Noise-aware method for Speech Enhancement

reporter：朱清扬
date: 2021.08.12

CONTENTS

PART 01

Introduction

Introduction

Weiner Filtering

- the signal keeps stable
- Statistical properties remains the same.

VAE-NMF

Train for VAE-NMF

- NMF can just produce limited noise
- Standard VAE is sensitive to noise

NAE

PART 02

Basic methods and theoretical basis

Basic methods

Spectrum Subtraction

Spectral subtraction is based on a simple assumption: The noise in speech is only additive noise

$$\text{let } D(w) = P_s(w) - P_n(w)$$

$$P_s'(w) = \begin{cases} D(w), & \text{if } D(w) > 0 \\ 0, & \text{otherwise} \end{cases}$$

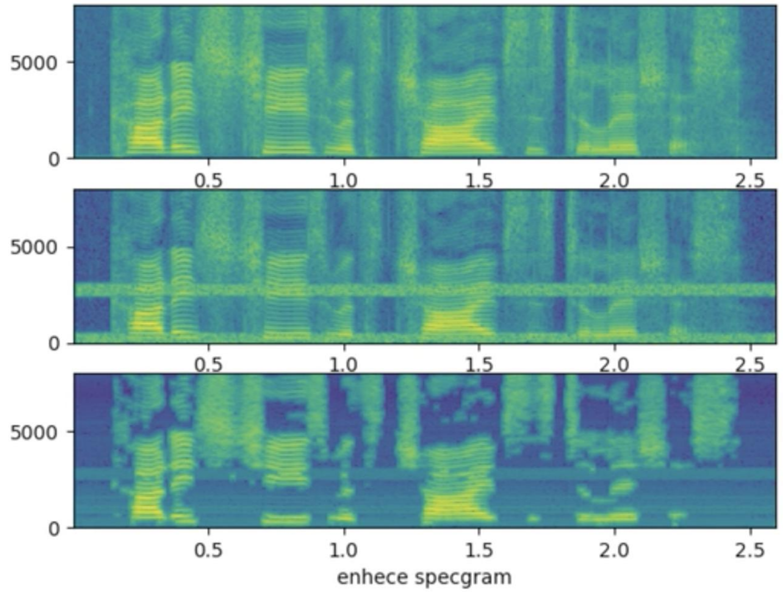
'music noise!' →

$$\text{let } D(w) = P_s(w) - \alpha P_n(w)$$

$$P_s'(w) = \begin{cases} D(w), & \text{if } D(w) > \beta P_n(w) \\ \beta P_n(w), & \text{otherwise} \end{cases}$$

$$\text{with } \alpha \geq 1, \text{ and } 0 < \beta << 1$$

$\boldsymbol{P_n}$: Generally, it is assumed that the first N frames of the input speech is the silence time, that is, there is no voice input during this time, only noise, which can be called the floor noise.



- $\boldsymbol{\alpha > 1}$: In this way, compared with the previous method, it can ensure a stronger denoising effect and can remove most of the noise, so that the residual noise will be much less.

- $\boldsymbol{0 < \beta \ll 1}$: The advantage of setting a lower limit is that the residual peak is less significant.

Basic methods

Wiener Filtering

- **simple assumption**: The observed signal X (k) and the random noise b (k) (independent of the source signal) are zero-mean and steady-state.

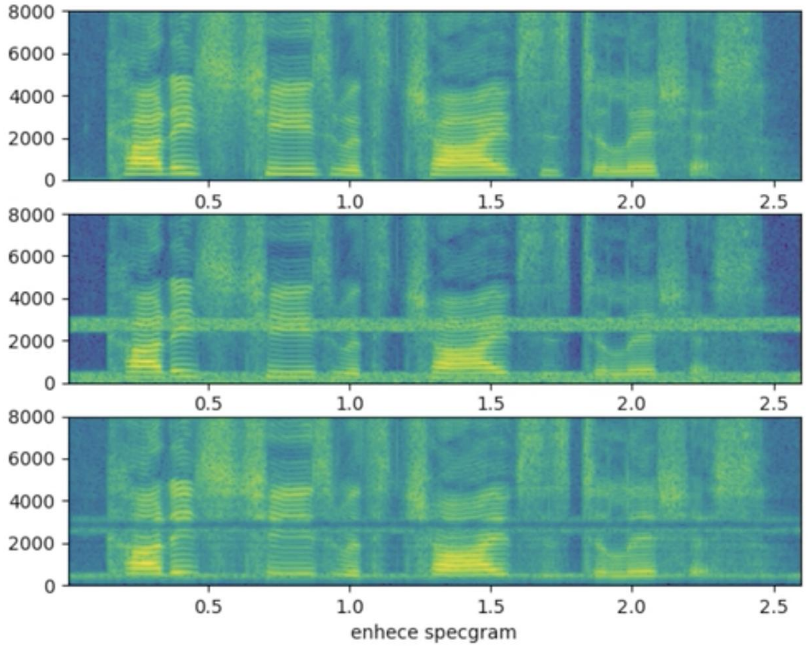
- **General function** for SISO model:_x0008_$x = s + n$; $s = speech, n = noise, x = noisy\ data$

