UnDiff

Unsupervised Voice Restoration with Unconditional Diffusion Model

Anastasiia Iashchenko[123*], Pavel Andreev[1*], Ivan Shchekotov[123*], Nicholas Babaev[1], Dmitry Vetrov[24]

[1] Samsung AI Center, Moscow; [2] HSE University, Moscow; [3] Skolkovo Institute of Science and Technology, Moscow; [4] Artificial Intelligence Research Institute, Moscow; [*] equal contribution

Interspeech 2023

Unsupervised Voice Restoration

Audio restoration tasks are inverse problems with the aim to **Restore the Signal** $\mathbf{x}$ from observations $\mathbf{y} = A(\mathbf{x})$ that have suffered **a Known Type of Degradation** A.

- e.g., bandwidth extension, declipping, and Mel-spectrogram inversion.

These problems are ill-posed in the sense that several different restorations may be equally plausible. Algorithms are typically constructed for

- A particular type of degradation using domain knowledge, and
- Data-driven generative models have also been recently proposed.

Unsupervised Voice Restoration

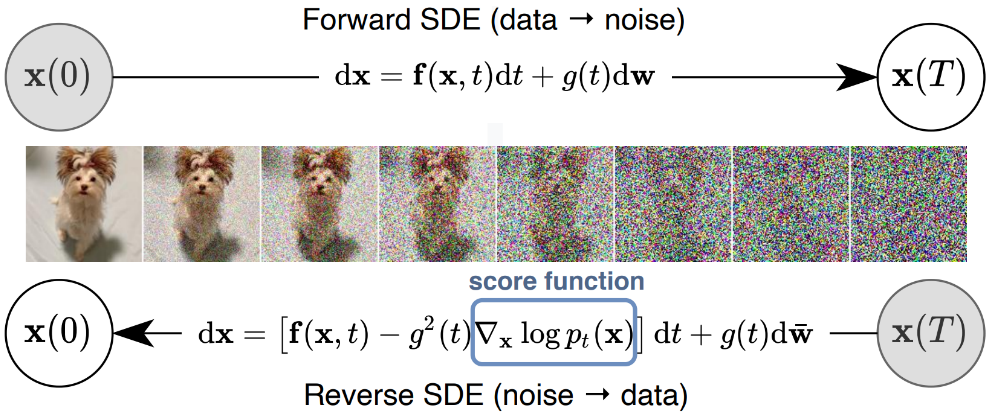
However, A common shortcoming is that an algorithm engineered for one type of degradation is typically **Not Useful for Others**.

To address this issue, this study introduces UnDiff, which uses a differentiable operator A to guide the **Diffusion Probabilistic Model** to solve various inverse tasks of speech processing.

- Bandwidth extension
- Declipping
- Neural vocoding
- Source separation

UnDiff showcases the latent potential in solving general inverse problems for speech processing under the premise of **Unsupervised Learning**.

Diffusion Models



Algorithm 3 PC sampling (VP SDE)

$\mathbf{x}_N \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $i = N - 1$ **to** 0 **do**
  $\mathbf{x}'_i \leftarrow (2 - \sqrt{1 - \beta_{i+1}})\mathbf{x}_{i+1} + \beta_{i+1}\mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_{i+1}, i + 1)$
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  $\mathbf{x}_i \leftarrow \mathbf{x}'_i + \sqrt{\beta_{i+1}}\mathbf{z}$ — Predictor
  for $j = 1$ **to** M **do** — Corrector
    $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
    $\mathbf{x}_i \leftarrow \mathbf{x}_i + \epsilon_i \mathbf{s}_{\boldsymbol{\theta}^*}(\mathbf{x}_i, i) + \sqrt{2\epsilon_i}\mathbf{z}$
return $\mathbf{x}_0$

cite: Score-Based Generative Modeling through Stochastic Differential Equations

Connect normal distribution and data distribution through Markov chain or stochastic differential equation (SDE).

