for waveform synthesis.

The work about NAC-based speaker anonymization could be considered part of the "second generation" of VA models that were developed around and along with the 2024 edition of the VPC. Previous research had largely focused on x-vector–based pipelines and target selection strategies. Anonymization was attempted by careful embedding manipulation, placing considerable emphasis on pseudo-speaker identity. The VPC 2024 brought along a new approach: generally speaking, VA was now framed more as a *'downsampling and upsampling'* problem.

The 'downsampling' of the input signal is the step where the speaker-specific cues are purged: it corresponds to the VQ layer of baseline B5, systems T14 and T19-3 of the 2024 VPC (see Section 2.4.4); the semantic token extraction of B4; the first residual quantizer with speaker embedding subtraction of T10 (Section 2.4.4); the phoneme-level transcription of B3; the textual features extraction of T30 (Section 2.4.4), and so on. In other words, there should be a step where the signal is reduced to a lower-dimensional form that contains the speech content and additional attributes for the downstream task of interest.

The 'upsampling' is the consequent synthesis of the anonymized signal. This is where any possible conditioning usually takes place, usually on some pseudo-speaker representation: e.g, the speaker embedding used in B3, all variants of T38, and many other systems (see Section 2.4.4); the acoustic tokens in B4; the one-hot representation in B5. Optionally, some additional information that aims to preserve specific attributes can also condition the synthesis process, like the character-level conditioning added to B4, the emotion embedding of T09, or the non-content embedding of T25-1,2 (Section 2.4.4).

One could argue that, of these two steps, the 'downsampling' primarily determines the level of privacy protection that a VA system can provide. However, in VA — as in any other privacy-related task — the level of security of a system is only ever as strong as the attacker is [140]. For this reason, to thoroughly examine the entire VA pipeline, the next chapter of this dissertation focuses on the role of the attacker in evaluating a VA system.

Chapter 5

The role of the attacker

Throughout this dissertation, the level of privacy protection provided by a VA system is assessed using the EER of an ASV model adapted to the domain of anonymized data. As a result, when evaluating anonymization performance, the attacker is as important as the defender. In this chapter, we take a closer look at this perspective. We begin by addressing the issue of *privacy overestimation* [54], namely the situation in which part of the measured EER is attributable not to the anonymization system itself, but to ineffective training of the attacking ASV model. We then introduce a technique for watermarking unprotected data that may be exploited by an attacker, with the goal of reducing the effectiveness of the adversarial ASV system. This constitutes a novel form of proactive defense that can be layered on top of an anonymization system to further enhance privacy guarantees by actively mitigating attacks.

5.1 The risks and detection of overestimated privacy protection

Estimates of privacy protection reflect both that of the anonymization system as well as the ASV system used for evaluation. Performance can be overestimated if the ASV system performs sub-optimally for reasons other than the anonymization itself. For example, previous work [38] showed that some system design choices might prevent the ASV system from learning discriminative cues from anonymized data, resulting in exaggerated estimates of anonymization performance. They are due not to strong anonymization, but rather to the use of a suboptimal ASV attack model. The inadvertent overestimation of performance can be challenging to avoid; contrary to the design of reliable ASV systems, strong anonymization calls for *poor* ASV performance. Poor performance can be achieved relatively easily, perhaps due to inadequate training choices or even design oversight. As a result, and other than to protect scientific integrity, there is comparatively less incentive to optimize an ASV model used for the evaluation of anonymization systems. This raises challenges in benchmarking; results depend on the degree to which each ASV attack model is optimized with respect to the anonymization system under evaluation.

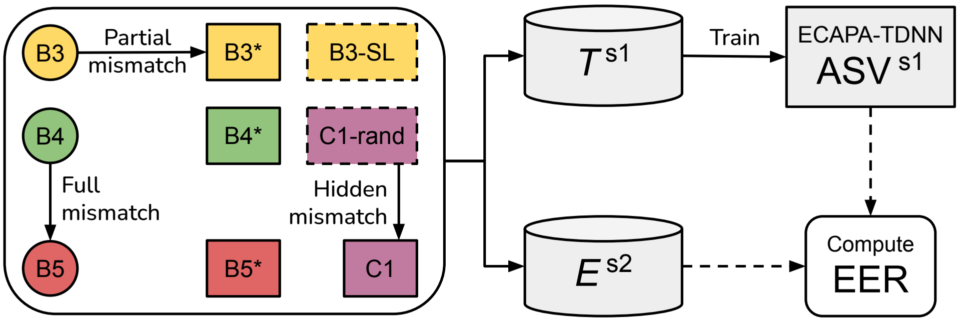


Figure 5.1: Illustration of the mismatched evaluation procedure (best viewed in color). s_1 and s_2 are voice anonymization systems from P. Matched: $s_1 = s_2$. Full mismatch: $s_1 \neq s_2$, different colors. Partial mismatch: $s_1 \neq s_2$, same color, different shapes. Hidden mismatch: $s_1 \neq s_2$, same color, different borders.

We report an investigation of potential evaluation pitfalls and propose what is, to the best of our knowledge, the first solution for their detection.[1] We show that exaggerated privacy protection estimates can result from a mismatch between the distributions of anonymized test data and that used to train the ASV attack model. First, we prove it for the most extreme scenario where the training and evaluation datasets are anonymized by different anonymization systems. Second, we show how a data mismatch and consequent privacy misevaluation can happen even when only a single block of the system is changed between the generation of training and evaluation data. We give examples from the anonymization literature in which this has happened, proving the relevance of our arguments. Once the distribution mismatch is addressed, we show that levels of anonymization performance can, in some cases, be much lower than those reported. Finally, we show that the proposed approach to detection is effective in protecting against untrustworthy performance estimates.

We report an evaluation of anonymization performance for a set of scenarios involving some form of mismatch between the anonymization system under test and that used to generate ASV training data. In this section we describe the set of systems used in our experiments. They include 3 VPC 2024 baseline systems: B3, B4 and B5. For reasons which will become apparent later, we also used one more system based on Self-Supervised Learning (SSL) and Orthogonal Householder Neural Networks (OHNNs), proposed by Miao et al. in [77]. The SSL-OHNN system, henceforth referred to as C1, uses a HuBERT-based soft content encoder [117] to extract the spoken content and an ECAPA-TDNN model [73] to extract a speaker embedding from the original utterance. Anonymization of the speaker embedding is then performed using an OHNN while voice conversion is performed directly using a HiFi-GAN vocoder [63].

[1] Results throughout Section 5.1 are reproduced from [54], a collaborative work led by the candidate, with some results contributed by co-authors.

Table 5.1: Results for full mismatch scenarios (EER, %). The diagonal grayed-out cells correspond to the matched condition (reproduced from Table 2.5), the remaining cells to the full mismatch.

Attacker	Evaluation Data		
	E^{B3}	E^{B4}	E^{B5}
$\mathrm{ASV}^{\mathrm{B3}}$	27.05	44.23	44.40
$\mathrm{ASV}^{\mathrm{B4}}$	36.99	29.82	41.67
$\mathrm{ASV}^{\mathrm{B5}}$	37.28	38.72	34.34

Table 5.2: Results for the partial mismatch scenario. The grayed-out cells correspond to the matched condition (reproduced from Table 2.5).

Evaluation data	Attacker	EER (%)
E^{B3}	$\mathrm{ASV}^{\mathrm{B3}}$	27.05
	$\mathrm{ASV}^{\mathrm{B3*}}$	28.11
E^{B4}	$\mathrm{ASV}^{\mathrm{B4}}$	29.82
	$\mathrm{ASV}^{\mathrm{B4*}}$	35.83
E^{B5}	$\mathrm{ASV}^{\mathrm{B5}}$	34.34
	$\mathrm{ASV}^{\mathrm{B5*}}$	43.71

We evaluate anonymization performance according to the usual VPC 2024 policy [37, 141] (see Section 2.4) and as shown in Figure 5.1. The evaluation of system S is performed using two sets of anonymised LibriSpeech data [46]: (a) the LibriSpeech test data (E^{S}), including both enrollment data (A^{S}) and trials (D^{S}); (b) the LibriSpeech *train-clean-360* set of data [46] which is used by the attacker to train an ASV system using similarly anonymized data. For each anonymisation system S, the attacker trains a new ASV model $\mathrm{ASV}^{\mathrm{S}}$ using anonymized training data T^{S}. A data partition corresponding to 10% of T^{S} is set aside and used for validation during training. As usual, $\mathrm{ASV}^{\mathrm{S}}$ is an ECAPA-TDNN model [73], and the attacker is semi-informed.

5.1.1 Mismatch of anonymized data

We first show that differences between the anonymization systems used to generate anonymized evaluation data E^{S} and that used by the adversary for ASV training T^{S} can lead to privacy protection overestimation. We then show similar findings when the two anonymization systems differ only in terms of a single module. Next, we show that estimates of performance can be lower still if the attacker instead trains the ASV system in a scenario subtly different to the usual semi-informed attack. In all of these cases, there is a *mismatch* between the two anonymization systems. Last, we propose a technique to detect such mismatches with a view to mitigating the overestimation of anonymization performance.

Full mismatch

Under *full mismatch* conditions, the two anonymization systems are completely different. As illustrated in Figure 5.1, system s_1 is used by the attacker to generate ASV training data T^{s_1}. System s_2 is used to produce anonymized evaluation data E^{s_2}. Our goal is to show that a complete domain mismatch between T^{S} and E^{S} can make the training of the attacking ASV suboptimal.

While that might sound intuitively true, no previous work in the literature has provided evidence of it. Verifying this assumption is the foundation of our subsequent experiments. We performed experiments using all possible pairings among systems B3, B4 and B5.

Results are presented in Table 5.1. Each row reflects performance for ASV^{s_1} trained using T^{s_1}, while columns correspond to evaluation data E^{s_2}. Shown in bold face along the diagonal are results for matched conditions where $s_1 = s_2$ — directly reproduced from Table 2.5 of Section 2.4.3. This is the usual *semi-informed* attack scenario described in Section 2.1.2. Off-diagonal results correspond to the conditions of interest where $s_1 \neq s_2$. Note that the evaluation data E^{s_2} includes both the enrollment data A^{s_2} and the trial data D^{s_2}. This reflects the case of interest of this analysis, where the system designer erroneously anonymizes *all* evaluation data with mismatched system s_2. Conversely, in the case where a real attacker would have a mismatched system, the enrollment data A would have to be anonymized with s_1 — the same as the training data, since the attacker handles both A and T. However, our goal is to study the cases of privacy *overestimation* from the standpoint of a system designer.

Our results show that, when compared to the matched condition, the impact of mismatched anonymization systems can be substantial. An EER of 27% is produced when anonymized evaluation data and ASV training data are both generated using the same B3 system. However, the EER increases to 44% (near random guess) when the ASV system is trained using data generated using B4 or B5. Results are similar for B4 for which the EER increases from 30% (matched) to 42% (B5). We note that systems with a higher *matched* privacy level on the diagonal tend to have a less pronounced overestimation of the *mismatched* privacy level outside the diagonal: for example, the EER of B3 increases by 17 percentage points in a mismatched scenario, while B5's only increases by around 4 — less pronounced, yet not insignificant. While the above is an extreme, unrealistic example, we show next the potential for similar performance overestimates even when the differences between each anonymization system is more subtle, including real examples from the literature.

Partial mismatch

We evaluate the performance of an anonymization system under a more realistic, *partial mismatch* scenario in which the anonymization systems differ only in terms of a single module. This is less evident than the *complete mismatch* condition and more likely to happen as a result of a system design oversight. Experiments were performed using the following system variants:

- B3*, a variation of B3 in which the HiFi-GAN is replaced with a BigVGAN vocoder [142];
- B4*, a variation of B4 in which the EnCodec decoder is replaced with the Vocos [135] vocoder trained with character-level conditioning (corresponds to the system described in Section 4.3);

- B5* which uses a factorized time delay neural network TDNN-F [51] bottleneck feature extractor instead of the wav2vec 2.0 [80] encoder.[2]

The experimental setup is unchanged, but this time involves the evaluation of E^{B3} using either $\mathrm{ASV}^{\mathrm{B3}}$ or $\mathrm{ASV}^{\mathrm{B3}^*}$, E^{B4} using either $\mathrm{ASV}^{\mathrm{B4}}$ or $\mathrm{ASV}^{\mathrm{B4}^*}$, and E^{B5} using either $\mathrm{ASV}^{\mathrm{B5}}$ or $\mathrm{ASV}^{\mathrm{B5}^*}$.

Results are presented in Table 5.2. For each row, the difference between the two values measures the privacy overestimation given by the use of a mismatched ASV attacker. For evaluation data E^{B3}, the difference in performance is negligible, likely because the use of only a different vocoder has no consequential impact upon anonymization behavior. For E^{B4} the impact is more substantial, with a difference of 6% in EER for $\mathrm{ASV}^{\mathrm{B4}}$ and $\mathrm{ASV}^{\mathrm{B4}^*}$.

The character conditioning mechanism of B4* likely has a greater impact on anonymization, hence leading to a greater mismatch of around 6%. For E^{B5} the difference in EER is even more pronounced, with a difference of almost 10%. The use of a different feature extractor early in the pipeline results in almost completely different systems (see results in Table 5.1).

Even a partial mismatch between anonymization systems can still induce substantially exaggerated performance estimates; however, as intuition would suggest, the mismatch is less pronounced than in the case of a complete mismatch. While even this scenario might easily be avoided, there are practical examples in the literature. In [77] (C1) and the system proposed in [112], the attacker is assumed to retrain one of the system modules, thereby inducing a partial mismatch. In system T08-5 [88] of the 2024 VPC (see Section 2.4.4), E^{S} is formed by mixing data generated using two different anonymization systems, and the attacker is assumed not to know with which system each test utterance is anonymized; since the ASV model cannot be matched to both systems, there is again a partial mismatch. In all of these works, evaluation is reportedly performed under the VPC-defined semi-informed attack model. This observation shows that its definition is inadequate and that differences in interpretation can lead to questionable performance comparisons. This finding calls for the design of an automatic method to estimate the reliability of results derived using an ASV system $\mathrm{ASV}^{\mathrm{S}}$ regardless of the attack definition. We propose one such method in Section 5.1.2.

Hidden mismatch

As we now show, some mismatches can be more insidious. We start with the case of system C1 [77] for which results are shown to the left in Table 5.3. When evaluation is performed using $\mathrm{ASV}^{\mathrm{C1}}$ and E^{C1} we obtain an EER of 41%. However, we found that the EER can be reduced very substantially to 10% (a 74% relative decrease) through a trivial retraining of the ASV system. The retrained system, denoted $\mathrm{ASV}^{\mathrm{C1\text{-}rand}}$, stems from the anonymization of each utterance within

[2] System B5* is identical to the B6 baseline of the 2024 VPC described in Section 2.4.3. Denoted here as B5* for consistency as a *variant* of system B5.

$T^{\text{C1-rand}}$ using an embedding extracted for a speaker selected at random from the *LibriTTS train-other-500* dataset [47] rather than that extracted for speakers selected using the OHNN-based target speaker selection strategy. We now illustrate why such a training approach results in a more effective attack. The OHNN-based target speaker selection strategy is a completely deterministic function OHNN($\mathbf{x}$) applied to a speaker embedding $\mathbf{x}$. It is well-known [116] that, if the target speaker selection strategy is not sufficiently random, then utterance-level anonymization behaves similarly to speaker-level anonymization. This is because, if two embeddings $\mathbf{x}_1$ and $\mathbf{x}_2$ are extracted from utterances produced by the same speaker, then $\mathbf{x}_1 \approx \mathbf{x}_2$. It follows then that $\text{OHNN}(\mathbf{x}_1) \approx \text{OHNN}(\mathbf{x}_2)$, hence speaker-level, rather than utterance-level anonymization.[3]

It has been shown previously that speaker-level anonymization of T^{S} results in a weaker attack [53], yet a thorough explanation for why is lacking. We now offer such an explanation. If speaker A in T^{S} is consistently anonymized towards target speaker B, ASV$^{\text{S}}$ may simply learn to associate speaker B's voice characteristics with label A. As a result, it may rely on the $B \Rightarrow A$ mapping instead of learning speaker-discriminative features that persist despite anonymization. Consequently, ASV$^{\text{S}}$ is likely to fail on E^{S}, which includes different speakers and different anonymization mappings. This is the case for `C1`. By using a random target speaker selection strategy which results in $T^{\text{C1-rand}}$ being anonymized at the utterance level, ASV$^{\text{C1-rand}}$ will no longer learn unreliable cues. This results in the learning of other, more reliable cues, hence the lower EER.