  $$\hat{s} = h * x \quad ; \hat{s}:\ est.\ speech.$$

- **main object:** $min\ E(|\hat{s} - s|^2)$

  $$J = E(|s|^2) - hP_{xs} - \bar{h}\bar{P}_{xs} + |h|^2 P_{xx};$$

  $$P_{xs} = E(x\bar{s}); P_{xx} = E(|x|^2)$$



- Take the derivative with respect to y：

  $$\frac{\partial J}{\partial h} = \bar{h}P_{xx} - P_{xs} = [hP_{xx} - P_{xs}]^* = 0$$

  $$h = \frac{{P_{xs}}^*}{P_{xx}}$$

- In the case of speech enhancement :
  $x = s + n$, s and n are independent with zero means.

  $$P_{xs} = P_{ss}$$

  $$P_{xx} = P_{nn} + P_{ss}$$

VAE–NMF

Definitions:

- STFT domain: $(f,n) \in \{0,...,F-1\} \times \{0,...,N-1\}$
  $noisy\ data\ X = (x_{f,n})$
- $\{speech\ data\ S = (s_{f,n})$
  $noise\ data\ B = (b_{f,n})$
- Latent space: $\mathrm{Z} = (z_n),\ Z \in R^{L \times N}$

Aim:

To find the most appropriate parameters to the model:

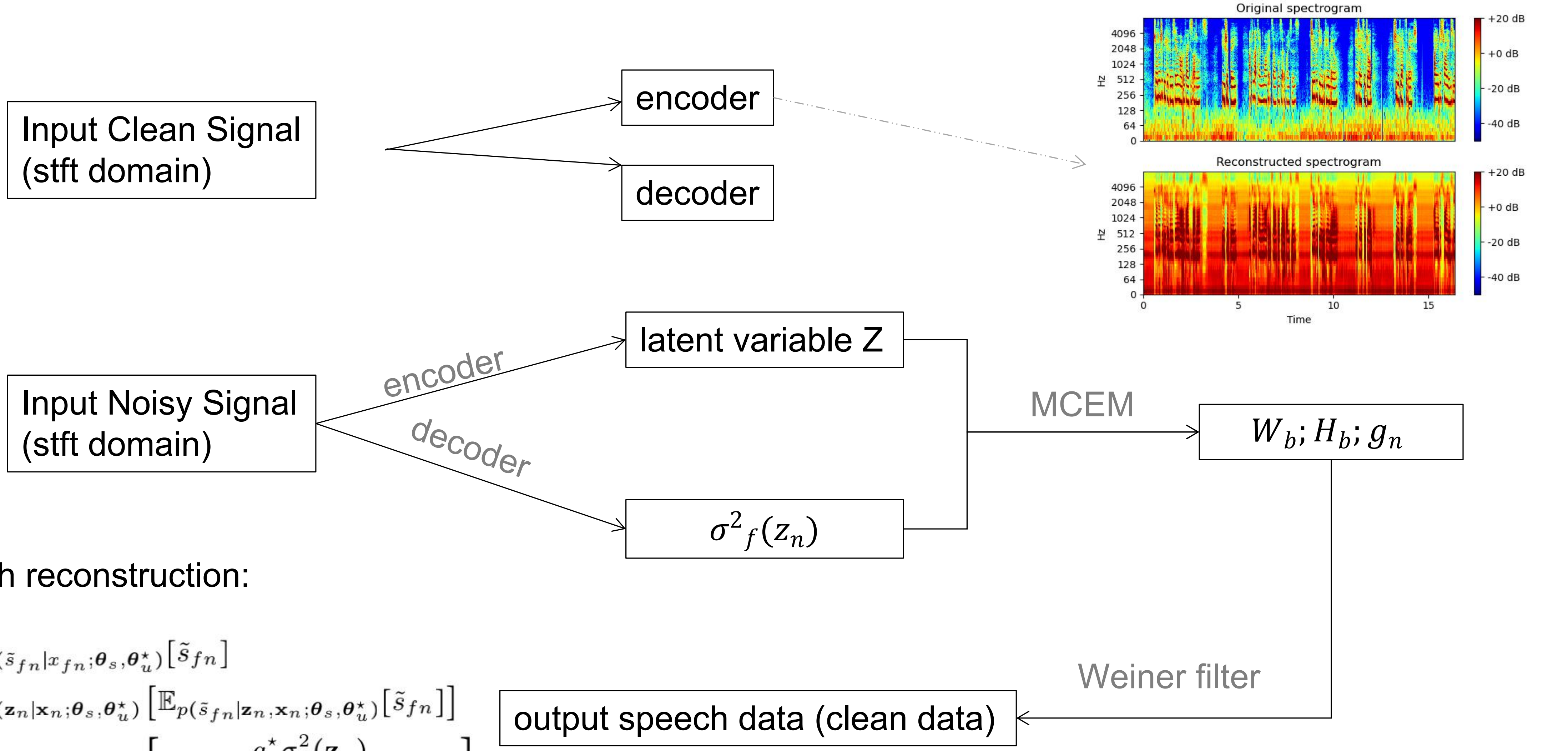
$$x_{f,n} = \sqrt{g_n} s_{f,n} + b_{f,n}\ .$$

