SEED: Speaker Embedding Enhancement Diffusion Model

Kihyun Nam[1], Jungwoo Heo[2], Jee-weon Jung[†3], Gangin Park[4], Chaeyoung Jung[1], Ha-Jin Yu[2], Joon Son Chung[1]

[1]*School of Electrical Engineering, Korea Advanced Institute of Science and Technology, South Korea*
[2]*School of Computer Science, University of Seoul, South Korea*
[3]*Language Technologies Institute, Carnegie Mellon University, United States*
[4]*Department of Electrical and Computer Engineering, Seoul National University, South Korea*

nkh.mmai@kaist.ac.kr, jungwoo4021@gmail.com, jeeweonj@ieee.org, ssonpull519@snu.ac.kr, codud9914@kaist.ac.kr, hjyu@uos.ac.kr, joonson@kaist.ac.kr

Abstract

A primary challenge when deploying speaker recognition systems in real-world applications is performance degradation caused by environmental mismatch. We propose a diffusion-based method that takes speaker embeddings extracted from a pre-trained speaker recognition model and generates refined embeddings. For training, our approach progressively adds Gaussian noise to both clean and noisy speaker embeddings extracted from clean and noisy speech, respectively, via forward process of a diffusion model, and then reconstructs them to clean embeddings in the reverse process. While inferencing, all embeddings are regenerated via diffusion process. Our method needs neither speaker label nor any modification to the existing speaker recognition pipeline. Experiments on evaluation sets simulating environment mismatch scenarios show that our method can improve recognition accuracy by up to 19.6% over baseline models while retaining performance on conventional scenarios. We publish our code here[1].

Index Terms: speaker recognition, diffusion probabilistic model, representation enhancement, real-world environment

1. Introduction

Speaker recognition systems are widely used in various fields such as user authentication, security, and voice interfaces. However, differences in recording equipment, background noise, and other environmental factors in real-world conditions can introduce substantial acoustic discrepancies between utterances from the same speaker, degrading the recognition performance [1, 2]. Consequently, achieving robustness against such domain or environment mismatches has become a critical challenge for real-world applications.

Such domain mismatch problems, including environment mismatch scenarios, ultimately widen the gap in the speaker embedding space and reduce the speaker identity similarity among different utterances from the same speaker. To mitigate this problem, a variety of methods have been proposed, including training with large-scale datasets [3, 4] containing real-world noise, simulating diverse noise conditions through data augmentation [5, 6], and leveraging Disentangled Representation Learning (DRL) [2, 7, 8] techniques to separate and suppress task-irrelevant information. Although these approaches have shown some effectiveness, they often require complex network architectures, numerous hyperparameters, and extensive training resources.

Recently, Diffusion Probabilistic Model (DPM) [9, 10] has gained significant attention as a powerful generative model, capable of producing high-fidelity data in various domains, such as image generation and speech enhancement. The core principle of DPM is to incrementally inject noise (Forward process) into the data until it approximates a Gaussian distribution, and then gradually remove this noise (Reverse process) to recover the original data distribution.

In the audio domain, DPM-based techniques have primarily been explored for front-end signal enhancement, where noisy waveforms or spectrograms are transformed into cleaner signals before being used by downstream tasks. Although these studies effectively extend existing noise suppression and filtering methods, they generally operate on raw audio signals, offering no guarantee of directly resolving domain mismatch in the speaker embedding space. Thus, applying diffusion models at the embedding level, where speaker identity is explicitly encoded, remains largely unexplored, leaving open the question of whether front-end audio enhancement alone can ensure higher-quality speaker embeddings.

In this study, we propose a novel DPM-based approach that minimises the discrepancy between speaker embeddings extracted from clean and noisy speech samples. Our method is built on the assumption that, since both the clean embedding and the noisy embedding (obtained from noisy audio) inherently encapsulate the same speaker identity, their difference in the embedding space is not excessively large. By applying the forward DPM process, we progressively add Gaussian noise to both clean speaker embedding and noisy speaker embedding, causing their distributions to converge toward an almost identical Gaussian representation. Subsequently, the reverse process incrementally removes the noise, reconstructing the embeddings into a clean form. In this process, even embeddings extracted under adverse conditions are refined into more clean representations, ensuring that utterances from the same speaker remain closely aligned in the embedding space, regardless of external recording factors. We refer to this DPM-based approach as the Speaker Embedding Enhancement Diffusion (SEED) model. To the best of our knowledge, SEED is the first attempt in speaker recognition to apply a diffusion model directly at the embedding level.