To provide empirical evidence in support of these arguments, we conducted a set of experiments using a variant of `B3`, denoted `B3-SL`, which performs speaker-level rather than utterance-level anonymization. We used B3-SL to create $E^{\text{B3-SL}}$ and $T^{\text{B3-SL}}$, both anonymized at the speaker level but with different target speaker mappings. As shown to the right in Table 5.3, the resulting ASV$^{\text{B3-SL}}$ system produces an EER of almost 45% when assessed using $E^{\text{B3-SL}}$. We then repeated the evaluation using $E^{\text{B3-SL}}$ and ASV$^{\text{B3}}$ trained using data anonymized at the utterance level (i.e. the standard VPC 2024 methodology) and obtained a greatly-reduced EER of 27%. ASV$^{\text{B3-SL}}$ provides an overestimation of privacy protection as a result of being trained using data anonymized at the speaker level. These findings show that the evaluation of anonymization systems performed using the same system as that used in generating E^{S} (i.e. the semi-informed attack) may not always provide reliable performance estimates. Such hidden mismatches can be challenging to mitigate.

[3] This property is of course not general and does not hold for particularly "non-smooth" functions. However, in practice, neural networks that produce speaker embeddings generally behave smoothly by design, since utterances that sound similar must be mapped to similar embeddings. The OHNN in [77] is trained specifically to *preserve* the distribution of unprotected speaker embeddings, and it reasonable to assume that it works similarly to a speaker embedding extractor. This assumption seems to hold in practice.

Table 5.3: Results for the hidden mismatch scenario (EER, %).

E^{C1}		$E^{\text{B3-SL}}$	
ASV$^{\text{C1}}$	ASV$^{\text{C1-rand}}$	ASV$^{\text{B3-SL}}$	ASV$^{\text{B3}}$
40.80	10.38	44.78	27.31

5.1.2 Detecting mismatches

The overestimation of privacy protection can be attributed to ASV$^{\text{S}}$ *overfitting* to the data distribution of T^{S} due either to the use of mismatched anonymization systems or the use of ineffective training strategies. This causes ASV$^{\text{S}}$ to underperform on E^{S}. In the following we propose one solution to detect overfitting: we follow the typical approach of reserving a portion of the training set for validation purposes. It is well know that a large gap in some performance metric computed on both training and validation data is an indication of overfitting [143]. In the case of VA, the most natural choice of metric is the speaker verification EER. We reserve 10% of T^{S} for validation purposes.[4] In contrast to E^{S}, the validation data contains the same speakers as T_s.

In order to be able to compute an EER on the speaker verification task, we design a new ASV protocol on the validation split. For each speaker, 5 utterances are used for enrollment. Those remaining are reserved for ASV trials. On average, there are 8 trials per speaker, 3 of which are target trials. ASV$^{\text{S}}$ is trained in the usual way. It is separately applied to the validation data in addition to the evaluation data E^{S}: this results in a pair of performance estimates, EER$_{\text{val}}$ and EER$_{\text{test}}$. Our hypothesis is that large differences in these two metrics can serve as an indication of overfitting. We performed this experiment for all evaluation cases reported so far: with systems B3, B4 and B5 under full and partial mismatch conditions, and C1 and B3-SL under the hidden mismatch condition. To provide more data points, we also computed the EER$_{\text{val}}$ and EER$_{\text{test}}$ couple for the matched condition of two more VPC 2024 baselines: B5* (which corresponds to B6) and the less competitive B2 baseline [64].

Results are illustrated in Figure 5.2, where the x and y axes correspond to EER$_{\text{test}}$ and EER$_{\text{val}}$ respectively. Green stars correspond to matched scenarios for 2024 VPC baseline systems. Red triangles correspond to full mismatch. Orange squares correspond to partial mismatch. Purple circles indicate hidden mismatch, while purple stars correspond to the same scenarios with 'corrected', more effective ASV training data generation. Annotations signify the systems involved in each evaluation, e.g. (B4,B5) corresponds to E^{B4} being used for evaluation of ASV$^{\text{B5}}$. For all cases, EER$_{\text{val}}$ is substantially lower than EER$_{\text{test}}$: this is on the account of the overlap between

[4] This does not diminish the total amount of data used to train the attacking ASV system: technically speaking, 10% of T^{S} is already reserved in the original VPC 2024 pipeline for the purpose of validating the *checkpoint* of the ASV model used for the final attack evaluation. However, the metric used to select the checkpoint is speaker identification accuracy, and the split is randomized for every training run. In contrast, our split is fixed for all runs to ensure that the results are comparable.

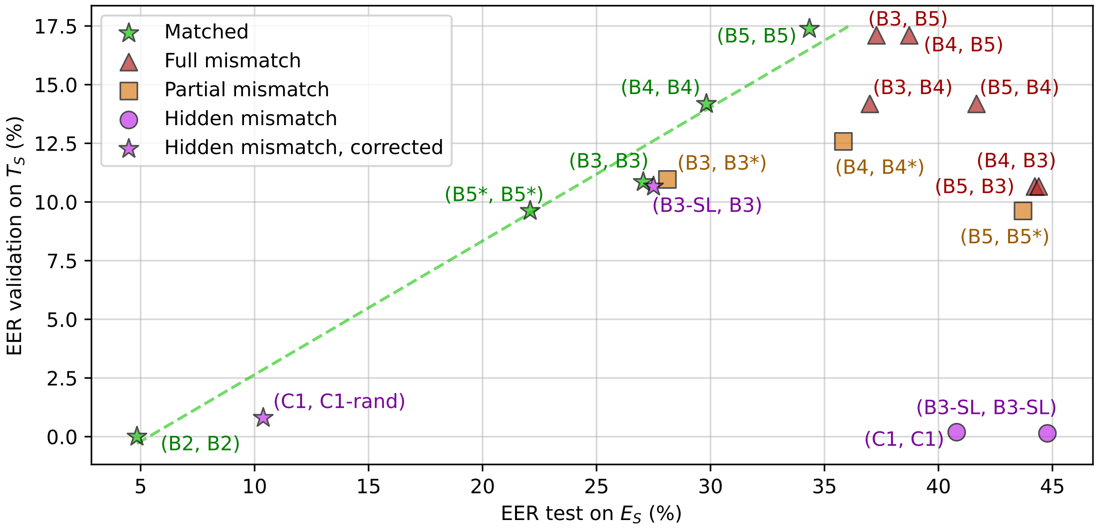


Figure 5.2: Values of EER$_{\text{val}}$ plotted against EER$_{\text{test}}$ for all considered systems, denoted as (E^{S}, T^{S}) pairs. The green dashed line is the regression line of the *"Matched"* systems. Systems falling below this line have a potentially mismatched evaluation.

speakers in the validation set and the training set T^{S}. The gap between EER$_{\text{val}}$ and EER$_{\text{test}}$ is more pronounced for mismatched scenarios. For instance, in the case of matched (B3,B3), the EER$_{\text{test}}$ of 27% falls to an EER$_{\text{val}}$ of 11%, a relative drop of 60%. However, when pairing B3 with B4 into the the fully mismatched case of (B4,B3), an EER$_{\text{test}}$ of 44% falls to an EER$_{\text{val}}$ of 10%, a relative drop of 76%. The trend is consistent; for mismatched scenarios the gap between EER$_{\text{test}}$ and EER$_{\text{val}}$ is higher than that for matched scenarios.

Also plotted in Figure 5.2 is a regression line computed over the EER$_{\text{test}}$, EER$_{\text{val}}$ point pairs corresponding to matched scenarios, all of which we assume to be well-evaluated reference cases. Nearly all other points, all corresponding to mismatched scenarios, fall below this line suggesting that the difference between EER$_{\text{val}}$ and EER$_{\text{test}}$ serves as an indicator of how well-trained or well-matched is ASV$^{\text{S}}$ to the evaluation of anonymization system S: the greater the gap, the lower the ratio, the higher the chance of there being a mismatch between training and evaluation data, and therefore a less reliable estimation of anonymization performance. We also note that cases of hidden mismatch (C1 and B3-SL) are the farthest from the regression line. However, once the training correction to the attacking ASV system is applied, their corresponding points (B3-SL, B3) and (C1, C1-rand) move closer to the regression line. It could be argued that the distance from the regression line can serve as an indicator of how reliable the privacy protection estimation is.

In the previous section, we suggested that the semi-informed paradigm may not always yield the best possible attack. The $\text{EER}_{\text{val}}/\text{EER}_{\text{test}}$ ratio (and the distance from the 'matched' regression line) can be computed regardless of the kind of conducted attack, as long as some training of ASV^{S} is done, making it flexible. We argue that the community should consider the adoption of such detection techniques for the evaluation of anonymization systems so as to reduce the risk of overestimating performance. To encourage work in this direction, we have integrated our approach into a publicly available fork of the 2024 VPC evaluation toolkit[5] which includes the automatic computation of EER_{val} within the validation step.

5.2 Proactive defense via adversarial noise

In the previous section, we showed how the EER used to estimate the privacy protection provided by an anonymization system can be seen as stemming from two contributions: the ability of the VA system to suppress speaker-specific voice traits from the input signal, and the *inability* of the attacking ASV system to learn whatever personally-identifiable cues are left. In our analysis, we considered the case where the latter contribution is unintentionally introduced by the system designer, and hence represents an overestimation of privacy protection; however, hindering the attacking ASV system could be viewed as a deliberate strategy to reinforce the privacy protection already provided by the VA system. In this scenario, the contribution to the EER from ASV^{S} would no longer reflect overestimation, but rather an additional layer of defense.

While the results presented in Section 5.1 showed that mirroring the defender's setup does not always yield an optimal attack, they also indicate that the attacker must employ a system structurally similar to the one that anonymized the trial utterances D^{S}, otherwise a partial mismatch may occur. This implies that, under the assumption of a worst-case scenario, the method used by the attacker to anonymize either A^S or T^{S} is at least partially known. Moreover, to perform any attack at all, the attacker has to somehow acquire unprotected audio A and T, and has to be able to anonymize them.

In this section, we investigate whether it is possible for the defender to exploit this knowledge to actively hinder the attacker. We explore this by designing adversarial perturbations that, when applied to clean speech, significantly reduce the quality of the resulting anonymized signal. Consider the standard scenario where the defender applies anonymization to the trial data D^S with a given VA system S. In order to re-identify the data in D^S, the attacker must collect unprotected data A containing the same speakers as in D^S. Data A could originate from sources where the speaker does not intend to be anonymized, e.g. social media posts where users intentionally use their real voices. These sources act as "points of failure," and we assume that the defender can identify and pre-process them before they are publicly released.

[5] https://github.com/DigitalPhonetics/voice-privacy-evaluation

The defender then applies to the clear data in A adversarial audio perturbations that aim to disrupt the functioning of system S in particular. If the attacker later acquires A and attempts to anonymize it for use as enrollment data in their attacking system ASV^{S}, the resulting A^S is of degraded quality, thereby compromising the attack.

5.2.1 Related works

To the best of our knowledge, the use of adversarial perturbations to enhance the robustness of VA systems is unexplored in the literature. In the context of speech security, adversarial noise generation has been applied in various contexts [144, 145]. Namely, it has been employed as a proactive defense against voice cloning [30, 31, 146, 147]: most works on this topic focus on generating noise that shifts an utterance's position within the speaker embedding space of one or more speaker encoders.[6] This prevents a VC or TTS system from using a protected utterance as a target speaker reference, since an accurate speaker embedding cannot be extracted from it. Our work differs in that it aims to protect the utterance used to extract the speech content of the anonymized signal rather than the target speaker identity (i.e., the pseudo-speaker in VA terminology). Moreover, the referenced algorithms preserve the intelligibility of the final synthetic signal despite protecting the threatened speaker's identity; in contrast, our goal is to degrade to the greatest extent possible the quality of the final anonymized signal, including its speech content, so that an attacker can extract as little usable information as possible.

Adversarial perturbations have naturally been explored for the task of misleading ASV systems [34, 148, 149]; however, they typically target ASV models directly, either in white-box or black-box contexts. Our goal is different: we aim to disrupt the functioning of an ASV system trained specifically to re-identify speakers from anonymized utterances, and we do so by targeting the VA system itself rather than the ASV model.

5.2.2 Adversarial noise generation

As previously illustrated, in the worst-case scenario, the attacker uses a similar VA system to the defender to anonymize A and T. This implies that the defender can, by assumption, rely on white-box access to the VA system most likely to be used by an informed attacker — an important advantage, since it is well known that adversarial noise generation is more effective in white-box settings than in black-box ones.

[6] While the authors of [30] frame their task as "voice anonymization", their algorithm only anonymizes a given utterance in the speaker embedding space and does not mask the voice of the original speaker acoustically. Therefore, it is closer in priciple to [146, 147] and does not fit VPC's definition of "voice anonymization" (see Section 2.1.1).

In general, the adversarial noise has to disrupt the *downsampling* process of the unprotected utterance (see Section 4.4), so that no meaningful information can be extracted from it — including any speaker-specific cue that ASV$^{\mathrm{S}}$ might exploit. However, to fully exploit the white-box setting, the adversarial noise generation algorithm must be tailored to the VA system under consideration. For the purpose of this work, we focus on B5, the VPC 2024 baseline with highest privacy protection level.

Objective function

As previously detailed in Section 2.4.3, B5 extracts the speech content of the input utterance by first processing it with wav2vec 2.0 [80], then applying VQ over the obtained continuous features. The presence of a VQ layer represents a challenge in the generation of adversarial noise, since it introduces a non-differentiable operation in the network topology [150]: gradient propagation can still be carried out via straight-through estimation [122], but its accuracy is reduced. This makes it impractical to adopt common approaches to end-to-end adversarial noise optimization [34, 145, 149, 151, 152]. However, our goal is to disrupt the extraction of features from the input utterance that contain speaker-specific cues. Therefore, it is sufficient to target with adversarial noise the wav2vec 2.0 embeddings *before* they are quantized by the VQ layer, avoiding the non-differentiable quantization operation. In the following, we detail our approach.

Let $\mathbf{u} \in \mathbb{R}^L$ be the unprotected input utterance, from which wav2vec 2.0 extracts a feature vector $\mathbf{h}_t$ at each time frame t. Our goal is to find an adversarial perturbation $\delta \in \mathbb{R}^L$ such that $\mathbf{u}^\delta \triangleq \mathbf{u}+\delta$, when processed by B5, produces a highly degraded output that is unsuitable for enrollment in ASV$^{\mathrm{B5}}$. To that end, we optimize δ so that, for as many time frames as possible, the codeword assigned to the quantized version of $\mathbf{h}_t$ differs from the codeword assigned in the unperturbed case (i.e., without the application of adversarial noise). More formally, let $\mathbf{h}_t^\delta$ be the feature vector extracted for time frame t from $\mathbf{u}^\delta$. Our objective can be expressed as

$$\max_{\delta} \frac{1}{T} \sum_t \mathbb{1}\left[\ \mathrm{VQ}\left(\mathbf{h}_t\right) \ \neq \ \mathrm{VQ}\left(\mathbf{h}_t^\delta\right)\ \right] \tag{5.1}$$

where $\mathbb{1}[\cdot]$ is the indicator function and T is the total number of time frames. As previously mentioned, this objective cannot be optimized directly with respect to δ due to the VQ operation. However, it is possible to optimize a similar objective function *before* the embeddings reach the quantization layer, yielding the same functional effect:

$$\max_{\delta} \frac{1}{T} \sum_t \left|\mathbf{h}_t - \mathbf{h}_t^\delta\right|_1 \tag{5.2}$$

With the quantization layer bypassed, gradients can be back-propagated normally through the feature extractor. This formulation serves as a differentiable surrogate for maximizing codeword flips, since increasing the distance between $\mathbf{h}_t$ and $\mathbf{h}_t^\delta$ raises the likelihood of a change in the assigned codeword after quantization.