Diffusion Models

Estimate noise ε

$$\mathbb{E}_{\mathbf{x}\sim p_{data},\varepsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left(\lambda(t)\|\mathbf{s}_\theta(\mathbf{x}_t,t)-\nabla_{\mathbf{x}_t}\log p_t(\mathbf{x}_t)\|_2^2\right)$$

$$=\underbrace{\mathbb{E}_{\mathbf{x}\sim p_{data},\varepsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left(\lambda(t)\|\mathbf{s}_\theta(\mathbf{x}_t,t)-\frac{\varepsilon}{\sqrt{1-\bar{\alpha}(t)}}\|_2^2\right)}_{\text{in VP-SDE, }\mathbf{x}_t=\sqrt{\bar{\alpha}(t)}\mathbf{x}+\sqrt{1-\bar{\alpha}(t)}\varepsilon}$$

Inverse Problems with Diffusion Models

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

$$\Rightarrow \nabla_{x_t} \log p(x_t|y) = \nabla_{x_t} \log p(y|x_t) + \underbrace{\nabla_{x_t} \log p(x_t)}_{\text{Unconditional Diffusion Models}}$$

Inverse Problems with Diffusion Models

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{y}|\mathbf{x})p(\mathbf{x})}{p(\mathbf{y})}$$

$$\Rightarrow \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|\mathbf{y}) = \underbrace{\nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\mathbf{x}_t)} + \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t)$$

$$\approx \nabla_{\mathbf{x}_t} \log p(\mathbf{y}|\hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\bar{\alpha}(t)}}(\mathbf{x}_t - (1 - \bar{\alpha}(t))\mathbf{s}_\theta(\mathbf{x}_t)))$$

$$= -\xi(t)\nabla_{x_t} \|\mathbf{y} - A(\hat{\mathbf{x}}_0)\|$$

$A(\mathbf{x})$ is a differentiable degradation operator.

Speech Inverse Tasks: Bandwidth Extension

Frequency bandwidth extension (also known as audio super-resolution) can be viewed as a realistic restoration of waveform's high frequencies.

The observation operator is a lowpass filter

$$\mathbf{y} = A(\mathbf{x}) = \mathrm{LPF}(\mathbf{x})$$

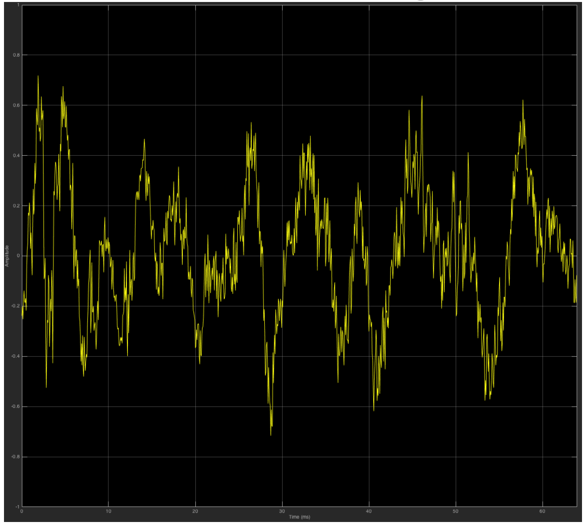
And apply data consistency steps, where the observed low frequencies are replaced with the denoised estimate following $\bar{\mathbf{x}}_0 = \mathbf{y} + \hat{\mathbf{x}}_0 - \mathrm{LPF}(\hat{\mathbf{x}}_0)$.

We can rewrite $\mathbf{s}_\theta$ using $\bar{\mathbf{x}}_0$

$$\Rightarrow \bar{\mathbf{s}}_\theta = \frac{1}{1-\bar{\alpha}(t)}\left(x_t - \sqrt{\bar{\alpha}(t)}\bar{x}_0\right)$$

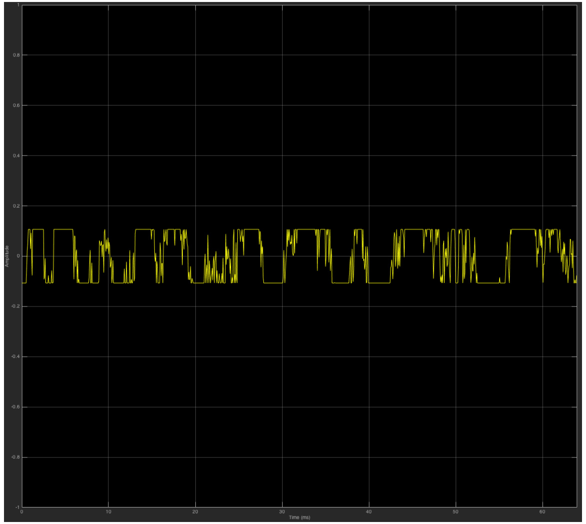
Speech Inverse Tasks: Declipping

What is clipping?



$$A(\mathbf{x}) = \mathbf{clip}(\mathbf{x})$$

$$= \frac{1}{2}(|\mathbf{x} + c| - |\mathbf{x} - c|)$$



cite: Audio Declipping

Speech Inverse Tasks: Neural Vocoding

The majority of modern speech synthesis systems decompose this task into two stages.