Our proposed SEED has several benefits. It requires no additional training or structural modifications to an already-trained speaker recognition model, making it versatile. It can be trained and applied on top of arbitrary speaker embeddings. Secondly, SEED does not rely on speaker labels; a corpus of clean speech where acoustic augmentation can be applied is enough — there are millions of hours of such data, e.g., corpus designed for ASR or TTS. Lastly, SEED's lightweight architecture, based on simple residual fully-connected (fc) blocks, demonstrates robust performance under environment mismatch condition, surpassing existing methods. Empirical evaluations across diverse speaker recognition systems and datasets show that SEED improves speaker identification accuracy by up to 19.6% over baseline methods in mismatched conditions, underscoring the potential of embedding-level diffusion as a fundamental solution for real-world speaker recognition system deployments.

2. Related works

2.1. Diffusion Probabilistic Model

Diffusion probabilistic models (DPMs) [9, 10] have emerged as a powerful generative framework for high-fidelity data generation
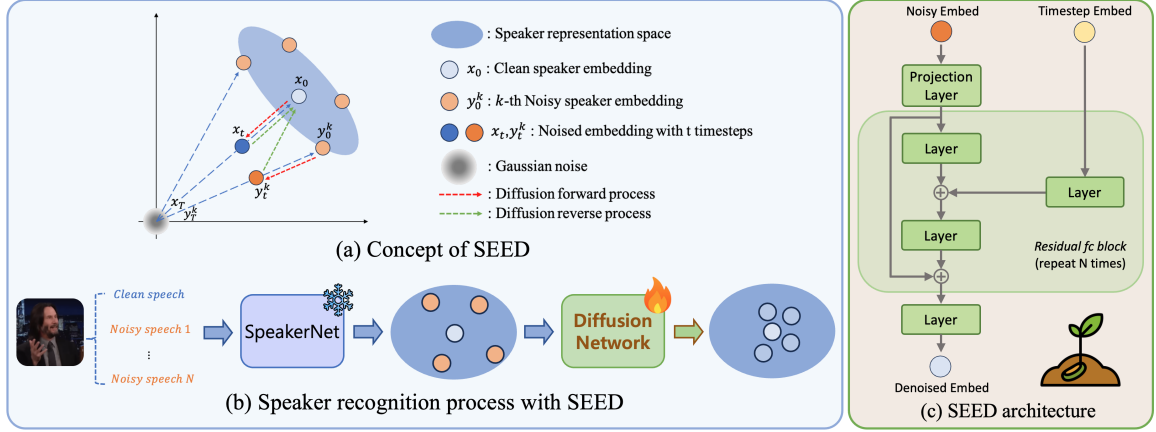

Figure 1: *Illustration of Speaker Embedding Enhancement Diffusion (SEED) model. (a) explains the concept of our diffusion mechanism. (b) shows the whole training process of SEED. (c) illustrates the architecture of SEED.*

across various domains [11, 12, 13]. These models operate through a forward process that gradually corrupts data with Gaussian noise, followed by a reverse process that reconstructs the original data distribution. However, diffusion models require numerous sampling steps, leading to slow inference. To address this, more efficient techniques, such as the Denoising Diffusion Implicit Model (DDIM) [10], have been introduced, utilising a non-Markovian process to accelerate sampling. In this work, we employ DDIM sampling to reduce the number of reverse steps while maintaining the quality of generated embeddings.

2.2. Feature enhancement

Speech enhancement. Speech enhancement primarily aims to reduce noise and reverberation. Traditional approaches rely on spectral subtraction [14], Wiener filtering [15], or statistical methods [16]. More recently, deep learning has enabled data-driven techniques, including DNN-based autoencoders [17], CNNs [18], GANs [19], VAEs [20], and diffusion-based restoration [21]. However, integrating these enhanced signals into downstream tasks often proves challenging: while intelligibility may improve, distortions can arise that compromise speaker-specific information [22, 23]. Joint optimisation efforts [24, 25] highlight the difficulty of fully preserving speaker-relevant features in enhancement-based speaker recognition systems.

Speaker embedding enhancement. Enhancing speaker embeddings to address domain mismatch has been extensively studied. Early methods based on i-vector and x-vector frameworks employed PLDA and length normalisation [26], but these often degrade under severe noise or unseen conditions. DRL aims to isolate speaker identity from nuisance factors [8, 2], yet typically demands additional labeling and design of specialised training objectives. More recently, refining extracted speaker embeddings has shown promise for mitigating domain shifts; for instance, [27] proposed post-processing to reduce variability introduced by recording conditions. Our work proposes a diffusion-based approach that enhances speaker embeddings directly, requiring no speaker labels and allowing the use of clean speech datasets of various audio tasks.