In this setting, we found the L1 loss to yield more consistent results than the more conventional Mean Squared Error (MSE). This is because the MSE grows quadratically with displacement of $\mathbf{h}_t^\delta$, which can lead to very large loss values and cause optimization divergence. The L1 loss, by contrast, produces smaller, more stable values and better suits our use case, since we only need to perturb the vector $\mathbf{h}_t^\delta$ just far enough from its original position to move it outside its Voronoi region and assign it a different codeword than $\text{VQ}(\mathbf{h}_t)$.

Optimization technique

For each (unprotected) utterance in A, we optimize a different δ according to (5.2) with Projected Gradient Descent (PGD) [153]. The boundary of the accepted values of δ is a L2-ball of radius ϵ. The gradient descent steps are computed with Adam [119].

During our initial experiments, we noticed that the adversarial noise became less effective if, after optimizing δ, the utterance $\mathbf{u}^\delta$ was padded before being forwarded through B5. This situation commonly arises when processing batches of utterances, as they are typically zero-padded on the right to match the duration of the longest utterance in the batch. The official VPC 2024 implementation of B5 follows this approach as well. To make the adversarial noise robust to padding, we zero-pad $\mathbf{u}^\delta$ by a random amount (up to P seconds) before each optimization step. The duration of the padding varies at each step. The zero-padded region is excluded from the perturbation (δ always has the same duration as $\mathbf{u}$).

5.2.3 Experimental setup and results

We optimize and apply adversarial noise for B5 to the unprotected utterances in *Libri-dev-enrolls* and *Libri-test-enrolls*. We carry out the rest of the evaluation in the usual way according to the VPC 2024 protocol. Note that, in the original VPC setup, B5 performs inference with a batch size of 8.

Optimization of δ is performed until one of the following stopping criteria are met: *(a)* 200 iterations are performed; *(b)* 95% of the quantized embeddings of $\mathbf{u}^\delta$ have been assigned a different codeword than its clean counterpart $\mathbf{u}$. The latter condition is equivalent to the function in (5.1) reaching a value of 0.95 or higher. The learning rate of Adam is set to 1e-4. The maximum padding P is 3 seconds. ϵ is set to 2.

Table 5.4: Privacy protection levels (EER, %) of system B5 in normal conditions (third column), with proactive adversarial protection (fourth column), and when the proactive protection's robustness is enhanced by random zero-padding (fifth column).

Dataset	**Unprotected**	**B5**	**B5 (adv. noise)**	**B5 (adv. noise + padding)**
Libri-dev	5.72	34.37	44.64	48.48
Libri-test	4.59	34.34	44.40	48.23

Results in privacy protection

Results are shown in Table 5.4. The third column shows the EER normally achieved by B5, as already seen in the previous sections. The fourth column reports the EER obtained by ASV$^{\mathrm{B5}}$ when the unprotected enrollment utterances include the adversarial noise δ. In the fifth column, δ is made more robust to batching through random padding during the optimization process. The proactive protection with adversarial noise works as expected, raising the EER from 34% to 44%. With the additional robustness from random padding, the EER further increases to 48%, approaching perfect privacy.

We apply the validation method presented in Section 5.1.2 to the proposed proactive defense technique. Figure 5.3 extends the scatterplot of Figure 5.2 by adding two new data points: ASV$^{\mathrm{B5}}$ evaluated on the data perturbed with adversarial noise (blue diamonds), both with and without padding augmentation (labeled as $B5+\delta+A$ and $B5+\delta$, respectively). The two new data points fall below the green regression line defined by the ground-truth *matched cases* (green stars). This is expected: as detailed in Section 5.1.2, it indicates that a portion of the EER$_{\mathrm{test}}$ stems from sub-optimal performance of the attacking ASV system. In the scenarios reported throughout Section 5.1, this reflected *unintentional* privacy overestimation. By contrast, in our case it represents a *deliberate* strategy, as the goal of our adversarial noise is to proactively hinder the attacker's ASV system.

The effectiveness of the proposed method becomes evident when listening to the anonymized utterances protected with adversarial noise: most of the speech content is destroyed and replaced by buzzing noise. Disrupting the feature extraction process achieves the intended effect, leaving no speaker-specific cue for ASV$^{\mathrm{B5}}$ to leverage in re-identifying the trial utterances in D^{B5}.

Note that the poor quality of the anonymized enrollment utterances does not pose a problem, as they are only used by the attacker to re-identify the anonymized trial utterances. However, it is crucial that the addition of adversarial noise does not excessively degrade the clean utterances: in our threat model, these were never meant to be anonymized and must therefore retain acceptable quality for downstream tasks and human listening. We now evaluate this aspect.

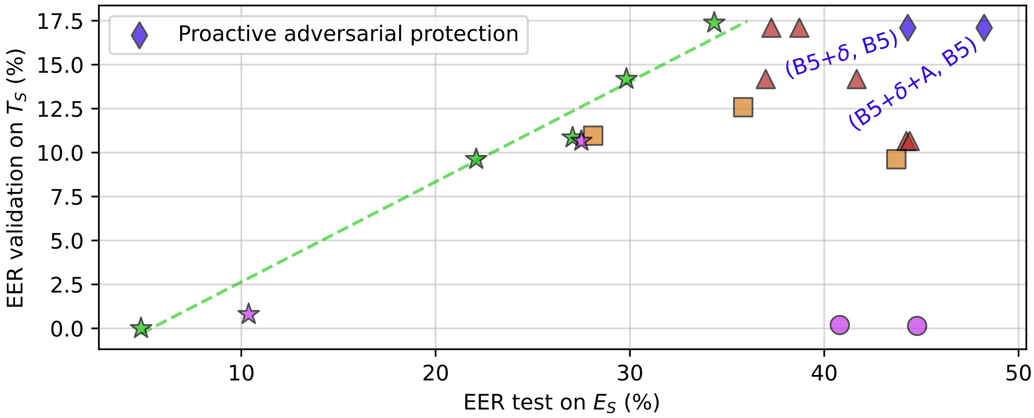


Figure 5.3: Extension of Figure 5.2 (values of EER_{val} plotted against EER_{test}) with the inclusion of the B5 data with proactive adversarial protection. As expected, the two new points are below the green regression line, since a consistent part of EER_{test} is due to the poor performance of ASV^{B5}.

Results in quality preservation

We assess the impact on quality of the adversarial noise with several metrics. First, we compare the WER scored by the ASR system of the VPC 2024 pipeline on the original unprotected enrollment utterances and on the enrollment utterances perturbed with the noise (we use random padding).[7] A small difference between the two is desirable as it indicates that an ASR system is not affected by the presence of the adversarial noise. We then compute the Signal-to-Noise Ratio (SNR), which indicates the relative energy of the speech signal compared to the added noise, expressed in decibels:

$$\text{SNR}_{\text{dB}}\left(\mathbf{u},\mathbf{u}^{\delta}\right) = 10\log_{10}\left(\frac{\sum_t (\mathbf{u}(t))^2}{\sum_t \left(\mathbf{u}^{\delta}(t) - \mathbf{u}(t)\right)^2}\right) \tag{5.3}$$

We also compute the Perceptual Evaluation of Speech Quality (PESQ) [154], which models human auditory perception to predict a mean-opinion-score-like estimate of overall speech quality, and Short-Time Objective Intelligibility (STOI) [155], which measures the correlation of temporal speech envelopes between clean and degraded signals to predict intelligibility. PESQ ranges from −0.5 (bad quality) to 4.5 (excellent quality), while STOI ranges from 0 (unintelligible) to 1 (perfect intelligibility).

[7] The WER here is different from that reported for unprotected utterances in the VPC 2024 description in Section 2.4.5 (specifically in Table 2.5), since the VPC 2024 pipeline computes the WER from the trial utterances, not enrollment utterances. The rationale behind this choice in the VPC 2024 protocol is that only trial utterances are meant to be used in a downstream task — hence the WER is estimated exclusively on them.

Table 5.5: Quality metrics for the data perturbed with adversarial noise. The values of SNR, PESQ and STOI are averaged across utterances.

Dataset	WER original	WER (adv. noise)	SNR_{dB}	PESQ	STOI
Libri-dev enrolls	1.73	2.46	28.25	2.51	0.97
Libri-test enrolls	1.83	2.62	27.28	2.61	0.97

We report the results in Table 5.5. WER values marginally increase from around 1.8% to 2.5%, less than a one-point increment in absolute terms: a level that falls within the expected range for clean speech. Values of SNR are relatively high, indicating that the noise energy is much weaker than the original signal. A STOI of 0.97 suggests that intelligibility is essentially unaffected: a listener should be able to understand nearly all the words in the utterance. A PESQ score of ~ 2.5 indicates moderate quality, meaning that the utterance remains usable though the noise may be noticeable in some cases. However, the metrics indicate that the overall quality degradation is marginal and does not seriously affect their usability — something we confirmed through informal listening tests.

5.3 Conclusions

In the first part of this chapter, we demonstrated the risk of overestimating the privacy protection provided by a VA system. Performance overestimation occurs when the EER reported for a specific VA system does not stem solely from the effectiveness of the system in suppressing speaker-specific cues from the input data, but also from the *ineffectiveness* of the attacking ASV model in detecting and exploiting said cues.

Using several state-of-the-art VA approaches, we showed that overestimation can be caused by mismatches between the data used to train the speaker verification system employed for evaluation and the anonymized data under test. We demonstrated the risk with artificially introduced mismatch and identified similar, hidden mismatches that afflict the evaluation of practical systems reported in the literature leading to exaggerated reports of performance. Based upon the comparison of results derived for test and validation sets, our method to detect overestimated performance is effective in identifying all examples reported. Consequently, we advocate for the adoption of such detection techniques within the community to protect trust in performance estimates.

While privacy overestimation stems from the attacking ASV system having suboptimal performance unbeknownst to the evaluator, proactively targeting an adversary's system can serve as a deliberate strategy to strengthen the protection provided by an existing VA system. In the second part of this chapter, we explored this idea. To implement a powerful attack, an adversary must employ a VA system similar to the defender's to generate anonymized training and enrollment data. The defender can then apply a proactive defense, adding adversarial noise to potentially exploitable data to prevent its anonymization. We demonstrated the effectiveness of this approach by applying it to one of VPC 2024 baselines, improving its performance to nearly-perfect privacy protection. This concludes our investigation of the attacker's role, representing the final contribution of this dissertation.

Chapter 6

Conclusions and future work

In this chapter, we provide a high-level summary of the research presented in this dissertation, along with some “lessons learned”. Furthermore, we outline potential research directions stemming from the work in each chapter and, where relevant, connect them to recent literature that points in those directions.

Chapter 2 - The VoicePrivacy Challenges

We presented the VPC challenges and their evaluation framework, which form the foundation of this dissertation. We introduced the core definition of VA: the task of obfuscating the voice identity of a speaker in a speech signal while maintaining linguistic content (i.e., “what is being said”) and other paralinguistic attributes (e.g., prosody, emotion). The task can be framed as an interaction between a *defender*, who uses a VA system to anonymize a dataset of utterances, and an *attacker*, who attempts to re-identify the speakers in that dataset using an ASV system.

Anonymization is typically achieved by transforming the voice identity of the original speaker into that of a *pseudo-speaker* — either a real speaker or a fictitious one — typically through a VC or TTS system, or a combination of both. Within the context of the VPC, anonymization can be applied in two ways: at the *speaker level* or at the *utterance level*. In the former, all utterances belonging to one speaker are mapped to the same pseudo-speaker; in the latter, utterances of the same speaker can be assigned different pseudo-speakers. In order to perform the attack, the attacker is assumed to have at their disposal a set of unprotected utterances belonging to the speakers they intend to identify, which are used as enrollment data for their ASV system.

We described the three editions of the VPC held so far: 2020, 2022, and 2024. In the case of the 2022 edition, we also reported the post-evaluation analysis of the results performed by the candidate. Several notable findings emerged.[1] First, we computed EERs for similarity scores

[1] Analogous post-evaluation for the 2024 edition is ongoing.

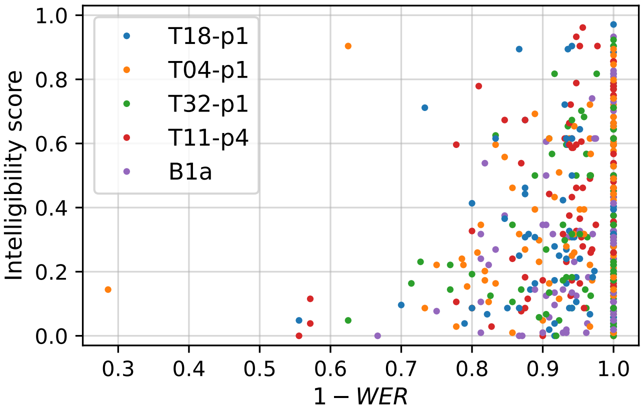


Figure 6.1: Value of 1−WER plotted against subjective intelligibility for a subset of VPC 2022 data. Reproduced from Figure 2.7 of Section 2.3.4.

provided by human listeners on a set of target and non-target trial utterances (either unprotected or anonymized) compared to enrollment utterances (always unprotected). Listeners achieved a 22% EER for unprotected data, noticeably higher than the average 4% achieved with an x-vector system. The same trend emerged with anonymized data, with human listeners scoring close to 50% EER for all VA systems. This suggests that, at least for untrained listeners, even simple signal-processing-based VA systems like B2 can provide some reasonable level of privacy protection, despite their poor performance against ASV-based semi-informed attacks. This result highlights the importance of clearly defining use cases in VA: system performance can vary notably depending on the evaluation scenario and the downstream task. An initial step towards defining a proper taxonomy of use cases for VA was taken in [28].

Second, we showed that there is no meaningful correlation between objective WER and subjective intelligibility; notably, utterances that were perfectly transcribed by the ASR system were often assigned low or even near-zero intelligibility scores (see Figure 6.1). In general, these results suggest that subjective and objective metrics are not functionally equivalent, and should not be used as proxies for one another. However, both the identifiability scores and the quality ratings came from untrained listeners who were not familiar with the anonymized speakers. Future work on subjective privacy evaluation should consider scenarios where the listeners have some degree of familiarity with the anonymized speakers and/or are trained to identify voice-related cues that might reveal the speaker's identity. Such a scenario was explored in [156], where the authors studied the subjective evaluation of VA for the purpose of anonymous therapy sessions, during which therapists gain familiarity with a patient over time. The ability of identifying anonymized speech was compared between a group of therapists and one of "amateurs": the former clearly outperformed the latter, suggesting that domain expertise aids in re-identification.

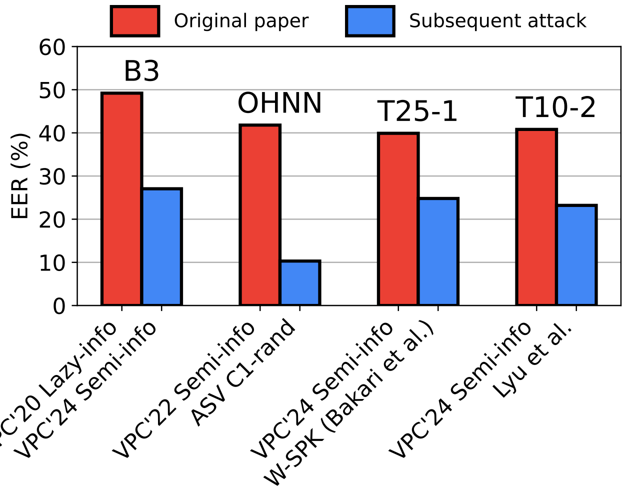


Figure 6.2: Examples of systems that initially achieved a high EER, which was then notably reduced by a subsequent attack. This shows the "cat-and-mouse" nature of the VA task.

Third, we performed DTW-based re-alignment of the F0 trajectories of system T04, to demonstrate the potential of designing TTS-based VA systems that could still perform acceptably in terms of pitch correlation. Baseline B3 is, in a sense, a concrete implementation of that idea, conditioning the TTS synthesis on the F0 curve of the original utterance.