Parameters:

- $s_{f,n} \sim N(0, \sigma^2{}_f(z_n)) : \sigma^2{}_f(z_n)$ *is defined by a Decoder;*
- $b_{f,n} \sim N(0, (W_b H_b)_{f,n})\ : W_b H_b$ *is a NMF of the noise vector;*
- $\sqrt{g_n}$ *: represents a frame-dependent but frequency-independent gain.*

VAE–NMF

baseline



speech reconstruction:

$$
\begin{aligned}
\hat{\tilde{s}}_{fn} &= \mathbb{E}_{p(\tilde{s}_{fn}|x_{fn};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)}[\tilde{s}_{fn}] \\
&= \mathbb{E}_{p(\mathbf{z}_n|\mathbf{x}_n;\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)}\left[\mathbb{E}_{p(\tilde{s}_{fn}|\mathbf{z}_n,\mathbf{x}_n;\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)}[\tilde{s}_{fn}]\right] \\
&= \mathbb{E}_{p(\mathbf{z}_n|\mathbf{x}_n;\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)}\left[\frac{g_n^\star\sigma_f^2(\mathbf{z}_n)}{g_n^\star\sigma_f^2(\mathbf{z}_n)+(\mathbf{W}_b^\star\mathbf{H}_b^\star)_{f,n}}\right]x_{fn}.
\end{aligned}
$$

VAE–NMF

MCEM

EM method

▷ E-Step: Compute $Q(\boldsymbol{\theta}_u;\boldsymbol{\theta}_u^\star)=\mathbb{E}_{p(\mathbf{z}|\mathbf{x};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)}[\ln p(\mathbf{x},\mathbf{z};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u)]$;

▷ M-Step: Update $\boldsymbol{\theta}_u^\star \leftarrow \arg\max_{\boldsymbol{\theta}_u} Q(\boldsymbol{\theta}_u;\boldsymbol{\theta}_u^\star)$.

Problem: $p(z|x,\theta_s,\theta_u)$ intractable!

MH method:

- proposal distribution: $\mathbf{z}_n \mid \mathbf{z}_n^{(m-1)};\epsilon^2 \sim \mathcal{N}(\mathbf{z}_n^{(m-1)},\epsilon^2\mathbf{I})$.
- acception: $\alpha=\min\left(1,\dfrac{p(\mathbf{x}_n\mid\mathbf{z}_n;\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star)\,p(\mathbf{z}_n)}{p\left(\mathbf{x}_n\mid\mathbf{z}_n^{(m-1)};\boldsymbol{\theta}_s,\boldsymbol{\theta}_u^\star\right)p\left(\mathbf{z}_n^{(m-1)}\right)}\right)$,