3. Proposed method

This section describes the configuration of our proposed SEED, including the batch setup with audio augmentation and the speaker embedding extraction process. The concept, process, and model structure of our methodology are shown in Figure 1.

3.1. Batch configuration with multi-pair audio augmentation

To simulate environment mismatch scenarios, we configure each mini-batch to contain pairs of *clean* and *noisy* audio data. Since the task involves transforming speaker embeddings extracted from noisy audio to match those from clean audio, it is essential to ensure that both embeddings share the same speaker identity and content. Noisy audio samples are generated through audio augmentation applied to one clean audio data a_{clean}. Specifically, perturbations such as environmental noise and reverberation are introduced to simulate diverse real-world conditions. This process produces N noisy variants, $a_{\text{noisy}}^0, a_{\text{noisy}}^1, ..., a_{\text{noisy}}^{N-1}$, where each variant retains the same content as the clean audio but exhibits different acoustic characteristics. These variants allow the diffusion model to generalise by learning robust reconstruction paths in the reverse process. For the i-th mini-batch, the paired audio data can be represented as:

$$(a_{\text{clean}}, \{a_{\text{noisy}}^0, a_{\text{noisy}}^1, ..., a_{\text{noisy}}^{N-1}\}).$$

Details of audio augmentation techniques are further described in Section 4.

3.2. Speaker embedding extractor

Our method utilises a *pre-trained speaker embedding network* that is not jointly optimised with SEED. This network extracts a D-dimensional speaker embedding vector from input audio a, capturing essential speaker identity features. The speaker embedding extraction process can be expressed as:

$$\mathbf{x}_0 = SpeakerNet(a_{clean}), \quad \mathbf{x}_0 \in \mathbb{R}^D,$$

where $\mathbf{x}_0$ is the corresponding clean speaker embedding and $SpeakerNet$ denotes speaker embedding network. Similarly, embeddings extracted from input noisy audio a_{noisy}^k are denoted by $\mathbf{y}_0^k$, where k indicates the k-th noisy variant. The embedding network typically consists of three main components: a feature extraction module (e.g., STFT or convolutional layers), an encoder network, and a pooling mechanism that aggregates temporal features into a fixed-size vector. These embeddings are then passed to the diffusion model for further enhancement. More details of the speaker embedding network architecture will be presented in Section 4.

3.3. Speaker Embedding Enhancement Diffusion

Our SEED builds upon the Denoising Diffusion Probabilistic Model (DDPM) [9], extending it to handle both clean and noisy speaker embeddings in a unified framework. This section first revisits the basic DDPM formulation based on ϵ-prediction (estimates Gaussian noise), then illustrates how we adapt it to directly estimate a clean speaker embedding $\mathbf{x}_0$ from the noisy speaker embedding $\mathbf{y}_0$ via *sample-prediction* (estimates target sample directly) formulation.

3.3.1. Basic DDPM Formulation

Let $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ be the original data (in our case, a speaker embedding). A forward diffusion process $q(\mathbf{x}_t | \mathbf{x}_0)$ progressively

corrupts $\mathbf{x}_0$ with Gaussian noise over $t \in \{1,...,T\}$ steps, defined as:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\overline{\alpha}_t}\mathbf{x}_0, (1-\overline{\alpha}_t)\mathbf{I}), \quad (1)$$

where $\alpha_t \in (0,1)$ is a noise scheduling parameter, $\overline{\alpha}_t = \prod_{\tau=1}^{t} \alpha_\tau$, and $\mathbf{I}$ is the identity matrix. To estimate $q(\mathbf{x}_{t-1}|\mathbf{x}_t)$, the parameterised reverse process $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$ aims to *denoise* $\mathbf{x}_t$ back to $\mathbf{x}_{t-1}$:

$$p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t,t), \sigma_t^2 \mathbf{I}), \quad (2)$$

where a neural network parameterised by θ predicts the mean $\mu_\theta(\mathbf{x}_t,t)$, and $\sigma_t^2 = 1-\alpha_t$ follows the same noise schedule. With standard DDPM, it is typically trained θ by minimising a variational lower bound on the data likelihood. As mentioned Section 2.1, we adopt an implicit sampler [10] (DDIM).