Lastly, inspecting the EER trends as a whole throughout the various editions of the VPC, one can observe clearly that the VA task is indeed a "cat-and-mouse game": for each evaluation setup, there is always some system that claims EER> 40%, until a more powerful attack is developed and brings the score down to ~ 25% or less (see Figure 6.2). That has been the case for B3 (49.2% [83] → 27.1% [37]), the OHNN-based system (41.8% [77] → 10.3% [54]), and more recently, systems T10-2 (40.8% [91] → 23.2% [157]) and T25-1 (39.9% [99] → 24.8% [158]) for the 2024 VPC. This shows the importance of the research on attacks on VA systems: the development of increasingly powerful attacks fosters further research, and ultimately pushes the boundary privacy protection provided by VA.

Chapter 3 - Vocoder drift

We introduced the notion of *vocoder drift*, which represents the contribution of the vocoder to the overall change in voice identity applied to an utterance by an x-vector–based VA pipeline. Given some utterance $\mathbf{u}$ to anonymize and its associated x-vector $\mathbf{x}_o$, we defined the vocoder drift as $d(\mathbf{x}_p, \mathbf{x}_a)$, where $\mathbf{x}_p = a(\mathbf{x}_o)$ is the result of the anonymization function $a(\cdot)$ applied to $\mathbf{x}_o$, $\mathbf{x}_a$ is the x-vector extracted from the final anonymized utterance, and d is some appropriate distance

Table 6.1: Merged view of (i) target distance and HiFi-NSF drift, (ii) privacy protection across x-vector domains (EER, %), and (iii) HiFi-NSF attack outcomes (EER, %) on LibriSpeech and VCTK test sets, separated by speaker sex. Results reproduced from Sections 3.3 and 3.4.

	Distances		***x*-vector domains (EER, %)**			**Attacks (EER, %)**	
	target	**drift**	**$\hat{O}$ dom.**	**$\hat{P}$ dom.**	**$\hat{A}$ dom.**	**Semi-informed**	**Drift reversal**
LibriSpeech (F)	1.3	0.97	0.54	2.51	16.2	4.01	4.23
LibriSpeech (M)	1.2	0.94	0.88	2.99	19.0	2.23	4.90
VCTK (F)	1.3	0.94	1.13	5.59	28.1	18.4	14.1
VCTK (M)	1.3	0.90	0.17	3.04	19.1	11.7	11.1

metric. The idea is to isolate the contribution of the vocoder, which is the only processing step separating $\mathbf{x}_p$ from $\mathbf{x}_a$, to the overall x-vector of the anonymized utterance. To have some means of comparison to the actual numerical value of the vocoder drift, we also defined the *target distance* $d(\mathbf{x}_o, \mathbf{x}_p)$, which represents the effect of the anonymization function $a(\mathbf{x}_o) = \mathbf{x}_p$ — intuitively, it should contribute exclusively to the overall x-vector position, with the vocoder introducing only negligible changes.

We found this not to be the case. Our experiments, which used an ECAPA-TDNN for x-vector extraction, the same pool-based anonymization function as baseline B1, and several vocoders, showed that the target distance and vocoder drift are of the same order of magnitude, with the vocoder drift being at least 0.5 times the value of the target distance: a value that cannot be considered negligible. The first two columns of Table 6.1 show this for the HiFi-NSF vocoder, with the average target distance and drift values being 1.2 and 0.94 respectively.

To further investigate the impact of vocoder drift on privacy protection, we performed ASV experiments using $\mathbf{x}_o$ vectors (unprotected), $\mathbf{x}_p$ (anonymized, but before the final vocoder synthesis) and $\mathbf{x}_a$ (extracted from the final anonymized utterance, corresponding to a lazy-informed attack). In our setup, EERs in the domains of $\mathbf{x}_o$ and $\mathbf{x}_p$ were comparable (~ 0.7% and ~ 2.7% on LibriSpeech in the case of the HiFi-NSF, see Table 6.1), indicating that the anonymization function itself contributes little to the overall privacy protection. Conversely, and consistent with our results for vocoder drift, the domain of $\mathbf{x}_a$ showed a marked increase in EER (~ 17%), pointing to a substantial contribution of the vocoder to privacy protection.

Last, we trained an attack model that has the goal of inverting said contribution, retrieving the original speaker identity but only reversing the effect of the vocoder. The power of this attack was, in some instances, comparable to that of a semi-informed attack: for example, as shown in Table 6.1, drift reversal and semi-informed attack scored ~ 4.5% and ~ 3.1% EER respectively in case of the HiFi-NSF vocoder.

In Section 3.7, we discussed the main insight gained from our study of vocoder drift: to maximize privacy preservation, one should focus less on the pseudo-speaker selection strategy and instead prioritize the voice conversion technique. Considering subsequent research in VA, including that of the candidate, the results of Chapter 3 corroborate later findings in the field, which we now describe.

One of our conclusions was that anything beyond a random pseudo-speaker choice was potentially unnecessary. It was further corroborated by the findings of team T12 of the 2024 VPC [89], who attempted to replace the GAN-based embedding anonymizer in B3 (see Sections 2.4.3 and 2.4.4) with a random selection from an external pool of embeddings taken from *LibriTTS-train-clean-100* [47]. Doing so resulted in no significant changes in EER. Indeed, in the 2024 VPC, more participants seem to have preferred a simple random selection of pseudo-speaker: it is the case of systems T08, T19, T30 and T38 (Section 2.4.4).

This idea can be elaborated further: deterministic pseudo-speaker selection algorithms should be avoided, as they can result in privacy overestimation. The case of the OHNN-based VA system against $\text{ASV}^{\text{C1-rand}}$ presented in Section 5.1 is a prime example, but not the only one. In [159], Franzreb et al. propose a VA technique that achieves nearly 50% EER when the pseudo-speaker is constrained to be of the same sex as the original speaker; however, the EER drops to $\sim 27\%$ when using a completely random pseudo-speaker selection. To a lesser extent, a similar phenomenon was observed by Champion in [38] for the "dense" pseudo-speaker selection algorithm, which constrains a speaker to a certain cluster of pseudo-speakers.

Another finding of Chapter 3 was that the cause of drift arose from the interaction between the anonymized x-vector $\mathbf{x}_p$ and the rest of the features processed by the vocoder. In hindsight, this could have been presented as further proof that linguistic and prosodic features carry personally-identifiable information that need to be sanitized. The issue was first addressed explicitly in [76], which led to the development of systems B5 and B6, and subsequently inspired the candidate's work on NAC language modeling (Chapter 4).

Last, the idea that the vocoder contributes to privacy protection can be generalized: additional processing layers generally introduce more randomness into the final waveform, and therefore better anonymity. This is the core idea of [132], where the authors proposed a simple VA strategy that anonymizes an utterance twice in succession with two different VA systems. As expected, this increased the levels of privacy protection (to various degrees depending on the VA system employed), always at the cost of a proportional degradation in content preservation.

The study of vocoder drift was an analysis specifically targeted at x-vector–based VA; with the field moving towards different techniques, it is difficult to envision further work on this topic. However, concepts that were tackled in Chapter 3 have repeatedly resurfaced in discussions within the VA community, confirming the relevance of this research.

Chapter 4 - Voice anonymization with neural audio codecs

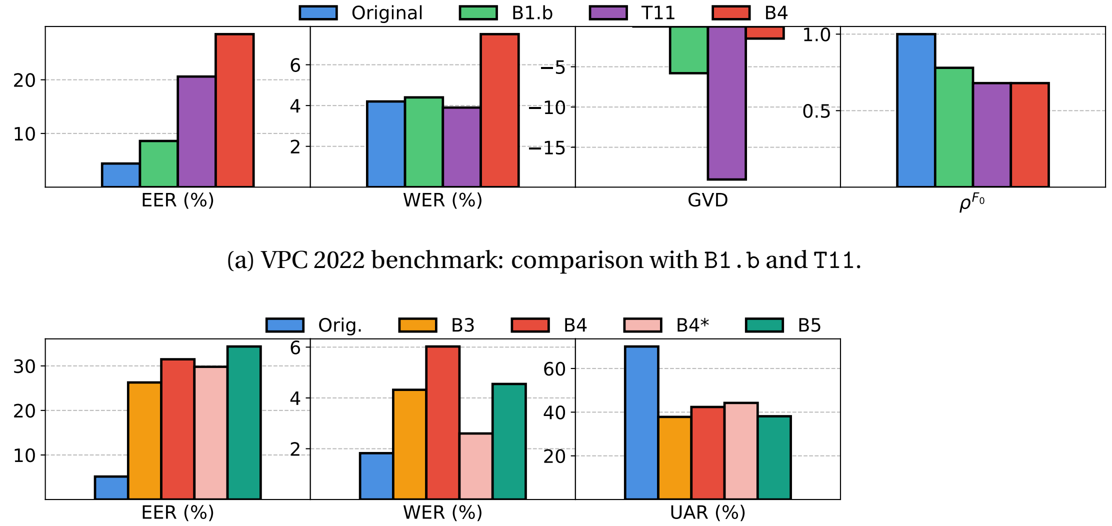
We proposed a novel approach to VA using NAC language modeling. Following the paradigms developed in [126, 129], and assembling modules from Bark,[2] we designed a VA system based on the interplay of two types of tokenized speech representations: *semantic tokens* and *acoustic tokens*. Semantic tokens come from a HuBERT-like model and encode the speech content of the input utterance; acoustic tokens are produced by the EnCodec NAC encoder from a pseudo-speaker's utterance and represent their voice identity. A pair of Transformer models subsequently generates a new set of acoustic tokens, which the NAC decoder converts into a waveform. This new, anonymized waveform retains the semantic content of the input utterance but is rendered in the voice of the pseudo-speaker. The pseudo-speaker is chosen at random from a pool of existing utterances. The characteristics of this system align with the conclusions from Chapter 3: it employed a more complex VC technique, while the pseudo-speaker selection strategy is simple — a random choice among a pool of candidates. Following its use as a baseline for the 2024 VPC, the system was informally renamed as B4.

B4 was evaluated on the VPC benchmark in both the 2022 and 2024 editions, scoring competitively in both cases. Its results, along with those of other relevant VA systems for comparison, are summarized in Figure 5.1. On the 2022 benchmark (Figure 6.3a), B4 showcased above-average privacy protection (EER), strong voice distinctiveness (G_{VD}) and pitch correlation ($\boldsymbol{\rho}^{F_0}$), but underperformed slightly on speech content preservation (WER). This prompted the development of an extended version of the system, later named B4*, which incorporates the novel mechanism of character-level conditioning to improve speech content retention in the final anonymized utterance. On the 2024 benchmark (Figure 6.3b), both B4 and B4* scored a competitive EER, with B5 being the only baseline with a higher score; both surpassed B3 and B5 in terms of UAR; additionally, B4* achieved a WER close to unprotected speech, demonstrating the effectiveness of the character-level conditioning.

B4 (and, to a lesser extent, B4*) constitutes the most tangible and impactful output of the candidate's work, in no small part thanks to its inclusion among the VPC 2024 baselines. One of its intended goals was to explore alternative approaches to VA beyond the prevailing paradigm of "x-vector + anonymization function + vocoder". In hindsight, this goal was realized to some extent, as subsequent systems also adopted NACs. Among the VPC 2024 participants, three systems incorporated NACs into their submissions [89, 91, 93], with one being further developed after the challenge [160] and achieving strong privacy protection ($> 40\%$ EER).

At the time of its development, system B4 was assembled using the few open-weight modules available in the field of NAC-based TTS. As of today, this research area has flourished: several open-weight NACs [92, 161, 162, 163, 164, 165] and TTS systems powered by NAC language

[2] https://github.com/suno-ai/bark



(a) VPC 2022 benchmark: comparison with B1.b and T11.

(b) VPC 2024 benchmark: comparison with B3 and B5.

Figure 6.3: Performance summary of B4 and B4* compared to other relevant systems on the 2022 and 2024 VPC benchmarks.

modeling [21, 22, 166, 167] are now publicly available as foundations for further work. Some even offer VC capabilities off the shelf [168, 169] — a feature that was virtually unavailable when B4 was developed. Moreover, LLMs [170] have become increasingly relevant in the field of speech processing, with Large Audio Models (LAMs) emerging as a central research direction [171, 172, 173, 174]. All of these tools are potential building blocks for more effective NAC-based VA.

In particular, a relevant research question is how large models intrinsically affect the suppression of speaker-specific information. In Chapter 3, we showed that a vocoder can contribute to the overall privacy protection provided by a VA system. Does the same hold for a large auto-regressive Transformer? Does scale translate into a more effective VC, and therefore to a stronger suppression of speaker-specific voice cues? The question becomes more relevant when considering the phenomenon of *emergent abilities* in LLMs, where a large model appears to 'spontaneously' improve on a set of downstream tasks simply by virtue of scale [175]. Can the same behavior be observed in LAMs? If so, can auto-regressive foundation audio models learn spontaneously to anonymize voice? Some research on the matter has been conducted [176] with ASR models like Whisper [177], showing promising zero-shot results on audio classification taks such as sound event classification and acoustic scene classification. However, to the best of our knowledge, analogous research has not been conducted on privacy-related downstream tasks.

Lastly, other potential research directions pertain to the discrete nature of the speech representations used in B4. The original system uses a semantic token extractor not specifically designed for privacy-related downstream tasks. Future work could involve developing privacy-preserving semantic tokens. Several works in VA align with this direction: [91, 160] remove speaker-specific information from tokenized speech representations by subtracting speaker embeddings; [89, 93] attempt to exploit the factorized structure of the FACodec NAC [92]. Futhermore, the SpeechTokenizer NAC [163] is trained to concentrate semantic information in its first codebook layer, though, to the best of our knowledge, its use in VA remains unexplored. Even without privacy-specific optimization schemes, one could attempt to identify the NAC's codewords that contain speaker-specific information at inference time and suppress them. Exploiting the variable-bandwidth nature of some NACs is another promising direction. Using a lower bandwidth (typically involving fewer layers of the hierarchical codebooks) may reduce the amount of speaker-specific information leaking into the final anonymized waveform, though at the likely cost of reduced content preservation. Bandwidth could therefore act as a dynamic parameter for adjusting the privacy–utility tradeoff in real time, according to the user's needs.

With B4*, we introduced the character-level conditioning mechanism as a means to re-inject content information into the decoding process of B4. By now, the idea of "conditioning the synthesis on something necessary for the downstream task at hand" is somewhat established within the VA community: most notably, the prosody extraction branch of B3 [83] was designed to address the low $\boldsymbol{\rho}^{F_0}$ value that its predecessor, system T04 [65], exhibited on the VPC 2022 benchmark. More recently, system T09 of the VPC 2024 applied a similar technique to emotion preservation, conditioning the vocoder on an emotion feature vector [102]. As in character-level conditioning, the vocoder is trained so that the same emotion feature vector can be re-extracted from the anonymized utterance. Taken together, these findings highlight a broader design principle: knowing the downstream task in advance is essential for building effective VA systems. By incorporating task-specific feature extraction modules and reconstruction losses, one can ensure that the anonymized signal preserves the attributes most relevant to the intended application.

Chapter 5 - The role of the attacker

We described the phenomenon of privacy overestimation and implemented practical examples to illustrate it. Privacy overestimation occurs when the attacking system $\mathrm{ASV}^{\mathrm{S}}$ is trained suboptimally for the kind of data it is attempting to re-identify. We first focused on the case where suboptimal training is due to a distribution mismatch between the training data T^{S_1} of $\mathrm{ASV}^{\mathrm{S}_1}$ and the evaluation data E^{S_2}. This scenario was simulated by using completely different VA systems S_1 and S_2 for the generation of the training and evaluation data (*full mismatch*). We found that this condition led to a marked increase in EER, and hence to an overestimation of privacy protection. A similar but less pronounced overestimation was observed when S_1 and S_2 are in a condition of *partial mismatch*, i.e. where the two VA systems only differed by a single module.

