



语音信号处理：第九章作业讲解



主讲人 彭震东



作业题目

1、计算题：

给定目标信号 `target_signal.wav`，模型输出信号 `estimate_signal.wav`，请编程实现 SI-SDR 的计算。

2、实践题：

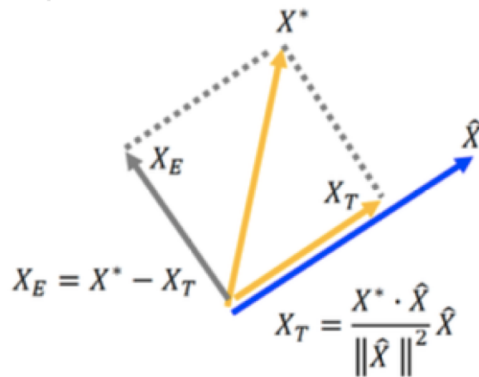
语音分离技术的迭代速度比较快，且拥有良好的开源生态，方便大家选择一个开源框架快速上手。本题目是一个半开放类型的题目，目的是让大家自己动手实现一个或几个经典的语音分离算法，加深对该问题的理解。

Asteroid 是比较常用的基于 pytorch 的语音分离工具包。请大家熟悉 Asteroid 的基本框架，并且任选其中一个 recipe，训练一个分离模型，在开源数据集（推荐使用 WSJ0-2mix）上观察分离效果。

计算题

- Pytorch or Tensorflow
- $X^* \cdot \hat{X}$ 点积得到标量，即需 sum

SI-SDR:



$$SISDR = 10 \log_{10} \frac{\|X_T\|^2}{\|X_E\|^2}$$

```
1 import torch
2 from itertools import permutations
3 import numpy as np
4 import soundfile as sf
5
6 def sisnr(x, s, eps=1e-8):
7     """
8     calculate training loss
9     input:
10     x: separated signal, N x S tensor
11     s: reference signal, N x S tensor
12     Return:
13     sisnr: N tensor
14     """
15
16     def l2norm(mat, keepdim=False):
17         return torch.norm(mat, dim=-1, keepdim=keepdim)
18
19     if x.shape != s.shape:
20         raise RuntimeError(
21             "Dimension mismatch when calculate si-snr, {} vs {}".format(
22                 x.shape, s.shape))
23     x_zm = x - torch.mean(x, dim=-1, keepdim=True)
24     s_zm = s - torch.mean(s, dim=-1, keepdim=True)
25     t = torch.sum(x_zm * s_zm, dim=-1, keepdim=True) * s_zm / (l2norm(s_zm, keepdim=True)**2)
26     return 20 * torch.log10(l2norm(t) / (l2norm(x_zm - t)))
27
28 if __name__ == "__main__":
29     x, _ = sf.read('estimate_signal.wav', dtype='float32')
30     s, _ = sf.read('target_signal.wav', dtype='float32')
31     ests = torch.from_numpy(x)
32     egs = torch.from_numpy(s)
33
34     snr_new = sisnr(ests, egs)
35     print('SI-SDR:', snr_new)
```

- 使用 ConvTasNet 的 Recipe:
 - Librimix: 数据集 LibriMix (LibriSpeech + Wham)
 - Wham: 数据集 WSJMix (WJ0 + Wham)
- egs/librimix/convTasNet
 1. 数据下载与生成
 2. 模型训练和评估

```
. local/prepare_data.sh --storage_dir data
```

```
./run.sh --stage 2
```

• by 鄢聪

感谢各位聆听
Thanks for Listening