3.3.2. Diffusion Process of SEED

In forward process (the red arrow dash lines in Figure 1-a), we interpret $\mathbf{x}_0$ as a *clean* speaker embedding and $\mathbf{y}_0$ as its *noisy* counterpart. Applying Eq. (1), we can sample a corrupted state $\mathbf{x}_t$ via

$$\mathbf{x}_t = \sqrt{\overline{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0,\mathbf{I}), \quad (3)$$

and similarly, a corresponding corrupted *noisy* embedding $\mathbf{y}_t$ can be written as

$$\mathbf{y}_t = \sqrt{\overline{\alpha}_t}\mathbf{y}_0 + \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon}. \quad (4)$$

Thus, $\mathbf{x}_t$ and $\mathbf{y}_t$ are Gaussian corrupted embeddings, each from $\mathbf{x}_0$ and $\mathbf{y}_0$, reparameterised with $\boldsymbol{\epsilon}$.

In reverse process (the green arrow dash lines in Figure 1-a), based on Eq. (2), the reverse step $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)$ is instantiated by a network ϵ_θ that does ϵ-*prediction* which estimates $\boldsymbol{\epsilon}$, yielding the reconstruction $(\mathbf{x}_t \rightarrow \mathbf{x}_0)$ as:

$$\hat{\mathbf{x}}_0(\mathbf{x}_t,t) = \frac{1}{\sqrt{\overline{\alpha}_t}}\left(\mathbf{x}_t - \sqrt{1-\overline{\alpha}_t}\epsilon_\theta(\mathbf{x}_t,t)\right). \quad (5)$$

This reconstructs the clean embedding $\mathbf{x}_0$ from $\mathbf{x}_t$. However, standard DDPM addresses only a single distribution (either $q(\mathbf{x}_0)$ or $q(\mathbf{y}_0)$), whereas SEED also seeks reconstruction of $\mathbf{y}_0$ from $\mathbf{y}_t$ to be closer to $\mathbf{x}_0$. To achieve this, we consider Eq. (5) for $\mathbf{y}_0$ as:

$$\hat{\mathbf{y}}_0(\mathbf{y}_t,t) = \frac{1}{\sqrt{\overline{\alpha}_t}}\left(\mathbf{y}_t - \sqrt{1-\overline{\alpha}_t}\epsilon_\theta(\mathbf{y}_t,t)\right) \approx \mathbf{x}_0. \quad (6)$$

Unlike the standard reverse process $(\mathbf{x}_t \rightarrow \mathbf{x}_0)$ or $(\mathbf{y}_t \rightarrow \mathbf{y}_0)$, this cross-reconstruction $(\mathbf{y}_t \rightarrow \mathbf{x}_0)$ acts as *noise removal* in the speaker embedding space.

3.3.3. Training Objective

SEED jointly considers two objectives: (i) learning distribution of clean speaker embedding $\mathbf{x}_0$ for *high-quality* sampling, and (ii) learning the reverse process of $(\mathbf{y}_t \rightarrow \mathbf{x}_0)$ for embedding enhancement of Eq. (6) via direct optimisation of

$$||\mathbf{x}_0 - \hat{\mathbf{y}}_0(\mathbf{y}_t,t)||, \quad (7)$$

which introduces an additional scaled noise term as

$$||\mathbf{x}_0 - \hat{\mathbf{y}}_0(\mathbf{y}_t,t)|| = \Big\|\frac{1}{\sqrt{\overline{\alpha}_t}}\left(\mathbf{x}_t - \sqrt{1-\overline{\alpha}_t}\boldsymbol{\epsilon}\right) \\ - \frac{1}{\sqrt{\overline{\alpha}_t}}\left(\mathbf{y}_t - \sqrt{1-\overline{\alpha}_t}\epsilon_\theta(\mathbf{y}_t,t)\right)\Big\| \quad (8)$$

in terms of ϵ-*prediction*. Rearranging the terms within the norm reveals

$$\mathbf{x}_t - \mathbf{y}_t = \sqrt{\overline{\alpha}_t}(\mathbf{x}_0 - \mathbf{y}_0),$$

which acts as an additional *scaled noise* factor. If $||\mathbf{x}_0 - \mathbf{y}_0||$ is large, this scaled noise term may destabilise the training of the reverse process. However, under our assumption that $\mathbf{x}_0$ and $\mathbf{y}_0$ come from the same utterance of the same speaker, we expect the scaled noise to be moderate and the model learns to regress the scaled noise for each timestep accordingly. Furthermore, the *multi-pair audio augmentation* in Section 3.1 ensures the model encounters diverse noisy variants $\{\mathbf{y}_0^k\}$ around each $\mathbf{x}_0$, improving generalisation to various scaled noise patterns.