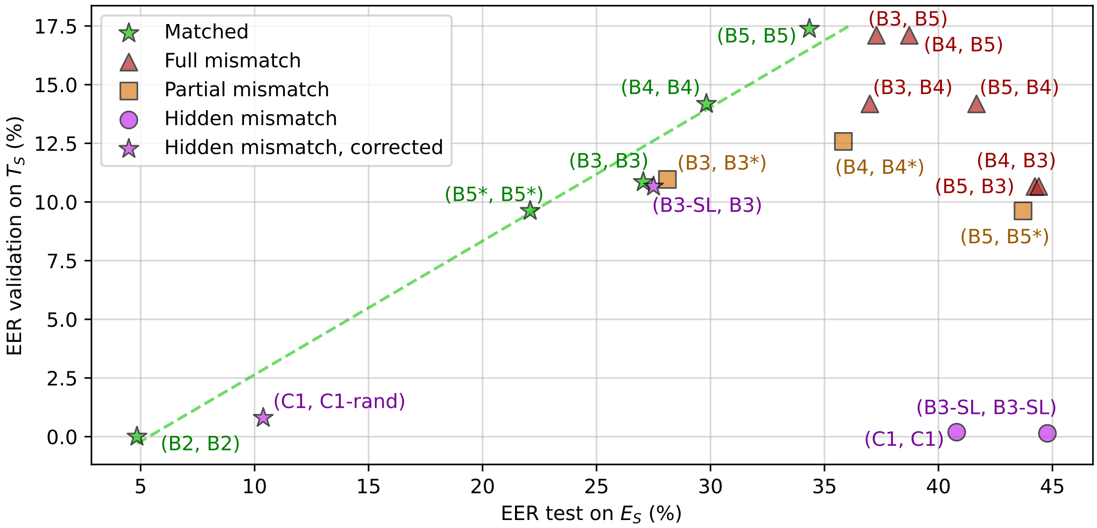


Figure 6.4: Values of EER_{val} plotted against EER_{test} for all considered systems, denoted as (E^{S}, T^{S}) pairs. The green dashed line is the regression line of the *"Matched"* systems. Systems falling below this line have a potentially mismatched evaluation. Reproduced from Section 5.1.2.

First and foremost, these experiments were designed to shed light on a known practical issue in the field of VA, which had received little to no attention in the literature, perhaps because of its seemingly trivial nature: technical errors in privacy evaluation are easy to make. A mistake in data generation can greatly inflate EER levels and give a false sense of protection. The community needs to be aware of this risk and remain cautious about non-reproducible results presented in the literature. Moreover, these experiments also suggest that, since an ASV system trained on T^{S_1} does not appear to learn useful cues to de-identify E^{S_2}, mixing training data from different anonymization systems to obtain a stronger attack may prove inconclusive or even detrimental. Additional support for this observation is provided in [178], where adopting a similar strategy for the attacking ASV system did not lead to meaningful improvements.

Our investigation of *hidden mismatches* reported in Section 5.1.1 showed that privacy can be overestimated even when the attacker uses the same system as the defender, in accordance with the semi-informed paradigm. That is because the mismatch is *hidden* within the system, since the training data it generates is unsuitable for an ASV model to learn reliable cues to re-identify the evaluation data (hence the *mismatch*).

In seeking a solution, we proposed a method to detect potential overestimation arising from data mismatch, based on the computation of a validation EER computed using the training data. The underlying principle is straightforward: a large gap between test and validation EERs serves as evidence of overfitting, which in turn indicates sub-optimal ASV training and, consequently, potential overestimation. The proposed approach was able to identify most of the

Table 6.2: Summary of proactive protection with adversarial noise on system B5. Results are shown for privacy protection (EER) and quality preservation (WER on unprotected enrollment utterance before and after the application of adversarial noise, SNR). Results reproduced from Section 5.2.3.

Dataset	**EER, %** **(original)**	**EER, %** **(adv. noise + padding)**	**WER, %** **(original)**	**WER, %** **(adv. noise + padding)**	**SNR_{dB}**
Libri-dev	34.37	48.48	1.73	2.46	28.25
Libri-test	34.34	48.23	1.83	2.62	27.28

described cases of mismatch, as shown in the scatterplot in Figure 6.4. However, it has a number of shortcomings: it relies on "good" ground truth cases as reference points for $\text{EER}_{\text{val}}/\text{EER}_{\text{test}}$ (for which we used the VPC 2024 baselines B2 through B6), and it remains qualitative, unable to pinpoint the exact source of potential overestimation.

This is just one manifestation of a more general issue in VA: because there is no "optimal" attacker, there is also no optimal privacy evaluation method. While alternative privacy metrics to the EER — sometimes argued to be more suitable — have been proposed in the past [179] and are still being investigated today [180,181], they nonetheless rely on some speaker representation to be computed. Therefore, the problem persists. A potential direction for future work would be to extend the proposed method by using systems operating on unprotected data as the ground-truth reference, rather than relying on "well-evaluated systems" such as the VPC baselines. The rationale is that the ultimate goal of an attacker is to re-identify speakers as reliably as if the data were not anonymized at all; hence, the true gold standard for assessing attack optimality is the performance of conventional ASV systems on unprotected speech.

It should be noted that suboptimal training of the attacking ASV system can lead to privacy overestimation when done unintentionally, since a better-designed attack would yield a lower EER. However, deliberately hindering a potential attacking ASV system can be a legitimate proactive defense. We investigated this approach in the second part of Chapter 5. To mount a strong attack, the adversary must anonymize unprotected enrollment utterances A with a VA system similar to the defender's. Based on this insight, we designed a countermeasure that the defender can apply to unprotected data that may be exposed publicly. The countermeasure adds an adversarial perturbation to the utterance that disrupts the synthesis performed by the same VA system used to anonymize the trial data. When the attacker anonymizes enrollment utterances perturbed with this noise, the resulting audio is highly degraded and unsuitable for enrollment, producing high EERs. We implemented the attack on B5, targeting the intermediate quantized embeddings and forcing codeword changes at each time step. The adversarial noise is optimized with projected gradient descent (PGD). The results are summarized in Table 6.2:

the attacker's EER was raised to nearly 50% (see third column), at the cost of a small increase in WER (compare fourth and fifth columns). Although the adversarial noise is faintly audible, it does not materially reduce the signal's utility or intelligibility.

This is the last contribution of this dissertation, and arguably that which creates the greatest opportunities for further work. It was inspired by the rising popularity of proactive defenses in speech security tasks such as anti-spoofing. Such defense techniques are usually based on watermarking in the case of source verification [182, 183, 184] or adversarial noise in the case of voice cloning prevention [30, 31]. Our approach is closer to the latter and constitutes, to the best of our knowledge, the first attempt to actively disrupt the attacker in VA.

This naturally leaves many aspects open for deeper investigation. Apart from the obvious extension to more VA and ASV systems, further issues typically associated with adversarial attacks should also be addressed. First, the adversarial noise δ is optimized individually for each utterance, which can be computationally expensive. One potential remedy is to adopt *universal* perturbations. Prior research (including the candidate's [145]) has shown that it is possible to craft *universal* perturbations that work across different utterances [149]; future work should consider exploring this direction. Secondly, the adversarial noise could be made less audible using frequency-masking techniques [185]. This is important because these perturbations are applied to unprotected utterances that will often be consumed by humans. Last, the attacker might attempt to remove the adversarial noise using adversarial purification methods (e.g., denoising or purification techniques [186]). The robustness of our approach to such countermeasures remains to be evaluated.

Ethical considerations

In Section 1.1.2, we highlighted the increasing concern around voice deepfakes, especially regarding the ease of access to deepfake generation models by non-expert users. One could argue that VA systems and deepfake generation systems share the same underlying technologies: VC and TTS. This raises an ethical question: by researching and developing VA systems, are we also facilitating the spread of deepfake technologies? Could VA models be repurposed or modified for malicious use, such as voice cloning?

While the answer clearly depends on the VA system in question, it is safe to say that it is, in most cases, affirmative: modules and other components of VA systems can indeed be adapted to assemble voice cloning systems. In fact, in Section 3.7, we argued that VA research should focus on effective VC rather than pseudo-speaker selection — an approach that incidentally mirrors the objectives of an attacker aiming to reproduce a specific speaker's voice. Therefore, developers of VA systems must also consider how to mitigate the risk of their misuse.

The most straightforward mitigation is simply not to release the VA system publicly. This is likely if the system is proprietary; while the most obvious motivation for this choice is the protection of intellectual property, keeping the VA model undisclosed also substantially reduces the risk of misuse. Additionally, it makes it much harder to carry out a semi-informed or lazy-informed attack as described in Section 2.1.2, as the privacy adversary lacks access to the model needed to generate the anonymized data A^{VA} and T^{VA} and consequently adapt the ASV system. While the semi-informed attack has some shortcomings — as we showed in Section 5.1 — it is still a serious threat, and mitigating it remains highly beneficial for the overall security of the VA system.

However, keeping the VA system undisclosed has important downsides. First, it prevents the community from transparently reproducing and validating the system: without access to the implementation or model weights, independent researchers cannot verify the claimed anonymization performance (as we did in Section 5.1.1 with system C1) or detect potential flaws in the system (such as the subgroup biases described by Leschanowsky et al. in [187]). Note that the EU AI Act encourages transparency for general-purpose AI models: under Article 53(a), providers must maintain technical documentation describing the model's capabilities and training data, and make it available upon request to the competent supervisory authorities [25]. While this transparency is directed at regulators rather than the public, it reflects a broader principle that openness can contribute to making a VA system more trustworthy.

Moreover, relying on secrecy as a protective measure (an approach known as "security by obscurity") is regarded as a poor practice in the field of information security. The idea dates back to Kerckhoffs's principle in cryptography, which states that a system should remain secure even if everything about it, except the secret key, is public [188]. By analogy, a VA system should be designed so that its privacy guarantees hold even if its architecture and algorithms are known to the attacker.

If the VA system is made available publicly, active countermeasures must be taken to avoid misuse. The specific threat of voice cloning can be mitigated by restricting the number of pseudo-speakers supported by the system. This strategy was implemented in B5, which uses one-hot representations to encode voice identities (see Section 2.4.3). Training the system on a single voice would be another possible, albeit more extreme, solution. While this approach prevents the use of arbitrary voices in the VA system, it also limits its flexibility.

An interesting example is that of Bark, the TTS system on which B4 was based. As detailed in Section 4.2.3, Suno AI, the original developers of Bark, provided voice prompts for only a limited number of speakers. Thus, in practice, the TTS capabilities of Bark are limited to those voices. Although this choice is not explicitly justified in the official documentation, it likely aimed to prevent misuse of Bark for voice cloning.[3] This solution represents a middle ground between pseudo-speaker restriction and security by obscurity: while it *is* technically possible to use arbitrary voices in Bark, Suno AI did not release the semantic encoder required to generate custom speaker prompts, effectively limiting the speaker voices to those released with the system. However, as explained in Section 4.2.3, the *CoquiTTS* community later trained a separate semantic encoder to generate arbitrary speaker prompts for Bark, enabling the use of custom TTS voices. This outcome highlights the limitations of relying on security by obscurity.

An alternative approach involves watermarking [182, 183, 184], whereby an imperceptible signature is embedded into a synthetic audio signal to indicate that it has been artificially generated. This solution does not actively prevent misuse of the VA system for voice cloning, but it enables the detection of such misuse. Its advantage is that it does not limit the capabilities of the system in any way; however, its main drawback is that the watermark can be removed — either by perturbing the generated utterance or by disabling the watermarking step altogether, if feasible.

Ultimately, as in many engineering problems, there is no universally best solution: it should be chosen based on the model specifications and the scenario at hand. What is important for the design of a VA system is to be aware of the risks of misuse, and to actively take an informed decision about their mitigation.

Final remarks

In this dissertation, we tackled the task of VA from several angles. The research work of the candidate was largely carried out between the VPC 2022 and 2024 editions, following the evolution of the field from one edition to the next.

The initial investigation on vocoder drift can be viewed as an analysis of the behavior of x-vector–based VA pipelines. That work, while delivering several concrete findings, contributed to the broader conclusion that the usual VA recipe — F0, anonymized x-vector and linguistic features fed into a vocoder — was becoming outdated and suffered from several shortcomings that needed to be addressed.

B4 was an attempt to seek an alternative solution, and received positive attention from the community. While the principle of NAC language modeling may still see future developments and improvements, it aligned with (and contributed to) the "new wave" of approaches that appeared after the VPC 2022, coinciding with an increase of interest in the topic.

[3] Nevertheless, the original README explicitly states (multiple times) that the system "does not support custom voice cloning": https://github.com/suno-ai/bark?tab=readme-ov-file#-faq.

With many new ideas for the role of the defender now emerging, the final part of this dissertation examined the other side of the problem — attackers and evaluation. Performance evaluation in VA has historically been problematic, mainly for two reasons. First, the absence of a precise target downstream task calls for broad, all-purpose quality metrics that are generic enough to cover a wide number of use cases. These metrics are, of course, challenging to define and validate. Second, privacy evaluation relies on the performance of an inherently imperfect attacker. As we detailed extensively, the attacking ASV training strategy can be sub-optimal; more powerful speaker representation backbones may be employed as they become available and change our perception of what systems are most secure; an ASV-based adversary may not even be the most suitable and realistic choice of threat model. With our work, we sought to gain insights into why privacy evaluation might be unreliable, proposed a means to detect such cases, and then turned those same insights into an active defense.

Still, a "gold standard" for privacy evaluation remains an open problem and an important research direction, alongside several other aspects of the task of VA. With this dissertation, the candidate has brought novel contributions not just in terms of VA techniques but also in terms of evaluation, and presents his vision for continued research in both directions.

Bibliography

[1] Nili Steinfeld. “I agree to the terms and conditions”: (How) do users read privacy policies online? An eye-tracking experiment. *Computers in Human Behavior*, 55:992–1000, 2016.

[2] Tom Bäckström and Fedor Vitiugin. Beyond User-centric: Modelling Privacy and Fairness Effects of Speech Interfaces on Community- and Society-Levels. In *5th Symposium on Security and Privacy in Speech Communication*, pages 13–17, 2025.

[3] Jenny T. Liang, Chenyang Yang, and Brad A. Myers. A Large-Scale Survey on the Usability of AI Programming Assistants: Successes and Challenges. In *Proceedings of the IEEE/ACM 46th International Conference on Software Engineering*, ICSE ’24, New York, NY, USA, 2024. Association for Computing Machinery.

[4] Capgemini. Voice on the go: How can auto companies provide a superior in-car voice experience?, 2019.

[5] Faruk Lawal Ibrahim Dutsinma, Debajyoti Pal, Suree Funilkul, and Jonathan H Chan. A systematic review of voice assistant usability: An ISO 9241-11 approach. *SN Comput Sci*, 3(4):267, May 2022.

[6] Google. Gemini for google assistant, 2024. Accessed online.

[7] Sudarsana Reddy Kadiri, Rashmi Kethireddy, and Paavo Alku. Parkinson’s disease detection from speech using single frequency filtering cepstral coefficients. In *Interspeech 2020*, pages 4971–4975, 2020.

[8] Jun Chen, Jieping Ye, Fengyi Tang, and Jiayu Zhou. Automatic detection of alzheimer’s disease using spontaneous speech only. In *Interspeech 2021*, pages 3830–3834, 2021.

[9] Maximilian Bauser, Fabian Kraus, Friedrich Koehler, Kristen Rak, Rüdiger Pryss, Christof Weiß, Andreas Hotho, Guy Fagherazzi, Stefan Frantz, Stefan Stoerk, and Fabian Kerwagen. Voice assessment and vocal biomarkers in heart failure: A systematic review. *Circulation: Heart Failure*, 18(8):e012303, 2025.

[10] Madhu R. Kamble, Jose Patino, Maria A. Zuluaga, and Massimiliano Todisco. Exploring auditory acoustic features for the diagnosis of covid-19. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 566–570, 2022.

[11] Muthu Selvam and R. Thalapathi Rajasekaran. Natural language processing for voice-based banking interactions: Enhancing user interaction through voice commands. In *2025 IEEE 4th International Conference on Computing and Machine Intelligence (ICMI)*, pages 1–6, 2025.

[12] BBC News. Hsbc introduces voice recognition and touch security, 2016. Accessed: 2025-09-30.

[13] Pindrop. How to combat return fraud, 2023. Accessed: 2025-09-30.

[14] Yi-Chin Huang and Cheng-Hung Tsai. Speech-based interface for visually impaired users. In *2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 1223–1228, 2018.