1. Low-resolution intermediate representations (e.g., linguistic features, mel-spectrograms) are predicted from text data.
2. Transform intermediate representations to raw waveform by **Neural Vocoder**.

Vocoder: $\mathcal{L} = ||\mathbf{x} - \text{Vocoder}(A(\mathbf{x}))||_2^2$

Issue of Supervised Vocoder

Vocoders based on different intermediate representations **CANNOT be Used Universally**.

Speech Inverse Tasks: Neural Vocoding

The majority of modern speech synthesis systems decompose this task into two stages.

1. Low-resolution intermediate representations (e.g., linguistic features, mel-spectrograms) are predicted from text data.
2. Transform intermediate representations to raw waveform by **Neural Vocoder**.

Degradation Operator of Vocoding Task

$$A(\mathbf{x}) = \mathrm{Mel}(\mathbf{x})$$

Speech Inverse Tasks: Source Separation

The goal of single-channel speech separation is to extract individual speech signals from a mixed audio signal, in which multiple INDEPENDENT speakers are talking simultaneously.

$$p(x^{(1)}, x^{(2)}|y) = p(x^{(1)}|y)p(x^{(2)}|y) = \frac{p(y|x^{(1)}, x^{(2)})p(x^{(1)})p(x^{(2)})}{p(y)}$$

$$\nabla_{x_t^{(1)}} \log p(x_t^{(1)}|y) = \nabla_{x_t^{(1)}} \log p(y|x_t^{(1)}, x_t^{(2)}) + \nabla_{x_t^{(1)}} \log p(x_t^{(1)})$$

Specifically, note that y depends only the sum of x1 and x2,

$$\Rightarrow y = x^{(1)} + x^{(2)} = \frac{1}{\sqrt{\bar{\alpha}(t)}}(x_t^{(1)} + x_t^{(2)}) - \sqrt{\frac{2(1-\bar{\alpha}(t))}{\bar{\alpha}(t)}}\varepsilon$$

$$\Rightarrow p(y|x_t^{(1)}, x_t^{(2)}) = p(y|x_t^{(1)} + x_t^{(2)}) = \mathcal{N}(y; \frac{1}{\sqrt{\bar{\alpha}(t)}}(x_t^{(1)} + x_t^{(2)}), \frac{2(1-\bar{\alpha}(t))}{\bar{\alpha}(t)})$$

Speech Inverse Tasks: Source Separation

$$p(y|x_t^{(1)}, x_t^{(2)}) = p(y|x_t^{(1)} + x_t^{(2)}) = \mathcal{N}(y; \frac{1}{\sqrt{\bar{\alpha}(t)}}(x_t^{(1)} + x_t^{(2)}), \frac{2(1-\bar{\alpha}(t))}{\bar{\alpha}(t)})$$

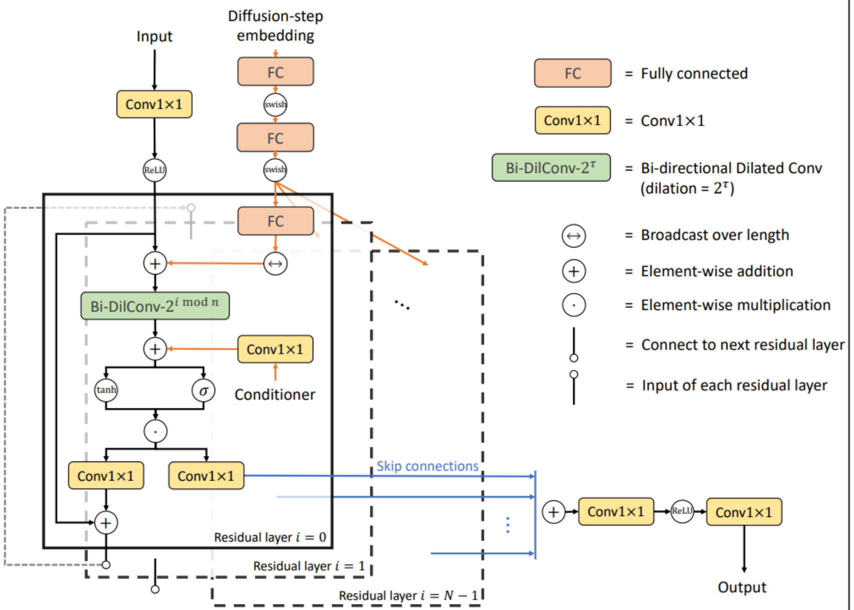
$$\log p(y|x_t^{(1)}, x_t^{(2)}) = -\frac{1}{2}\frac{\bar{\alpha}(t)(y-\frac{1}{\sqrt{\bar{\alpha}(t)}}(x_t^{(1)}+x_t^{(2)}))^2}{2(1-\bar{\alpha}(t))} + C$$