Finally, we propose a loss function that can implicitly optimise the scaled noise and epsilon terms via *sample-prediction*:

$$L_{\text{SEED}} = \mathbb{E}\left[||\mathbf{x}_0 - f_\theta(\mathbf{x}_t,t)|| + \sum_{k=0}^{N-1}||\mathbf{x}_0 - f_\theta(\mathbf{y}_t^k,t)||\right], \quad (9)$$

Here, f_θ denotes the trainable SEED network. Specifically, $f_\theta(\mathbf{x}_t,t)$ and $f_\theta(\mathbf{y}_t^k,t)$ directly predict $\hat{\mathbf{x}}_0$ and $\hat{\mathbf{y}}_0^k$, respectively. As a result, SEED can (i) ensure high-quality speaker embedding generation for clean samples and (ii) remove noise from $\mathbf{y}_t^k$ effectively, all within a unified diffusion framework.

4. Experiments

4.1. Model Configuration

4.1.1. Speaker embedding network

We employ three pre-trained speaker embedding networks. For the spectrogram-based speaker network, we selected '**H / ASP**' version of **ResNet34** [28] and **ECAPA-TDNN** [29]. For the raw-waveform-based speaker network, we chose ECAPA-TDNN network combined with '**WavLM Base+**' version of WavLM [30], named **WavLM-ECAPA**. The output speaker embedding dimensions of the ResNet34, ECAPA-TDNN, and WavLM-ECAPA models are 512, 256, and 192, respectively.

4.1.2. Speaker Embedding Enhancement Diffusion (SEED)

The SEED model adopts a representation diffusion model (RDM) [32] network composed of residual fc blocks. Each residual fc block consists of an input layer, a timestep embedding projection layer, and an output layer, where each layer includes a LayerNorm [33], a SiLU [34], and a linear layer. RDM [32] network is illustrated in Figure 1-c. For all baseline speaker recognition models, we use three residual fc blocks without the optional context layer, following the same configuration as [32]. Additionally, the hidden unit size of each layer within the fc blocks is set to twice the dimension of the speaker embedding used in each baseline model.

4.2. Implementation details

Datasets. For training, we use approximately 1,000 hours of clean speech data, comprising 460 hours from LibriTTS-R [35] and 577 hours from Libri-Light [36]. Notably, **no speaker labels are required**, as our training only depends on clean speech data. For evaluation, we employ five evaluation protocols: the three VoxCeleb1 variants (Vox1-O, Vox1-E, and Vox1-H) to evaluate generalisation performance, and VoxSRC23 [37] along with VC-Mix [38] to evaluate environment robustness, as these latter datasets reflect environment mismatch scenarios.

Multi-pair audio augmentation. For multi-pair audio augmentation, we generate three types of noisy speech from each clean utterance by applying reverberations from a simulated RIR dataset [6] and music and background noises from MUSAN [5]. The SNR is randomly selected within 0–15 dB for background noise and 5–15 dB for music noise. Consequently, each utterance forms a set of four paired samples (one clean and three noisy variants).

Training & Inference. All of our experiments utilise the PyTorch framework [39] together with the open-source `vox-celeb_trainer`[2]. We select AdamW Optimizer [40] with an initial learning rate of 0.0005. Our implementation is conducted on a single NVIDIA RTX 4090 GPU with 24 GB memory. Training takes around 60 epochs. For the diffusion process, we use a scaled linear noise schedule with 1,000 training timesteps T in the DDIM sampler [10] from `diffusers`[3].

In the inference step, we assume that all input speaker embeddings will be $\mathbf{y}_t$ and regenerate them via the diffusion process of SEED. Unlike typical DDIM sampling, our *sample-prediction*

---

[2]https://github.com/clovaai/voxceleb_trainer
[3]https://github.com/huggingface/diffusers