[15] IDC (via Box). The untapped value of unstructured data, 2024. Accessed: 2025-10-01.

[16] Jaesung Huh, Joon Son Chung, Arsha Nagrani, Andrew Brown, Jee-weon Jung, Daniel Garcia-Romero, and Andrew Zisserman. The voxceleb speaker recognition challenge: A retrospective. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:3850–3866, 2024.

[17] 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*, pages 1693–1697, 2020.

[18] Massimiliano Todisco, Xin Wang, Ville Vestman, Md. Sahidullah, Héctor Delgado, Andreas Nautsch, Junichi Yamagishi, Nicholas Evans, Tomi H. Kinnunen, and Kong Aik Lee. Asvspoof 2019: Future horizons in spoofed and fake audio detection. In *Interspeech 2019*, pages 1008–1012, 2019.

[19] Xin Wang, Héctor Delgado, Hemlata Tak, Jee weon Jung, Hye jin Shim, Massimiliano Todisco, Ivan Kukanov, Xuechen Liu, Md Sahidullah, Tomi H. Kinnunen, Nicholas Evans, Kong Aik Lee, and Junichi Yamagishi. Asvspoof 5: crowdsourced speech data, deepfakes, and adversarial attacks at scale. In *The Automatic Speaker Verification Spoofing Countermeasures Workshop (ASVspoof 2024)*, pages 1–8, 2024.

[20] Nicolas M. Müller, Karla Pizzi, and Jennifer Williams. Human perception of audio deepfakes. In *Proceedings of the 1st International Workshop on Deepfake Detection for Audio Multimedia*, DDAM '22, page 85–91, New York, NY, USA, 2022. Association for Computing Machinery.

[21] Zhihao Du, Changfeng Gao, Yuxuan Wang, Fan Yu, Tianyu Zhao, Hao Wang, Xiang Lv, Hui Wang, Xian Shi, Keyu An, et al. CosyVoice 3: Towards In-the-wild Speech Generation via Scaling-up and Post-training. *arXiv preprint arXiv:2505.17589*, 2025.

[22] Xin Li, Kaikai Jia, Hao Sun, Jun Dai, and Ziyang Jiang. Muyan-TTS: A Trainable Text-to-Speech Model Optimized for Podcast Scenarios with a $50 K Budget. *arXiv preprint arXiv:2504.19146*, 2025.

[23] Andreas Nautsch, Catherine Jasserand, Els Kindt, Massimiliano Todisco, Isabel Trancoso, and Nicholas Evans. The GDPR & Speech Data: Reflections of Legal and Technology Communities, First Steps Towards a Common Understanding. In *Interspeech 2019*, pages 3695–3699, 2019.

[24] European Parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council.

[25] Regulation (eu) 2024/1689 of the european parliament and of the council of 13 june 2024 on artificial intelligence (artificial intelligence act), 2024. OJ L, 2024/1689.

[26] Cynthia Dwork. Differential privacy. In Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener, editors, *Automata, Languages and Programming*, pages 1–12, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[27] Natalia Tomashenko, Xin Wang, Xiaoxiao Miao, Hubert Nourtel, Pierre Champion, Massimiliano Todisco, Emmanuel Vincent, Nicholas Evans, Junichi Yamagishi, and Jean-François Bonastre. The VoicePrivacy 2022 Challenge Evaluation Plan, 2022.

[28] Sarina Meyer and Ngoc Thang Vu. Use Cases for Voice Anonymization. In *5th Symposium on Security and Privacy in Speech Communication*, pages 73–84, 2025.

[29] 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*, 29:132–157, 2021.

[30] Rui Wang, Liping Chen, Kong Aik Lee, and Zhen-Hua Ling. Asynchronous Voice Anonymization Using Adversarial Perturbation On Speaker Embedding. In *Interspeech 2024*, pages 4443–4447, 2024.

[31] Seoyoung Park, Thien-Phuc Doan, and Souhwan Jung. An Imperceptible Adversarial Watermarking to Prevent Voice Cloning. In *5th Symposium on Security and Privacy in Speech Communication*, pages 56–60, 2025.

[32] Felix Kreuk, Yossi Adi, Moustapha Cisse, and Joseph Keshet. Fooling End-To-End Speaker Verification With Adversarial Examples. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1962–1966, 2018.

[33] Guangke Chen, Sen Chenb, Lingling Fan, Xiaoning Du, Zhe Zhao, Fu Song, and Yang Liu. Who is Real Bob? Adversarial Attacks on Speaker Recognition Systems. In *2021 IEEE Symposium on Security and Privacy (SP)*, pages 694–711, 2021.

[34] Massimiliano Todisco, Michele Panariello, Xin Wang, Héctor Delgado, Kong Aik Lee, and Nicholas Evans. Malacopula: adversarial automatic speaker verification attacks using a neural-based generalised Hammerstein model. In *The Automatic Speaker Verification Spoofing Countermeasures Workshop (ASVspoof 2024)*, pages 94–100, 2024.

[35] 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. *Comput. Speech Lang.*, 74(C), July 2022.

[36] 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 *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2802–2806, 2020.

[37] Natalia Tomashenko, Xiaoxiao Miao, Pierre Champion, Sarina Meyer, Xin Wang, Emmanuel Vincent, Michele Panariello, Nicholas Evans, Junichi Yamagishi, and Massimiliano Todisco. The VoicePrivacy 2024 Challenge Evaluation Plan, 2024.

[38] Pierre Champion. *Anonymizing Speech: Evaluating and Designing Speaker Anonymization Techniques.* PhD thesis, Université de Lorraine, 2023.

[39] Pierre Champion, Denis Jouvet, and Anthony Larcher. Evaluating x-vector-based speaker anonymization under white-box assessment. In Alexey Karpov and Rodmonga Potapova, editors, *Speech and Computer*, pages 100–111, Cham, 2021. Springer International Publishing.

[40] Brij Mohan Lal Srivastava. *Speaker anonymization : representation, evaluation and formal guarantees.* PhD thesis, Université de Lille, 2021.

[41] Natalia Tomashenko, Xiaoxiao Miao, Emmanuel Vincent, and Junichi Yamagishi. The First VoicePrivacy Attacker Challenge Evaluation Plan, 2024.

[42] Michele Panariello, Natalia Tomashenko, Xin Wang, Xiaoxiao Miao, Pierre Champion, Hubert Nourtel, Massimiliano Todisco, Nicholas Evans, Emmanuel Vincent, and Junichi Yamagishi. The VoicePrivacy 2022 Challenge: Progress and Perspectives in Voice Anonymisation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:3477–3491, 2024.

[43] Arsha Nagrani, Joon Son Chung, and Andrew Zisserman. VoxCeleb: A Large-Scale Speaker Identification Dataset. In *Interspeech 2017*, pages 2616–2620, 2017.

[44] Joon Son Chung, Arsha Nagrani, and Andrew Zisserman. VoxCeleb2: Deep Speaker Recognition. In *Interspeech 2018*, pages 1086–1090, 2018.

[45] Arsha Nagrani, Joon Son Chung, Weidi Xie, and Andrew Zisserman. Voxceleb: Large-scale speaker verification in the wild. *Computer Speech & Language*, 60:101027, 2020.

[46] 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, 2015.

[47] 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. In *Interspeech 2019*, pages 1526–1530, 2019.

[48] Christophe Veaux, Junichi Yamagishi, and Kirsten MacDonald. CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit (version 0.92). https://datashare.is.ed.ac.uk/handle/10283/3443, 2019.

[49] 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)*, pages 5329–5333, 2018.

[50] Sergey Ioffe. Probabilistic linear discriminant analysis. In Aleš Leonardis, Horst Bischof, and Axel Pinz, editors, *Computer Vision – ECCV 2006*, pages 531–542, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[51] Daniel Povey, Gaofeng Cheng, Yiming Wang, Ke Li, Hainan Xu, Mahsa Yarmohammadi, and Sanjeev Khudanpur. Semi-Orthogonal Low-Rank Matrix Factorization for Deep Neural Networks. In *Interspeech 2018*, pages 3743–3747, 2018.

[52] Paul-Gauthier Noé, Jean-François Bonastre, Driss Matrouf, N. Tomashenko, Andreas Nautsch, and Nicholas Evans. Speech Pseudonymisation Assessment Using Voice Similarity Matrices. In *Interspeech 2020*, pages 1718–1722, 2020.

[53] Ali Shahin Shamsabadi, Brij Mohan Lal Srivastava, Aurélien Bellet, Nathalie Vauquier, Emmanuel Vincent, Mohamed Maouche, Marc Tommasi, and Nicolas Papernot. Differentially Private Speaker Anonymization. *Proceedings on Privacy Enhancing Technologies*, 2023.

[54] Michele Panariello, Sarina Meyer, Pierre Champion, Xiaoxiao Miao, Massimiliano Todisco, Ngoc Thang Vu, and Nicholas Evans. The Risks and Detection of Overestimated Privacy Protection in Voice Anonymisation. In *5th Symposium on Security and Privacy in Speech Communication*, pages 8–12, 2025.

[55] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, et al. The Kaldi speech recognition toolkit. 2011.

[56] Paul-Gauthier Noé, Andreas Nautsch, Nicholas Evans, Jose Patino, Jean-François Bonastre, Natalia Tomashenko, and Driss Matrouf. Towards a unified assessment framework of speech pseudonymisation. *Computer Speech & Language*, 72:101299, 2022.

[57] Daniel Hirst. A Praat plugin for Momel and INTSINT with improved algorithms for modelling and coding intonation. ICPhS XVI, Saabrücken, 2007.

[58] Andrew Rosenberg and Bhuvana Ramabhadran. Bias and Statistical Significance in Evaluating Speech Synthesis with Mean Opinion Scores. In *INTERSPEECH*, pages 3976–3980, 2017.

[59] Fuming Fang, Xin Wang, Junichi Yamagishi, Isao Echizen, Massimiliano Todisco, Nicholas Evans, and Jean-Francois Bonastre. Speaker Anonymization Using X-vector and Neural Waveform Models. In *Speech Synthesis Workshop*, pages 155–160, 2019.

[60] Kavita Kasi and Stephen A. Zahorian. Yet Another Algorithm for Pitch Tracking. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages I–361–I–364, May 2002. ISSN: 1520-6149.

[61] Xin Wang, Shinji Takaki, and Junichi Yamagishi. Neural source-filter-based waveform model for statistical parametric speech synthesis. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5916–5920, 2019.

[62] 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*, pages 1713–1717, 2020.

[63] Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. Hifi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis. *Advances in Neural Information Processing Systems*, 33:17022–17033, 2020.

[64] Jose Patino, Natalia Tomashenko, Massimiliano Todisco, Andreas Nautsch, and Nicholas Evans. Speaker anonymisation using the McAdams coefficient. In *INTERSPEECH*, pages 1099–1103, 2021.

[65] Sarina Meyer, Pascal Tilli, Florian Lux, Pavel Denisov, Julia Koch, and Ngoc Thang Vu. Cascade of phonetic speech recognition, speaker embeddings gan and multispeaker speech synthesis for the VoicePrivacy 2022 Challenge . In *Proc. 2nd Symposium on Security and Privacy in Speech Communication*, 2022.

[66] 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.

[67] 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. In *9th International Conference on Learning Representations, ICLR, Virtual Event, Austria, May 3-7*. OpenReview.net, 2021.

[68] Florian Lux and Thang Vu. Language-agnostic meta-learning for low-resource text-to-speech with articulatory features. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6858–6868, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[69] Hyeong-Seok Choi, Juheon Lee, Wansoo Kim, Jie Lee, Hoon Heo, and Kyogu Lee. Neural Analysis and Synthesis: Reconstructing Speech from Self-Supervised Representations. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 16251–16265. Curran Associates, Inc., 2021.

[70] Zhuoyuan Yao, Di Wu, Xiong Wang, Binbin Zhang, Fan Yu, Chao Yang, Zhendong Peng, Xiaoyu Chen, Lei Xie, and Xin Lei. WeNet: Production Oriented Streaming and Non-Streaming End-to-End Speech Recognition Toolkit. In *INTERSPEECH*, pages 4054–4058, 2021.

[71] Eric Moulines and Francis Charpentier. Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones. *Speech communication*, 9(5-6):453–467, 1990.

[72] J. Laroche and M. Dolson. Improved phase vocoder time-scale modification of audio. *IEEE Transactions on Speech and Audio Processing*, 7(3):323–332, 1999.

[73] Brecht Desplanques, Jenthe Thienpondt, and Kris Demuynck. ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification. In *Interspeech 2020*, pages 3830–3834, 2020.

[74] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Commun. ACM*, 63(11):139–144, October 2020.

[75] Xiaoxiao Miao, Xin Wang, Erica Cooper, Junichi Yamagishi, and Natalia Tomashenko. Language-Independent Speaker Anonymization Approach Using Self-Supervised Pre-Trained Models. In *The Speaker and Language Recognition Workshop (Odyssey 2022)*, pages 279–286, 2022.

[76] Pierre Champion, Anthony Larcher, and Denis Jouvet. Are disentangled representations all you need to build speaker anonymization systems? In *Interspeech 2022*, pages 2793–2797, 2022.

[77] Xiaoxiao Miao, Xin Wang, Erica Cooper, Junichi Yamagishi, and Natalia Tomashenko. Speaker Anonymization Using Orthogonal Householder Neural Network. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:3681–3695, 2023.

[78] Carlos Busso, Murtaza Bulut, Chi-Chun Lee, Abe Kazemzadeh, Emily Mower, Samuel Kim, Jeannette N. Chang, Sungbok Lee, and Shrikanth S. Narayanan. IEMOCAP: Interactive emotional dyadic motion capture database. *Journal of Language Resources and Evaluation*, 42:335–359, 2008.

[79] Mirco Ravanelli, Titouan Parcollet, Peter Plantinga, Aku Rouhe, Samuele Cornell, Loren Lugosch, Cem Subakan, Nauman Dawalatabad, Abdelwahab Heba, Jianyuan Zhong, Ju-Chieh Chou, Sung-Lin Yeh, Szu-Wei Fu, Chien-Feng Liao, Elena Rastorgueva, François Grondin, William Aris, Hwidong Na, Yan Gao, Renato De Mori, and Yoshua Bengio. SpeechBrain: A general-purpose speech toolkit, 2021. arXiv:2106.04624.

[80] Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: a framework for self-supervised learning of speech representations. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS ’20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[81] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML ’06, page 369–376, New York, NY, USA, 2006. Association for Computing Machinery.

[82] J. Kahn, M. Rivière, W. Zheng, E. Kharitonov, Q. Xu, P. E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen, T. Likhomanenko, G. Synnaeve, A. Joulin, A. Mohamed, and E. Dupoux. Libri-Light: A Benchmark for ASR with Limited or No Supervision. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7669–7673, 2020. https://github.com/facebookresearch/libri-light.

[83] Sarina Meyer, Florian Lux, Julia Koch, Pavel Denisov, Pascal Tilli, and Ngoc Thang Vu. Prosody Is Not Identity: A Speaker Anonymization Approach Using Prosody Cloning. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[84] Yifan Peng, Siddharth Dalmia, Ian Lane, and Shinji Watanabe. Branchformer: Parallel MLP-attention architectures to capture local and global context for speech recognition and understanding. In *International Conference on Machine Learning (ICML)*, pages 17627–17643, 2022.

[85] Yuxuan Wang, Daisy Stanton, Yu Zhang, RJ-Skerry Ryan, Eric Battenberg, Joel Shor, Ying Xiao, Fei Ren, Ye Jia, and Rif A. Saurous. Style tokens: Unsupervised style modeling, control and transfer in end-to-end speech synthesis. In *International Conference on Machine Learning (ICML)*, pages 5180–5189, 2018.

[86] Sarina Meyer, Pascal Tilli, Pavel Denisov, Florian Lux, Julia Koch, and Ngoc Thang Vu. Anonymizing Speech with Generative Adversarial Networks to Preserve Speaker Privacy. In *2022 IEEE Spoken Language Technology Workshop (SLT)*, pages 912–919, 2023.

[87] Michele Panariello, Francesco Nespoli, Massimiliano Todisco, and Nicholas Evans. Speaker Anonymization Using Neural Audio Codec Language Models. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4725–4729, 2024.