M-Step:
To maximize the Q function:

$$\mathbf{H}_b \leftarrow \mathbf{H}_b \odot \left[\frac{\mathbf{W}_b^\top\left(|\mathbf{X}|^{\odot 2}\odot\sum_{r=1}^{R}\left(\mathbf{V}_x^{(r)}\right)^{\odot -2}\right)}{\mathbf{W}_b^\top\sum_{r=1}^{R}\left(\mathbf{V}_x^{(r)}\right)^{\odot -1}}\right]^{\odot 1/2}$$

$$\mathbf{W}_b \leftarrow \mathbf{W}_b \odot \left[\frac{\left(|\mathbf{X}|^{\odot 2}\odot\sum_{r=1}^{R}\left(\mathbf{V}_x^{(r)}\right)^{\odot -2}\right)\mathbf{H}_b^\top}{\sum_{r=1}^{R}\left(\mathbf{V}_x^{(r)}\right)^{\odot -1}\mathbf{H}_b^\top}\right]^{\odot 1/2}$$

$$\mathbf{g}^\top \leftarrow \mathbf{g}^\top \odot \left[\frac{\mathbf{1}^\top\left[|\mathbf{X}|^{\odot 2}\odot\sum_{r=1}^{R}\left(\mathbf{V}_s^{(r)}\odot\left(\mathbf{V}_x^{(r)}\right)^{\odot -2}\right)\right]}{\mathbf{1}^\top\left[\sum_{r=1}^{R}\left(\mathbf{V}_s^{(r)}\odot\left(\mathbf{V}_x^{(r)}\right)^{\odot -1}\right)\right]}\right]^{\odot 1/2}$$

distributions in model:

- latent $\mathbf{z}_n \sim \mathcal{N}(\mathbf{0},\mathbf{I})$;
- speech $s_{fn}\mid\mathbf{z}_n;\boldsymbol{\theta}_s \sim \mathcal{N}_c(0,\sigma_f^2(\mathbf{z}_n))$,
- noise $b_{fn};\mathbf{w}_{b,f},\mathbf{h}_{b,n} \sim \mathcal{N}_c\left(0,(\mathbf{W}_b\mathbf{H}_b)_{f,n}\right)$
- mixed voice
  $x_{fn}\mid\mathbf{z}_n;\boldsymbol{\theta}_s,\boldsymbol{\theta}_{u,fn} \sim \mathcal{N}_c\left(0,g_n\sigma_f^2(\mathbf{z}_n)+(\mathbf{W}_b\mathbf{H}_b)_{f,n}\right)$