Table 1: *EER and minDCF Results on five evaluation sets.* **boldface score** *indicates that the model achieved the highest performance among the comparison groups on the same baseline and evaluation set.* → *means passing the output from the left model to the right model.* + *means scratch training the baseline network with additional strategies or networks on the right.* ↑ *indicates performance improvement over baseline.*

| | Environmental mismatch set | | | | Generalization set | | | | | |
| | VoxSRC23 | | VC-Mix | | Vox1-O | | Vox1-E | | Vox1-H | |
Model	EER	minDCF	EER	minDCF	EER	minDCF	EER	minDCF	EER	minDCF
Baseline										
ResNet34 [28]	5.50	0.308	3.07	0.245	0.88	0.079	1.07	0.076	2.21	0.147
ECAPA-TDNN [29]	5.93	0.335	2.96	0.261	0.90	0.064	1.16	0.081	2.38	0.152
WavLM-ECAPA [30]	5.07	0.283	2.32	0.195	0.83	0.063	0.98	0.058	1.92	0.115
Disentangled Representation Learning										
ResNet34 + DRL [2]	5.35↑	0.306↑	2.58↑	0.223↑	**0.86**↑	**0.068**↑	1.10	0.078	**2.20**↑	**0.142**↑
ECAPA-TDNN + DRL [2]	5.81↑	0.325↑	2.43↑	**0.212**↑	**0.82**↑	0.067	1.16	**0.080**↑	2.38	0.156
Embedding Enhancement Diffusion										
ResNet34 [28] → SEED	**5.31**↑	**0.303**↑	**2.56**↑	**0.219**↑	0.87↑	0.079	1.09	0.076	2.23	0.149
ECAPA-TDNN [29] → SEED	**5.53**↑	**0.309**↑	**2.38**↑	0.227↑	0.86↑	0.069	1.20	0.083	**2.30**↑	**0.148**↑
WavLM-ECAPA [30]→ SEED	**4.94**↑	**0.280**↑	**2.29**↑	0.200	**0.81**↑	0.065	**0.97**↑	0.058	1.92	0.115

Table 2: *Performance comparison between conventional audio enhancement-based approach and our embedding enhancement diffusion-based approach.* → *means passing the output from the left model to the right model.* **Params (M)** *denotes the number of model parameters in millions,* **RTF** *(Real-Time Factor) indicates the processing speed relative to real-time for end-to-end process, and* **Memory (GiB)** *represents the memory consumption during inference.* **RTF** *and* **Memory** *measurements are based on 4-second audio size. We report the Equal Error Rate (EER) for* **VC-Mix** *and* **Vox1-O** *evaluation set.*

Model	Params (M)	RTF	Memory (GiB)	VC-Mix	Vox1-O
SE [31] → ResNet34 [28]	44.6	0.0041	0.88	5.14	1.21
SE [31] → ECAPA-TDNN [29]	46.6	0.0038	0.88	4.63	1.17
SE [31] → WavLM-ECAPA [30]	137.7	0.0069	2.89	31.29	30.67
SE [31] → WavLM-ECAPA [30] (fine-tuned)	137.7	0.0069	2.89	2.24	0.81
ResNet34 [28] → SEED	20.8	0.0022	0.59	2.56	0.87
ECAPA-TDNN [29] → SEED	18.8	0.0020	0.59	2.38	0.86
WavLM-ECAPA [30] → SEED	105.6	0.0030	2.66	2.29	0.81

approach directly generates the cleansed output embedding from $\mathbf{y}_t$ in a single-step. We set the timestep t to 50 for single-step sampling. Empirically, we observed negligible performance differences compared to multi-step sampling. For WavLM-ECAPA, we apply a feature ensemble technique where the original speaker embedding and SEED's output embedding are summed to obtain the final speaker embedding.

Evaluation. We report two metrics: (1) Equal Error Rate (EER), where false accept and false reject rates converge, and (2) minimum Detection Cost Function (minDCF) from NIST SRE [41], using $C_{miss} = 1, C_{fa} = 1$, and $P_{target} = 0.05$. For ResNet34 and ECAPA-TDNN, each utterance is split into ten 4-second segments, and all segment pairs are evaluated to yield an averaged similarity score. By contrast, WavLM-ECAPA uses the entire utterance. We follow the scoring protocols described in [28, 30].

5. Results

In this section, we summarise and analyse our experimental results. To assess both environment-robustness and generalisation performance across diverse speaker recognition systems, we compare three baselines, an adversarial DRL-based method [2] designed to remove environmental factors from the speaker representation, and our proposed method (SEED). These results are presented in Table 1. Moreover, we evaluate the practical benefits of our approach compared to an existing audio enhancement-based approach [31], demonstrating the efficacy of SEED under real-world constraints in Table 2.

Environment-robustness performance. As shown in Table 1, integrating our SEED framework with each baseline model yields up to a 19.6% improvement on the environmental mismatch evaluation sets. Although a DRL approach [2] similarly achieves notable gains, it requires additional speaker and environment labels and more complex training procedures. In contrast, SEED requires no speaker labels and can be trained on a smaller training dataset, yet delivers competitive or superior outcomes. For example, on the VoxSRC23 set, applying SEED to ECAPA-TDNN outperforms the method in [2] by nearly 4.8%. Furthermore, in the VC-Mix set, ECAPA-TDNN with SEED almost matches the performance of

the WavLM baseline, which is pre-trained on a large datasets and diverse learning techniques. These results demonstrate that SEED effectively eliminates residual environmental information hidden in the speaker representation space, thereby improving the latent capacity of the speaker recognition system.