[88] Henry Li Xinyuan, Zexin Cai, Ashi Garg, Kevin Duh, Leibny Paola García-Perera, Sanjeev Khudanpur, Nicholas Andrews, and Matthew Wiesner. HLTCOE JHU Submission to the Voice Privacy Challenge 2024. In *4th Symposium on Security and Privacy in Speech Communication*, pages 61–66, 2024.

[89] Nikita Kuzmin, Hieu-Thi Luong, Jixun Yao, Lei Xie, Kong Aik Lee, and Eng-Siong Chng. NTU-NPU System for Voice Privacy 2024 Challenge. In *4th Symposium on Security and Privacy in Speech Communication*, pages 72–79, 2024.

[90] Jeongae Lee, Taeje Park, and Yeawon You. Voice Anonymization Using Emotion-Enriched Feature Integration with STT and TTS Models. In *4th Symposium on Security and Privacy in Speech Communication*, pages 50–54, 2024.

[91] Jixun Yao, Nikita Kuzmin, Qing Wang, Pengcheng Guo, Ziqian Ning, Dake Guo, Kong Aik Lee, Eng-Siong Chng, and Lei Xie. NPU-NTU System for Voice Privacy 2024 Challenge. In *4th Symposium on Security and Privacy in Speech Communication*, pages 67–71, 2024.

[92] Zeqian Ju, Yuancheng Wang, Kai Shen, Xu Tan, Detai Xin, Dongchao Yang, Yanqing Liu, Yichong Leng, Kaitao Song, Siliang Tang, et al. NaturalSpeech 3: Zero-Shot Speech Synthesis with Factorized Codec and Diffusion Models. *arXiv preprint arXiv:2403.03100*, 2024.

[93] Jiabei He, Jiaming Zhou, Haoqin Sun, Hui Wang, and Yong Qin. FaCodec-based Anonymization Solution Enhanced with Prosody Anonymization, 2025.

[94] Olivier Le Blouch, Rayane BAKARI, and Nicolas Gengembre. Tuning DISSC for Voice Privacy Challenge 2024. In *4th Symposium on Security and Privacy in Speech Communication*, pages 111–115, 2024.

[95] Gallil Maimon and Yossi Adi. Speaking style conversion in the waveform domain using discrete self-supervised units. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8048–8061, Singapore, December 2023. Association for Computational Linguistics.

[96] Sotheara Leang, Anderson Augusma, Dominique Vaufreydaz, Eric Castelli, Sethserey Sam, and Frédérique Letué. Exploring VQ-VAE with Prosody Parameters for Speaker Anonymization. In *4th Symposium on Security and Privacy in Speech Communication*, pages 127–131, 2024.

[97] Arnab Das, Carlos Franzreb, Tim Herzig, Philipp Pirlet, and Tim Polzehl. Comparing Speech Anonymization Efficacy by Voice Conversion Using KNN and Disentangled Speaker Feature Representations. In *4th Symposium on Security and Privacy in Speech Communication*, pages 121–126, 2024.

[98] Disong Wang, Liqun Deng, Yu Ting Yeung, Xiao Chen, Xunying Liu, and Helen Meng. VQMIVC: Vector Quantization and Mutual Information-Based Unsupervised Speech Representation Disentanglement for One-Shot Voice Conversion. In *Interspeech 2021*, pages 1344–1348, 2021.

[99] Wenju Gu, Zeyan Liu, Liping Chen, Rui Wang, Chenyang Guo, Wu Guo, Kong Aik Lee, and Zhen-Hua Ling. A Voice Anonymization Method Based on Content and Non-content Disentanglement for Emotion Preservation. In *4th Symposium on Security and Privacy in Speech Communication*, pages 116–120, 2024.

[100] Matthew Baas, Benjamin van Niekerk, and Herman Kamper. Voice Conversion With Just Nearest Neighbors. In *Interspeech 2023*, pages 2053–2057, 2023.

[101] Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. WavLM: Large-Scale Self-Supervised Pre-Training for Full Stack Speech Processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518, 2022.

[102] Tao Tan, Shutao Liu, Yibo Duan, Sheng Zhao, and Xi Shao. System description: Speaker anonymization system with sentiment transfer and feature interpolation. *voiceprivacy-challenge. org*, 2024.

[103] Hua Hua, Zengqiang Shang, Xuyuan Li, Peiyang Shi, Chen Yang, Li Wang, and Pengyuan Zhang. Emotional speech anonymization: Preserving emotion characteristics in pseudo-speaker speech generation. In *4th Symposium on Security and Privacy in Speech Communication*, pages 55–60, 2024.

[104] Seymanur Akti, Tuan Nam Nguyen, Yining Liu, and Alex Waibel. Voice Privacy - Investigating Voice Conversion Architecture with Different Bottleneck Features. In *4th Symposium on Security and Privacy in Speech Communication*, pages 44–49, 2024.

[105] Jingyi Li, Weiping Tu, and Li Xiao. Freevc: Towards High-Quality Text-Free One-Shot Voice Conversion. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5, 2023.

[106] Candy Olivia Mawalim, Kasorn Galajit, Jessada Karnjana, and Masashi Unoki. X-Vector Singular Value Modification and Statistical-Based Decomposition with Ensemble Regression Modeling for Speaker Anonymization System. In *Interspeech 2020*, pages 1703–1707, 2020.

[107] Pierre Champion, Denis Jouvet, and Anthony Larcher. Speaker information modification in the VoicePrivacy 2020 toolchain. Research report, INRIA Nancy, équipe Multispeech ; LIUM - Laboratoire d'Informatique de l'Université du Mans, November 2020.

[108] Henry Turner, Giulio Lovisotto, and Ivan Martinovic. Speaker Anonymization with Distribution-Preserving X-Vector Generation for the VoicePrivacy Challenge 2020, 2021.

[109] Fernando M. Espinoza-Cuadros, Juan M. Perero-Codosero, Javier Antón-Martín, and Luis A. Hernández-Gómez. Speaker De-identification System using Autoencoders and Adversarial Training, 2020.

[110] Jixun Yao, Qing Wang, Li Zhang, Pengcheng Guo, Yuhao Liang, and Lei Xie. NWPU-ASLP System for the VoicePrivacy 2022 Challenge. In *Proc. 2nd Symposium on Security and Privacy in Speech Communication*, 2022.

[111] Xiaojiao Chen, Guangxing Li, Hao Huang, Wangjin Zhou, Sheng Li, Yang Cao, and Yi Zhao. System description for Voice Privacy Challenge 2022 . In *Proc. 2nd Symposium on Security and Privacy in Speech Communication*, 2022.

[112] Liping Chen, Kong Aik Lee, Wu Guo, and Zhen-Hua Ling. Modeling Pseudo-Speaker Uncertainty in Voice Anonymization. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11601–11605, 2024.

[113] Pierre Champion, Thomas Thebaud, Gaël Le Lan, Anthony Larcher, and Denis Jouvet. On the Invertibility of a Voice Privacy System Using Embedding Alignment. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 191–197, 2021.

[114] Brij Mohan Lal Srivastava, Mohamed Maouche, Md Sahidullah, Emmanuel Vincent, Aurélien Bellet, Marc Tommasi, Natalia Tomashenko, Xin Wang, and Junichi Yamagishi. Privacy and utility of x-vector based speaker anonymization. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, June 2022.

[115] Unal Ege Gaznepoglu, Anna Leschanowsky, and Nils Peters. VoicePrivacy 2022 system description: speaker anonymization with feature-matched f0 trajectories. In *Proc. 2nd Symposium on Security and Privacy in Speech Communication*, 2022.

[116] Michele Panariello, Massimiliano Todisco, and Nicholas Evans. Vocoder drift in x-vector–based speaker anonymization. In *Interspeech 2023*, pages 2863–2867, 2023.

[117] Benjamin van Niekerk, Marc-André Carbonneau, Julian Zaïdi, Matthew Baas, Hugo Seuté, and Herman Kamper. A Comparison of Discrete and Soft Speech Units for Improved Voice Conversion. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6562–6566, 2022.

[118] Laurens van der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

[119] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[120] Xu Xiang, Shuai Wang, Houjun Huang, Yanmin Qian, and Kai Yu. Margin Matters: Towards More Discriminative Deep Neural Network Embeddings for Speaker Recognition. In *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1652–1656, 2019.

[121] Michele Panariello, Massimiliano Todisco, and Nicholas Evans. Vocoder drift compensation by x-vector alignment in speaker anonymisation. In *3rd Symposium on Security and Privacy in Speech Communication*, pages 16–20, 2023.

[122] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 6309–6318. Curran Associates Inc.

[123] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund, and Marco Tagliasacchi. SoundStream: An End-to-End Neural Audio Codec. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 30:495–507, November 2021.

[124] Alexandre Défossez, Jade Copet, Gabriel Synnaeve, and Yossi Adi. High Fidelity Neural Audio Compression. *Transactions on Machine Learning Research*, 2023. Featured Certification, Reproducibility Certification.

[125] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc.

[126] Zalán Borsos, Raphaël Marinier, Damien Vincent, Eugene Kharitonov, Olivier Pietquin, Matt Sharifi, Dominik Roblek, Olivier Teboul, David Grangier, Marco Tagliasacchi, and Neil Zeghidour. AudioLM: A Language Modeling Approach to Audio Generation. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 31:2523–2533, June 2023.

[127] Alec Radford and Karthik Narasimhan. Improving Language Understanding by Generative Pre-Training. 2018.

[128] Yu-An Chung, Yu Zhang, Wei Han, Chung-Cheng Chiu, James Qin, Ruoming Pang, and Yonghui Wu. w2v-BERT: Combining Contrastive Learning and Masked Language Modeling for Self-Supervised Speech Pre-Training. In *2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pages 244–250, 2021.

[129] Sanyuan Chen, Chengyi Wang, Yu Wu, Ziqiang Zhang, Long Zhou, Shujie Liu, Zhuo Chen, Yanqing Liu, Huaming Wang, Jinyu Li, Lei He, Sheng Zhao, and Furu Wei. Neural Codec Language Models are Zero-Shot Text to Speech Synthesizers. *IEEE Transactions on Audio, Speech and Language Processing*, 33:705–718, 2025.

[130] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019.

[131] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Comput.*, 9(8):1735–1780, nov 1997.

[132] Francesco Nespoli, Daniel Barreda, Jöerg Bitzer, and Patrick A. Naylor. Two-Stage Voice Anonymization for Enhanced Privacy. In *Proc. INTERSPEECH 2023*, pages 3854–3858, 2023.

[133] Michele Panariello, Massimiliano Todisco, and Nicholas Evans. Preserving spoken content in voice anonymisation with character-level vocoder conditioning. In *4th Symposium on Security and Privacy in Speech Communication*, pages 12–16, 2024.

[134] Jiangyi Deng, Fei Teng, Yanjiao Chen, Xiaofu Chen, Zhaohui Wang, and Wenyuan Xu. V-Cloak: Intelligibility-, Naturalness- & Timbre-Preserving Real-Time Voice Anonymization. pages 5181–5198.

[135] Hubert Siuzdak. Vocos: Closing the gap between time-domain and Fourier-based neural vocoders for high-quality audio synthesis. In *The Twelfth International Conference on Learning Representations*, 2024.

[136] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. pages 11976–11986.

[137] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: visual reasoning with a general conditioning layer. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18, pages 3942–3951. AAAI Press.

[138] Won Jang, Dan Lim, Jaesam Yoon, Bongwan Kim, and Juntae Kim. UnivNet: A Neural Vocoder with Multi-Resolution Spectrogram Discriminators for High-Fidelity Waveform Generation. In *Proc. Interspeech 2021*, pages 2207–2211, 2021.

[139] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*, 2019.

[140] Tom Bäckström. Privacy in speech technology, 2022.

[141] Sarina Meyer, Xiaoxiao Miao, and Ngoc Thang Vu. VoicePAT: An Efficient Open-Source Evaluation Toolkit for Voice Privacy Research. *IEEE Open Journal of Signal Processing*, 5:257–265, 2024.

[142] Sang gil Lee, Wei Ping, Boris Ginsburg, Bryan Catanzaro, and Sungroh Yoon. BigVGAN: A Universal Neural Vocoder with Large-Scale Training. In *The Eleventh International Conference on Learning Representations*, 2023.

[143] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning: with Applications in R*. Springer Publishing Company, Incorporated, 2014.

[144] Yuekai Zhang, Ziyan Jiang, Jesús Villalba, and Najim Dehak. Black-box attacks on spoofing countermeasures using transferability of adversarial examples. In *Proc. Interspeech 2020*, pages 4238–4242, 2020.

[145] Michele Panariello, Wanying Ge, Hemlata Tak, Massimiliano Todisco, and Nicholas Evans. Malafide: a novel adversarial convolutive noise attack against deepfake and spoofing detection systems. In *Proc. INTERSPEECH 2023*, pages 2868–2872, 2023.

[146] Jingyang Li, Dengpan Ye, Long Tang, Chuanxi Chen, and Shengshan Hu. Voice Guard: Protecting Voice Privacy with Strong and Imperceptible Adversarial Perturbation in the Time Domain. In Edith Elkind, editor, *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pages 4812–4820. International Joint Conferences on Artificial Intelligence Organization, 8 2023. Main Track.

[147] Shihao Chen, Liping Chen, Jie Zhang, KongAik Lee, Zhenhua Ling, and Lirong Dai. Adversarial Speech for Voice Privacy Protection from Personalized Speech Generation. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11411–11415, 2024.

[148] Yi Xie, Cong Shi, Zhuohang Li, Jian Liu, Yingying Chen, and Bo Yuan. Real-time, universal, and robust adversarial attacks against speaker recognition systems. In *Proc. ICASSP 2020*, pages 1738–1742, 2020.

[149] Xingyu Zhang, Xiongwei Zhang, Wei Liu, Xia Zou, Meng Sun, and Jian Zhao. Waveform level adversarial example generation for joint attacks against both automatic speaker verification and spoofing countermeasures. *Engineering Applications of Artificial Intelligence*, 116:105469, 2022.

[150] Mintong Kang, Chejian Xu, and Bo Li. AdvWave: Stealthy Adversarial Jailbreak Attack against Large Audio-Language Models. In *The Thirteenth International Conference on Learning Representations*, 2025.

[151] Nicholas Carlini and David Wagner. Audio Adversarial Examples: Targeted Attacks on Speech-to-Text. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 1–7, 2018.

[152] Chiara Galdi, Michele Panariello, Massimiliano Todisco, and Nicholas Evans. 2D-Malafide: Adversarial Attacks Against Face Deepfake Detection Systems. In *2024 International Conference of the Biometrics Special Interest Group (BIOSIG)*, pages 1–7, 2024.

[153] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards Deep Learning Models Resistant to Adversarial Attacks. In *International Conference on Learning Representations*, 2018.

[154] A.W. Rix, J.G. Beerends, M.P. Hollier, and A.P. Hekstra. Perceptual evaluation of speech quality (PESQ)-a new method for speech quality assessment of telephone networks and codecs. In *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No.01CH37221)*, volume 2, pages 749–752 vol.2, 2001.

[155] Cees H. Taal, Richard C. Hendriks, Richard Heusdens, and Jesper Jensen. An Algorithm for Intelligibility Prediction of Time–Frequency Weighted Noisy Speech. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(7):2125–2136, 2011.

[156] Carlos Franzreb, Arnab Das, Hannes Gieseler, Eva Charlotte Jahn, Tim Polzehl, and Sebastian Möller. Towards Audiovisual Anonymization for Remote Psychotherapy: a Subjective Evaluation. In *4th Symposium on Security and Privacy in Speech Communication*, pages 102–110, 2024.

[157] Xiang Lyu, Yuxuan Wang, Tianyu Zhao, and Huadai Liu. Fast Adaptation of Pretrained Speaker Verification System for Source Speaker Tracking. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–2, 2025.

[158] Rayane Bakari, Olivier Le Blouch, Nicholas Evans, Nicolas Gengembre, Michele Panariello, and Massimiliano Todisco. The influence of non-timbral cues in voice anonymisation and evaluation. In *5th Symposium on Security and Privacy in Speech Communication*, pages 35–42, 2025.