$$\Rightarrow \nabla_{x_t^{(1)}} \log p(y|x_t^{(1)}, x_t^{(2)}) = \frac{\sqrt{\bar{\alpha}(t)}(y - \frac{1}{\sqrt{\bar{\alpha}(t)}}(x_t^{(1)} + x_t^{(2)}))}{2(1-\bar{\alpha}(t))}$$

$\nabla_{x_t^{(1)}} \log p(y|x_t^{(1)}, x_t^{(2)})$ and $\nabla_{x_t^{(2)}} \log p(y|x_t^{(1)}, x_t^{(2)})$ can be derived without the need to compute the gradient of $\|\mathbf{y} - A(\mathbf{x}_t)\|_2^2$.

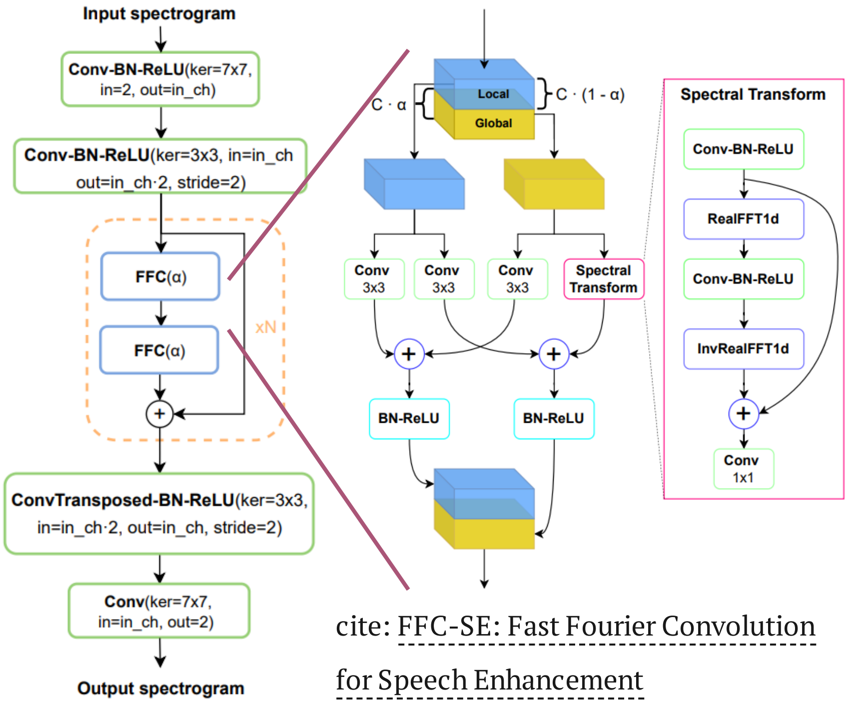
Architecture of Diffusion Model

Diffwave



cite: DiffWave: A Versatile Diffusion Model for Audio Synthesis

FFC-AE



cite: FFC-SE: Fast Fourier Convolution for Speech Enhancement

Experiments

Table 2: *Results of bandwidth extension (BWE) on VCTK.*

Model	Supervised	WV-MOS	LSD	MOS
Ground Truth	-	4.17	0	4.09 ± 0.09
BWE 2kHz →8kHz				
HiFi++ [15]	✓	**4.05**	1.09	$\mathbf{3.93 \pm 0.10}$
Voicefixer [31]	✓	3.67	1.08	3.64 ± 0.10
TFiLM [32]	✓	2.83	1.01	2.71 ± 0.10
UnDiff (Diffwave)	×	3.48	**0.96**	3.59 ± 0.11
UnDiff (FFC-AE)	×	3.59	1.13	3.50 ± 0.11
Input	-	2.52	1.06	2.42 ± 0.09
BWE 4kHz →8kHz				
HiFi++ [15]	✓	**4.22**	1.07	$\mathbf{4.04 \pm 0.10}$
Voicefixer [31]	✓	3.95	0.98	3.92 ± 0.10
TFiLM [32]	✓	3.46	0.83	3.43 ± 0.10
UnDiff (Diffwave)	×	4.00	**0.76**	3.74 ± 0.11
UnDiff (FFC-AE)	×	3.88	0.96	3.72 ± 0.10
Input	-	3.34	0.85	3.39 ± 0.10

Table 3: *Results of declipping (input SNR = 3 db) on VCTK.*