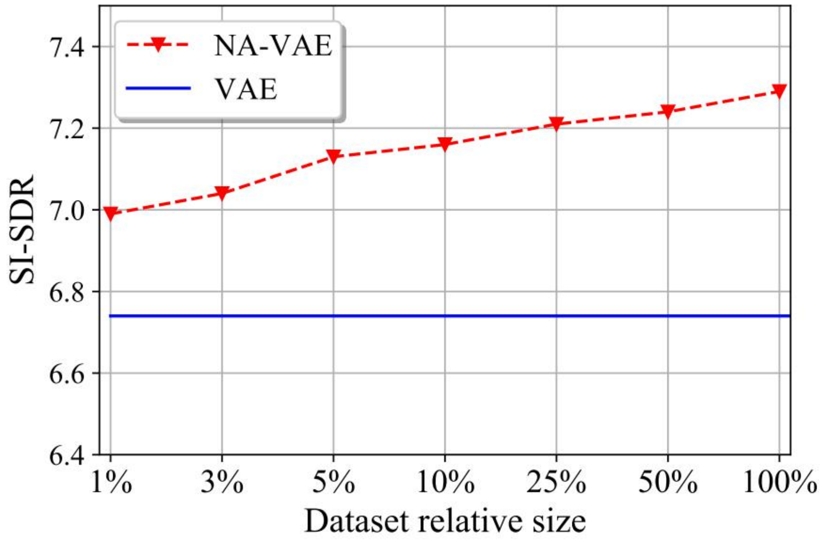
NAE

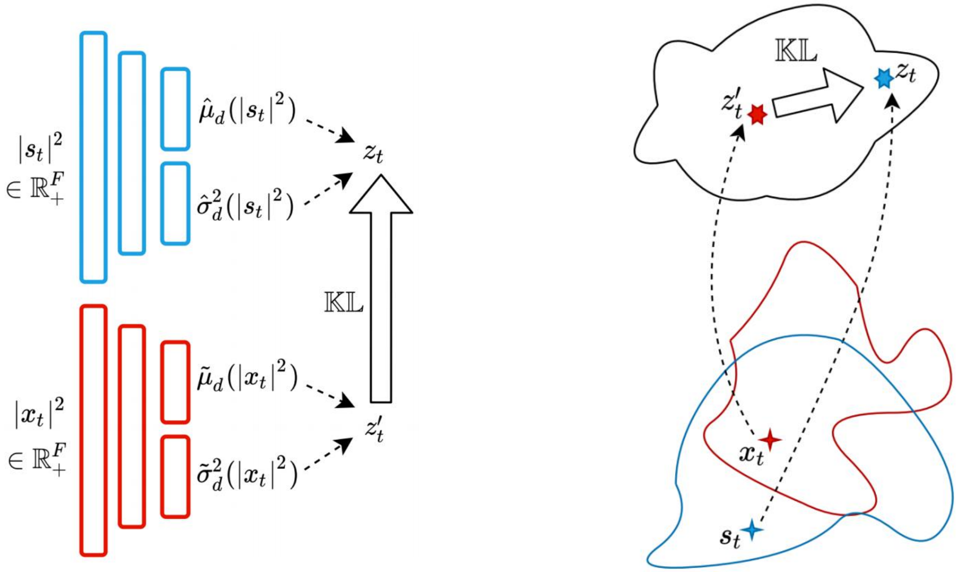
Variational Autoencoder for Speech Enhancement with a Noise–Aware Encoder

Two-step training (for encoder)**:**

- Based on speech data, train an encoder, with the latent space Z.
- Based on noisy data, train the encoder to minimize:

$$\mathcal{L}(\gamma) = \sum_t \mathbb{KL}(q_\phi(z_t|s_t)||q'_\gamma(z'_t|x_t))$$
$$= \sum_{t,d} \left\{ \frac{1}{2} \log \frac{\widetilde{\sigma}_d^2(|x_t|^2)}{\hat{\sigma}_d^2(|s_t|^2)} - \frac{1}{2} + \frac{\hat{\sigma}_d^2(|s_t|^2) + (\hat{\mu}_d(|s_t|^2) - \widetilde{\mu}_d(|x_t|^2))^2}{2\widetilde{\sigma}_d^2(|x_t|^2)} \right\}$$





- A technique to get the latent space: initialize the encoder with the trained data.
- Decoder got from traditional VAE, since the optimal mapping between latent space to clean data is difficult to get.

PART 03

Experiment results

Results

Data when train and test:

- clean_trainset_wav_16k: 11572pieces
- noisy_trainset_wav_16k: 11572 pieces
- clean(noisy)_testset_wav_16k： 2878pieces