[159] Carlos Franzreb, Arnab Das, Tim Polzehl, and Sebastian Möller. Private kNN-VC: Interpretable Anonymization of Converted Speech. In *Interspeech 2025*, pages 3224–3228, 2025.

[160] Jixun Yao, Hexin Liu, Eng Siong Chng, and Lei Xie. EASY: Emotion-aware Speaker Anonymization via Factorized Distillation. In *Interspeech 2025*, pages 3219–3223, 2025.

[161] Hubert Siuzdak, Florian Grötschla, and Luca A Lanzendörfer. SNAC: Multi-Scale Neural Audio Codec. In *Audio Imagination: NeurIPS 2024 Workshop AI-Driven Speech, Music, and Sound Generation*, 2024.

[162] Rithesh Kumar, Prem Seetharaman, Alejandro Luebs, Ishaan Kumar, and Kundan Kumar. High-fidelity audio compression with improved RVQGAN. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

[163] Xin Zhang, Dong Zhang, Shimin Li, Yaqian Zhou, and Xipeng Qiu. SpeechTokenizer: Unified Speech Tokenizer for Speech Language Models. In *The Twelfth International Conference on Learning Representations*, 2024.

[164] Alexandre Défossez, Laurent Mazaré, Manu Orsini, Amélie Royer, Patrick Pérez, Hervé Jégou, Edouard Grave, and Neil Zeghidour. Moshi: a speech-text foundation model for real-time dialogue. Technical report, 2024.

[165] Ryan Langman, Ante Jukić, Kunal Dhawan, Nithin Rao Koluguri, and Jason Li. Spectral Codecs: Improving Non-Autoregressive Speech Synthesis with Spectrogram-Based Audio Codecs, 2025.

[166] Siyi Zhou, Yiquan Zhou, Yi He, Xun Zhou, Jinchao Wang, Wei Deng, and Jingchen Shu. IndexTTS2: A Breakthrough in Emotionally Expressive and Duration-Controlled Auto-Regressive Zero-Shot Text-to-Speech. *arXiv preprint arXiv:2506.21619*, 2025.

[167] Xinsheng Wang, Mingqi Jiang, Ziyang Ma, Ziyu Zhang, Songxiang Liu, Linqin Li, Zheng Liang, Qixi Zheng, Rui Wang, Xiaoqin Feng, Weizhen Bian, Zhen Ye, Sitong Cheng, Ruibin Yuan, Zhixian Zhao, Xinfa Zhu, Jiahao Pan, Liumeng Xue, Pengcheng Zhu, Yunlin Chen, Zhifei Li, Xie Chen, Lei Xie, Yike Guo, and Wei Xue. Spark-TTS: An Efficient LLM-Based Text-to-Speech Model with Single-Stream Decoupled Speech Tokens, 2025.

[168] Resemble AI. Chatterbox-TTS. https://github.com/resemble-ai/chatterbox, 2025. GitHub repository.

[169] Yuancheng Wang, Jiachen Zheng, Junan Zhang, Xueyao Zhang, Huan Liao, and Zhizheng Wu. Metis: A Foundation Speech Generation Model with Masked Generative Pre-training. *arXiv preprint arXiv:2502.03128*, 2025.

[170] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7B, 2023.

[171] KimiTeam, Ding Ding, Zeqian Ju, Yichong Leng, Songxiang Liu, Tong Liu, Zeyu Shang, Kai Shen, Wei Song, Xu Tan, Heyi Tang, Zhengtao Wang, Chu Wei, Yifei Xin, Xinran Xu, Jianwei Yu, Yutao Zhang, Xinyu Zhou, Y. Charles, Jun Chen, Yanru Chen, Yulun Du, Weiran He, Zhenxing Hu, Guokun Lai, Qingcheng Li, Yangyang Liu, Weidong Sun, Jianzhou Wang, Yuzhi Wang, Yuefeng Wu, Yuxin Wu, Dongchao Yang, Hao Yang, Ying Yang, Zhilin Yang, Aoxiong Yin, Ruibin Yuan, Yutong Zhang, and Zaida Zhou. Kimi-Audio Technical Report, 2025.

[172] Tu Anh Nguyen, Benjamin Muller, Bokai Yu, Marta R. Costa-jussa, Maha Elbayad, Sravya Popuri, Christophe Ropers, Paul-Ambroise Duquenne, Robin Algayres, Ruslan Mavlyutov, Itai Gat, Mary Williamson, Gabriel Synnaeve, Juan Pino, Benoît Sagot, and Emmanuel Dupoux. SpiRit-LM: Interleaved spoken and written language model. *Transactions of the Association for Computational Linguistics*, 13:30–52, 2025.

[173] Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang-gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, and Bryan Catanzaro. Audio Flamingo 3: Advancing Audio Intelligence with Fully Open Large Audio Language Models. *arXiv preprint arXiv:2507.08128*, 2025.

[174] Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun MA, and Chao Zhang. SALMONN: Towards generic hearing abilities for large language models. In *The Twelfth International Conference on Learning Representations*, 2024.

[175] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. Emergent Abilities of Large Language Models. *Transactions on Machine Learning Research*, 2022. Survey Certification.

[176] Rao Ma, Adian Liusie, Mark Gales, and Kate Knill. Investigating the emergent audio classification ability of ASR foundation models. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4746–4760, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[177] Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. Robust speech recognition via large-scale weak supervision. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

[178] Thomas Thebaud, Nicholas Mehlman, Yaohan Guan, Laureano Moro-Velazquez, Jesus Villalba Lopez, Shrikanth Narayanan, and Najim Dehak. PPX-Anon: Prosody, Pitch and X-Vectors for De-Anonymization; our submission to the Voice Attacker Challenge 2024. In *5th Symposium on Security and Privacy in Speech Communication*, pages 61–67, 2025.

[179] 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*, pages 1708–1712, 2020.

[180] Nathalie Vauquier, Brij Mohan Lal Srivastava, Seyed Ahmad Hosseini, and Emmanuel Vincent. Legally validated evaluation framework for voice anonymization. In *Interspeech 2025*, pages 3229–3233, 2025.

[181] Tom Bäckström, Mohammad Hassan Vali, My Nguyen, and Silas Rech. Privacy Disclosure of Similarity Rank in Speech and Language Processing, 2025.

[182] Chia-Hua Wu, Wanying Ge, Xin Wang, Junichi Yamagishi, Yu Tsao, and Hsin-Min Wang. A Comparative Study on Proactive and Passive Detection of Deepfake Speech. In *Interspeech 2025*, pages 5328–5332, 2025.

[183] Lauri Juvela and Xin Wang. Collaborative Watermarking for Adversarial Speech Synthesis. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 11231–11235, 2024.

[184] Robin San Roman, Pierre Fernandez, Hady Elsahar, Alexandre Défossez, Teddy Furon, and Tuan Tran. Proactive detection of voice cloning with localized watermarking. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

[185] Yao Qin, Nicholas Carlini, Garrison Cottrell, Ian Goodfellow, and Colin Raffel. Imperceptible, Robust, and Targeted Adversarial Examples for Automatic Speech Recognition. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5231–5240. PMLR, 09–15 Jun 2019.

[186] Yibo Bai, Sizhou Chen, Michele Panariello, Xiao-Lei Zhang, Massimiliano Todisco, and Nicholas Evans. MDD: a Mask Diffusion Detector to Protect Speaker Verification Systems from Adversarial Perturbations. In *2025 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2025.

[187] Anna Leschanowsky, Ünal Ege Gaznepoglu, and Nils Peters. Voice Anonymization for All-Bias Evaluation of the Voice Privacy Challenge Baseline Systems. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4785–4789, 2024.

[188] Auguste Kerckhoffs. La cryptographie militaire. *Journal des sciences militaires*, IX, 1883.

Acknowledgments

This is most likely the last *Acknowledgments* section I will write in my life. I'm only adding it in the final version after the reviews, which means no one is double checking it: I am therefore free to regurgitate here whatever random crap my heart desires. I feel like Judas Priest in the music video of *Breaking the Law*. Let's begin.[1]

Very obvious choice, but family comes first. Thanks to my mom, my dad, my sister. A mamma, che in qualche modo è riuscita a tenere insieme fino ad ora una famiglia di tre Panarielli, impresa degna del premio Nobel per la pace; a papà, per gli slogan intellettuali che saltuariamente mi rifila, tipo "Pensa stocastico", salvo poi prendere e andare a bere al Gasoline;[2] a Valis, per tutta l'energia che ci porta, per tutto quello che è ora e quello che sarà in futuro.

The second place has to go to Luca M., effectively my oldest friend. Thanks for all the experiences we've shared, all the projects we have collaborated on and will keep collaborating on. Somehow, our activities keep getting intertwined, which I'm so very happy about. For the record, Luca was the one to show me the Bark model for the first time, which got me interested in NAC language modeling, which eventually culminated in [87]. At the time of writing, it is my highest-cited first author paper. Bro bumped up my h-index without even knowing what an h-index is.

Third runner-up: thanks to Luca D. for listening to power metal with me in high school. I still listen to power metal, he doesn't, and it definitely shows: he's a clever, strong and independent man, and I'm an idiot who makes funny voices in python for a living.[3] We are bound by having both pursued the way of research for the first few years of adult life. While I don't know where that will take us, I do know that I admire him greatly. There's still a lot I can learn from him.

[1] Just like any other cheesy *Acknowledgments* section, most of the people I'm thanking have absolutely nothing to do with this thesis. Then why am I thanking them? For the feels, for the bliss, and ultimately to make it look like this dissertation was some grandiose magnus opus at the apex of a journey rather than the result of sleep-deprived me frantically typing it out over both weekdays and weekends.

[2] Non glielo dirò mai, ma li apprezzo moltissimo lo stesso.

[3] Still completely worth all the power metal.

Completing the hometown roster: thanks to Filippo for all the good times we have whenever I go back home, and for being the only one (besides my mom) who understands all of my *Risoterapia* references.[4] To Paloma, Aurora and Maria for sticking around for long enough to see me get a PhD. To all the good guys of the *St. George* group, whom I don't hear from very often, but always greet me with a smile when we meet.

Let us move to Torino, la alta, elegante Torino. Thanks to the *Carbonara* group, in alphabetic order: Alessandro, Antonio, Martino. For taking care of each other as we were learning to step out of our homes into the mad labyrinth that is PoliTo. I bless the random circumstances that made us meet, and whatever wonderful butterfly effect ensued. And of course thanks to the *Data scienziati* who helped me survive through la bici di B., iCarl, la tesina, and whatever the fuck Data Ethics was. Oh, and you know... through COVID.
Honorable mention to Alessandrou Baldou and Lorenzo, who managed to populate the interstice between PoliTo and EURECOM, despite the unfortunate conditions. I should probably, like, text them at some point. Imagette.

Then comes Struer. Thanks to Sven [SSH] and Pablo [PMN] for being absolutely iconic supervisors of my adventures in B&O[5] — one of the best times of my life. The most heartfelt thanks to the man of 4 names, 5 languages, and numerous (mis)adventures André [ABFD], and the kind, most radiant and warm-hearted Niels [NEMK]: they even took the time to fly across Europe to spectate my defense. It was amazing to have you guys there, you rock! Further BOGO friends deserve a mention: Pia [PIPO], Dori [DHAU], Cassandra [CASR], Kate [KABO], Joe [JOWI], Nicolás [NIAL].[6] A little part of me named MIPA is still with you guys.

Finally, EURECOM: the melting pot that somehow became the stage for all the endeavors, good and bad, that resulted in this dissertation. Thanks first and foremost to Nick and Max, for believing in me since day one and bringing me to the most intriguing places during conferences, including: a stereotypically Guinness-centered pub in Dublin; a futuristic-looking whiskey bar in Seoul directly out of a John Wick movie; some whatever pretentiously fancy rooftop party in Kos (??). All in the name of speech privacy, of course. For science.
Thanks to Pepe for welcoming me to Antibes during the first few months, and for trying to pass down every drop of useful knowledge to face a PhD. Thanks to Wanying, Oubaïda and Hemlata for being amazing lab mates, for all the in-office banter, and for trying to make me feel at home as I was taking my first steps into this adventure. I'm trying to do the same for our new wave of colleagues, and I'm not sure I'm doing as much of a good job. On that note, I should thank them

[4] Gratis è ancora più bono!

[5] The *Supervisor iconicness* race is ultimately won by Nick and Max, but they have the unfair advantage of having supervised me for longer.

[6] Can't reminisce too much here because this is already getting too long — but they've been extensively thanked in my Master thesis, so it's fine. On that note, a negative acknowledgment (a dis-acknowledgment, if you will) to my PoliTo Master's thesis supervisor.

too: Mohamed first and foremost for putting up with all my bullshit for the longest time of all,[7] but also Yangyang, Yibo, Matteo, Emma, Shilpa, and Anna. You guys are still well into your own respective adventures, and I hope to be able to witness them for as long as possible — because I'm sure they will be great. Go team!
Though not officially part of the *Audio Security and Privacy* group, I like to believe Chiara is spiritually with us. I kinda consider her a semi-supervisor by now, so she deserves a thanks too.

Among the few people who have witnessed my PhD in its entirety is Nour — a special one indeed. She managed to warm my heart even in the most trying times. Sure, somehow we end up being in different parts of the world like 90% of the time despite technically living 15 minutes away from each other. But whenever we manage to meet, it feels comfortable and safe, and it's all that matters. Grazie, amica!

Then of course we have my amazing gymbros of JLP: Youri and Jean Flavien, with whom I've shared many adventures inside and outside EURECOM. I will always cherish the memory of our trips, our gym sessions and of course our lunches at Poke Nice and Pizza Noli, both of which will definitely go bankrupt without us. Youri, thanks for the many musical shenanigans, for being the other side of FeatherShell, for all the fun times we've spent recording stuff while pissing off your neighbor. JF, thanks for all the patience you've had in showing me the ways of the gym, even if tout était bien aussi. Jokes aside, you are actually a great coach because you always find a way to lift everyone's spirits[8] — and you do it with unmatched kindness, a rare feat. Anyway, hope to see you soon while driving your brand new Honda Civic!

Thanks to Federico for sharing so many dense conversations *(* sigh *)* and so many fun times behind the mic (more of which will hopefully come soon). Although you're back in Italy, I still firmly believe in what we sang multiple times: *no farewell could be the last one, if you long to meet again...* E grazie soprattuto per aver ascoltato e provato a capire, con tanta pazienza.

Vincenzo deserves a mention as well! Our chats were few and far between, but as meaningful as they could get. Thank you for taking the time to show me all the interesting sides of your personality, and thank you for allowing me to show you mine[9] (not saying that mine are interesting, but you get the point). You are a greatly talented individual, and I'm sure you'll go farther than I could ever hope to.

Thanks to the members of the anonymization police, who actually helped in making one of the papers contained in this thesis come together: Sarina, Pierre and Xiaoxiao. Hanging out with you guys at conferences was both educational AND fun. Who knows if we'll ever meet again — I certainly hope so — but I sure had a great time with you. You are great researchers, and I look up to every one of you.

[7] I'm grateful, my friend. Truly.

[8] Even though sometimes someone loses their temper... mais ça arrive.

[9] No, this is not an innuendo. Although it does sound a lot like one.

I could go on for pages and pages (as I have the tendency to do), but I'm afraid this would get too long, so let's start wrapping up. Thanks to Sophie for single-handedly holding together the whole PhD system at EURECOM; thanks to the donkeys for sharing the burdens of life with me, especially the one I talked to a while ago. Thanks to the guy from Moonshine for always saying hi, although I have to say his chicken wings are terribly overpriced. Thanks to all the guys from the Music Club for the unforgettable concerts. Thanks to Sappho for the conversations we've had on the walls of Antibes — I still hold onto your chain. Thanks to all the ravens that populate my tales and dreams (one in particular stands out). As I said long ago, thanks to power metal in general, just because.

Lastly, thanks to everyone I could not name, everyone whose name I don't even know, to all the voices that have contributed to the music up to this point; to all the characters that, one way or another, have been part of the stories I have crafted in my cosmos.

Mic drop.

Sophia Antipolis, 15 December 2025 ***Michele Panariello***