Model	Supervised	WV-MOS	SI-SNR	MOS
Ground Truth	-	3.91	-	3.84 ± 0.11
A-SPADE [33]	×	2.63	8.48	2.67 ± 0.11
S-SPADE [34]	×	2.69	8.50	2.55 ± 0.11
Voicefixer [31]	✓	2.79	-22.58	2.98 ± 0.12
Undiff (Diffwave)	×	**3.62**	**10.57**	$\mathbf{3.59 \pm 0.12}$
Undiff (FFC-AE)	×	3.01	7.35	3.06 ± 0.12
Input	-	2.30	3.82	2.19 ± 0.09

Table 4: *Results of neural vocoding (LJ speech dataset).*

Model	Supervised	WV-MOS	MOS
Ground Truth	-	4.32	4.26 ± 0.07
HiFi-GAN (V1) [18]	✓	**4.36**	$\mathbf{4.23 \pm 0.07}$
Diffwave [10]	✓	4.19	4.15 ± 0.07
Griffin-Lim [35]	×	3.30	3.46 ± 0.08
Undiff (Diffwave)	×	3.99	3.79 ± 0.08
Undiff (FFC-AE)	×	4.08	4.12 ± 0.07

Experiments: Source Separation

Table 5: *Results of source separation (VCTK dataset).*

Model	Supervised	SI-SNR	STOI
Mixture (input)	-	-0.04	0.69
Undiff (Diffwave)	×	5.73	0.79
Undiff (FFC-AE)	×	3.39	0.76
Conv-TasNet [36]	✓	**15.94**	**0.95**

Although Undiff is able to correctly separate voices in local regions, it mixes different voices within one sample.

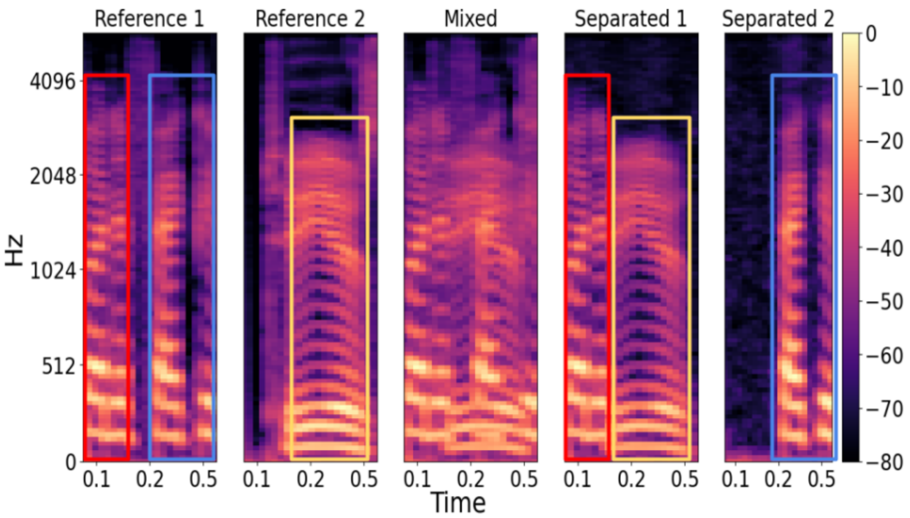


Figure 1: *Failure case of source separation with Undiff model.*

Conclusions

- The results highlight the potential of the **Unconditional Diffusion Models** to serve as general voice restoration tools.
- Enabling models to produce globally coherent voices during source separation could be An interesting directions for future work.