Practical and high-fidelity speaker representation generation. As shown in Table 1, our proposed method not only maintains remarkable generalisation performances on generalisation sets—achieving performance that is slightly higher or nearly indistinguishable from the baselines—but also delivers superior improvements on environment mismatch evaluation sets. This indicates that our approach consistently produces high-quality speaker embeddings. In other words, the diffusion-based paradigm demonstrates substantial potential for high-fidelity speaker representation generation at the embedding level.

Table 2 compares a conventional waveform-domain audio enhancement method with SEED. Despite requiring significantly fewer parameters, lower memory usage, and a reduced real-time factor, SEED outperforms most baselines on both VC-Mix and Vox1-O benchmarks. Notably, WavLM-ECAPA, which directly processes raw waveforms, suffers significant performance degradation unless it undergoes additional fine-tuning, whereas SEED provides both computational efficiency and robustness without any extra training. Consequently, it offers a compelling alternative to resource-intensive enhancement-based speaker recognition systems.

6. Conclusion

SEED leverages diffusion models for robust and generalised speaker embedding enhancement without requiring speaker labels or complex training. It seamlessly integrates with pre-trained speaker recognition models, enabling immediate deployment in speaker recognition systems for real-world applications. However, since SEED learns the gap between clean and noisy embeddings via the DDPM mechanism, some training instability may occur. To address this, future work will explicitly model the domain mismatch gap, ensuring stable generation even in extreme conditions and broadening SEED's applicability to diverse speech tasks.

7. Acknowledgements

8. References

[1] J. P. Campbell, "Speaker recognition: A tutorial," *Proceedings of the IEEE*, vol. 85, no. 9, pp. 1437–1462, 1997.

[2] K. Nam, H.-S. Heo, J. weon Jung, and J. Chung, "Disentangled representation learning for environment-agnostic speaker recognition," in *Proc. Interspeech*, 2024, pp. 2130–2134.

[3] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, "Voxceleb: Large-scale speaker verification in the wild," *Computer Speech & Language*, vol. 60, p. 101027, 2020.

[4] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep Speaker Recognition," in *Proc. Interspeech*, 2018, pp. 1086–1090.

[5] D. Snyder, G. Chen, and D. Povey, "Musan: A music, speech, and noise corpus," *arXiv preprint arXiv:1510.08484*, 2015.

[6] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. ICASSP*. IEEE, 2017, pp. 5220–5224.

[7] X. Wang, H. Chen, S. Tang, Z. Wu, and W. Zhu, "Disentangled representation learning," *arXiv preprint arXiv:2211.11695*, 2022.

[8] K. Nam, Y. Kim, J. Huh, H.-S. Heo, J. weon Jung, and J. S. Chung, "Disentangled representation learning for multilingual speaker recognition," in *Proc. Interspeech*, 2023, pp. 5316–5320.

[9] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Advances in neural information processing systems*, vol. 33, pp. 6840–6851, 2020.

[10] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=St1giarCHLP

[11] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[12] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. CVPR*, 2022, pp. 10 684–10 695.

[13] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "Audioldm: text-to-audio generation with latent diffusion models," in *Proc. ICML*, ser. ICML'23. JMLR.org, 2023.

[14] S. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 27, no. 2, pp. 113–120, 1979.

[15] J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proceedings of the IEEE*, vol. 67, no. 12, pp. 1586–1604, 1979.

[16] Y. Ephraim and D. Malah, "Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 32, no. 6, pp. 1109–1121, 1984.

[17] X. Lu, Y. Tsao, S. Matsuda, and C. Hori, "Speech enhancement based on deep denoising autoencoder." in *Proc. Interspeech*, vol. 2013, 2013, pp. 436–440.

[18] A. Pandey and D. Wang, "A new framework for cnn-based speech enhancement in the time domain," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 7, pp. 1179–1188, 2019.

[19] S. Pascual, A. Bonafonte, and J. Serrà, "Segan: Speech enhancement generative adversarial network," in *Proc. Interspeech*, 2017, pp. 3642–3646.