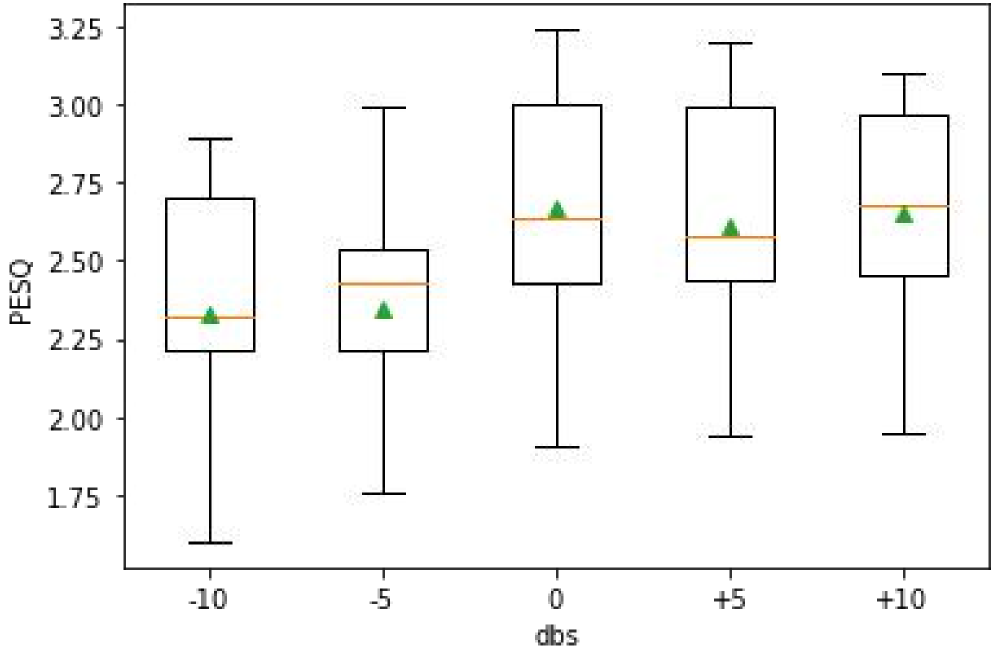


Figure 1. Performance comparison on PESQ with NAE on 5 different SNR cases.

SI-SDR	-10	-5	0	+5	+10
NAE	11.66	12.09	12.67	13.17	12.93
VAE-NMF	10.93	13.17	12.36	13.18	12.59
SS	7.89	10.81	10.90	11.19	11.01
WF	8.07	10.24	11.32	11.27	11.19
joint	10.90	12.13	12.19	12.38	12.56

Figure 2. Performance comparison on SI-SDR on 5 different SNR cases and trained and evaluated on 5 similar methods.

Results

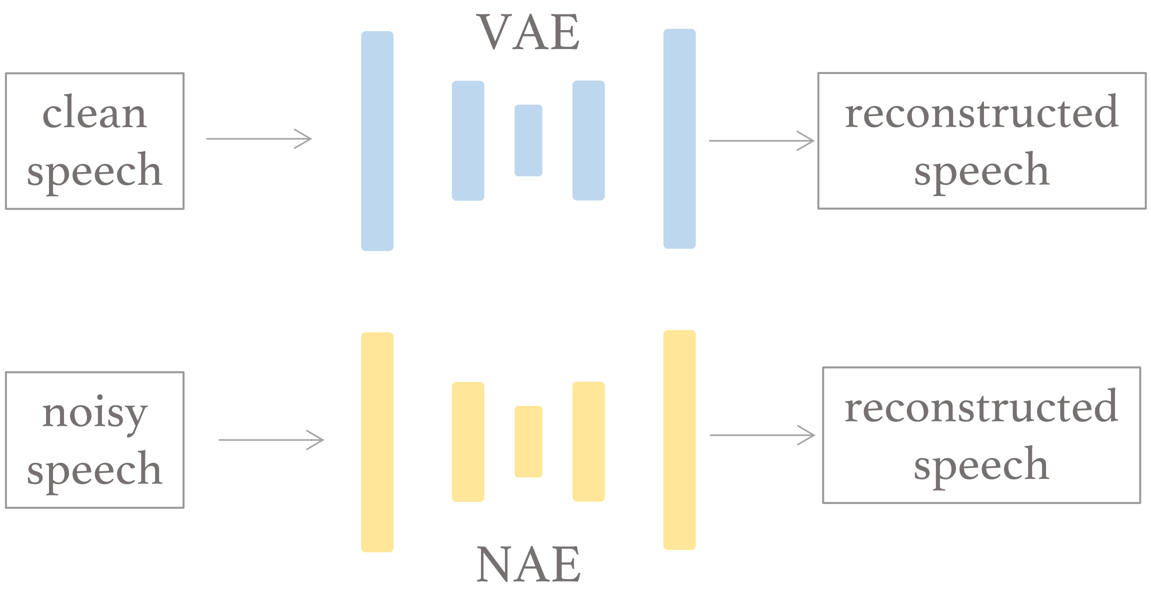
Data:

- clean_testset_wav_16k add white noise: 2878 pieces

SI-SDR	-10	-5	0	+5	+10
NAE	10.98	12.76	12.18	12.98	13.32
VAE-NMF	10.65	11.45	12.32	13.24	12.95
SS	6.77	8.29	8.78	8.91	10.03
WF	8.11	8.24	9.45	9.27	9.19
joint	10.18	11.25	11.98	12.87	13.01

Figure 3. Performance comparison on SI-SDR on 5 different SNR cases and trained and evaluated on 5 similar methods.

Joint Training



$$loss = loss_{vae} + loss_{nae} + KL(z_{vae}|z_{nae})$$

THANKS FOR YOUR WATCHING

汇报人：朱清扬