[20] S. Leglaive, L. Girin, and R. Horaud, "A variance modeling framework based on variational autoencoders for speech enhancement," in *2018 IEEE 28th international workshop on machine learning for signal processing (MLSP)*. IEEE, 2018, pp. 1–6.

[21] S. Welker, J. Richter, and T. Gerkmann, "Speech enhancement with score-based generative models in the complex stft domain," in *Proc. Interspeech*, 2022, pp. 2928–2932.

[22] K. Iwamoto, T. Ochiai, M. Delcroix, R. Ikeshita, H. Sato, S. Araki, and S. Katagiri, "How bad are artifacts?: Analyzing the impact of speech enhancement errors on asr," in *Proc. Interspeech*, 2022, pp. 5418–5422.

[23] O. Plchot, L. Burget, H. Aronowitz, and P. Matejka, "Audio enhancing with dnn autoencoder for speaker recognition," in *Proc. ICASSP*. IEEE, 2016, pp. 5090–5094.

[24] S. E. Eskimez, P. Soufleris, Z. Duan, and W. Heinzelman, "Front-end speech enhancement for commercial speaker verification systems," *Speech Communication*, vol. 99, pp. 101–113, 2018.

[25] Y. Wu, L. Wang, K. A. Lee, M. Liu, and J. Dang, "Joint feature enhancement and speaker recognition with multi-objective task-oriented network." in *Proc. Interspeech*, 2021, pp. 1089–1093.

[26] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems." in *Proc. Interspeech*, vol. 2011, 2011, pp. 249–252.

[27] J.-W. Jung, J.-H. Kim, H.-J. Shim, S. bin Kim, and H.-J. Yu, "Selective deep speaker embedding enhancement for speaker verification," in *The Speaker and Language Recognition Workshop (Odyssey 2020)*, 2020, pp. 171–178.

[28] Y. Kwon, H.-S. Heo, B.-J. Lee, and J. S. Chung, "The ins and outs of speaker recognition: lessons from VoxSRC 2020," in *Proc. ICASSP*. IEEE, 2021, pp. 5809–5813.

[29] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in tdnn based speaker verification," in *Proc. Interspeech*, 2020.

[30] S. Chen, C. Wang, Z. Chen, Y. Wu, S. Liu, Z. Chen, J. Li, N. Kanda, T. Yoshioka, X. Xiao *et al.*, "Wavlm: Large-scale self-supervised pre-training for full stack speech processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 6, pp. 1505–1518, 2022.

[31] A. Défossez, G. Synnaeve, and Y. Adi, "Real time speech enhancement in the waveform domain," in *Proc. Interspeech*, 2020, pp. 3291–3295.

[32] T. Li, D. Katabi, and K. He, "Return of unconditional generation: A self-supervised representation generation method," in *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[33] J. Lei Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *ArXiv e-prints*, pp. arXiv–1607, 2016.

[34] S. Elfwing, E. Uchibe, and K. Doya, "Sigmoid-weighted linear units for neural network function approximation in reinforcement learning," *Neural networks*, vol. 107, pp. 3–11, 2018.

[35] Y. Koizumi, H. Zen, S. Karita, Y. Ding, K. Yatabe, N. Morioka, M. Bacchiani, Y. Zhang, W. Han, and A. Bapna, "Libritts-r: A restored multi-speaker text-to-speech corpus," in *Proc. Interspeech*, 2023, pp. 5496–5500.

[36] J. Kahn, M. Riviere, W. Zheng, E. Kharitonov, Q. Xu, P.-E. Mazaré, J. Karadayi, V. Liptchinsky, R. Collobert, C. Fuegen *et al.*, "Libri-light: A benchmark for asr with limited or no supervision," in *Proc. ICASSP*. IEEE, 2020, pp. 7669–7673.

[37] J. Huh, J. S. Chung, A. Nagrani, A. Brown, J.-w. Jung, D. Garcia-Romero, and A. Zisserman, "The vox celeb speaker recognition challenge: A retrospective," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, 2024.

[38] H.-S. Heo, K. Nam, B.-J. Lee, Y. Kwon, M. Lee, Y. J. Kim, and J. S. Chung, "Rethinking session variability: Leveraging session embeddings for session robustness in speaker verification," in *Proc. ICASSP*, 2023.

[39] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," in *Proc. NeurIPS*, vol. 32, 2019.

[40] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. ICLR*, 2019.

[41] O. Sadjadi, C. Greenberg, E. Singer, D. Reynolds, L. Mason, and J. Hernandez-Cordero, "The 2018 nist speaker recognition evaluation," in *Proc. Interspeech*, 2019.