



# 南京大學

## 本科畢業論文

院 系 物理學院

專 業 聲科學與工程系

題 目 強化學習在單通道語音增強系統

的應用研究

年 級 2017 級 學 號 171840582

學生姓名 盧嘉偉

指導老師 盧晶 職 稱 教授

提交日期 2021 年 5 月 28 日



# 南京大学本科生毕业论文(设计、作品)中文摘要

题目：强化学习在单通道语音增强系统的应用研究

院系：物理学院

专业：声科学与工程系

本科生姓名：卢嘉伟

指导老师（姓名、职称）：卢晶教授

摘要：

本文研究了强化学习在单通道语音增强系统中的应用。以 Deep Xi 系统为例，该系统基于深度学习方法给出带噪语音的先验信噪比估计值，利用最小均方误差实现语音增强的功能。该系统需要大量的数据对内部的深度神经网络进行训练以提高语音增强效果。然而在现实中，收集数量庞大的训练数据往往是困难的。为了有效地利用已有数据，我们尝试实现 Deep Xi 系统的自优化。强化学习作为一种自优化方法，近年来已在众多领域得到成功应用。其中 Q-Learning 算法是一种贴合语音增强场景的无模型强化学习算法。本文基于 Deep Xi 系统的特征，建立了相适配的 Q-Learning 模型，设计了名为 Xi-Q 的强化学习算法。Xi-Q 算法以客观语音质量评估 (PESQ) 为基准设计奖励函数，利用 Deep Xi 系统输出的先验信噪比估计值给定一组动作，让系统在  $\epsilon$ -greedy 策略下进行动作选择，通过平衡“探索-利用” (exploration-exploitation) 来提升最大期望奖励。实验结果表明，本文所提出的 Xi-Q 算法能有效地改善 Deep Xi 系统语音增强的效果，实现系统的自我学习与优化。

关键词：强化学习；语音增强；Deep Xi；噪声估计；最小均方误差

# 南京大学本科生毕业论文(设计、作品) 英文摘要

THESIS: Application of Reinforcement Learning in Single-Channel Speech Enhancement System

DEPARTMENT: School of Physics

SPECIALIZATION: Acoustics

UNDERGRADUATE: Jiawei Lu

MENTOR: Professor Jing Lu

ABSTRACT:

This thesis studies the application of reinforcement learning in single-channel speech enhancement system. Taking Deep Xi as an example, a prior SNR estimation of noisy speech is given by the system based on deep learning, and the minimum mean square error (MMSE) is used to realize speech enhancement. For the purpose of improving the result of speech enhancement, the system needs a lot of data to train the deep neural network. However, it is often difficult to collect large amounts of training data. In order to effectively utilize the existing data, we try to realize self-optimization of Deep Xi. Reinforcement learning, as a self-optimization method, has been successfully applied in many fields in recent years. Q-Learning is a model-free reinforcement learning algorithm, which is suitable for the scenario of speech enhancement. This thesis establishes an adaptive Q-Learning model based on the characteristics of Deep Xi, and designs a reinforcement learning algorithm named Xi-Q. The Xi-Q algorithm uses Perceptual Evaluation of Speech Quality (PESQ) to design the reward function, and utilizes the estimated a prior SNR calculated by Deep Xi system to give a set of actions. It selects the actions under the  $\epsilon - greedy$  strategy to increase the maximum expected reward by balancing "exploration-exploitation". Experiments show that the proposed Xi-Q algorithm can effectively improve the speech enhancement results of Deep Xi system.

KEY WORDS: Reinforcement learning, Speech enhancement, Deep Xi, Noise estimation, Minimum mean square error

# 目 录

<b>1</b>	<b>绪论</b>	<b>1</b>
1.1	研究背景	1
1.2	相关工作	1
1.3	本文主要工作	4
1.4	本文结构	4
<b>2</b>	<b>Deep Xi 框架</b>	<b>7</b>
2.1	基于 MMSE 的噪声功率谱密度估计	7
2.2	利用 Deep Xi 估计先验信噪比	8
2.2.1	第一阶段, 输出先验信噪比映射版本的估计值 $\hat{\xi}_l$	9
2.2.2	第二阶段, 计算先验信噪比估计 $\hat{\xi}_l$	10
<b>3</b>	<b>Quality-Net 框架</b>	<b>11</b>
3.1	Quality-Net 的基本结构	11
3.2	Quality-Net 的实现细节	12
3.2.1	基本参数	12
3.2.2	目标函数	12
3.2.3	遗忘门偏置	12
<b>4</b>	<b>Xi-Q 算法</b>	<b>15</b>
4.1	Q-Learning 的基本框架	15
4.2	适配 Deep Xi 的 Q-Learning 模型	16
4.3	Xi-Q 算法	17
4.3.1	定义参数	17
4.3.2	具体内容	18
<b>5</b>	<b>实验</b>	<b>23</b>
5.1	实现细节	23
5.1.1	信号处理	23
5.1.2	训练集	23
5.1.3	测试集	23
5.1.4	训练 Deep Xi 中的 resnet-1.1n	24
5.1.5	训练 Quality-Net	24
5.1.6	执行 Xi-Q 算法	25
5.2	验证基于 Quality-Net 的 Xi-Q 算法结果	25
5.3	验证基于真实 PESQ 数值的 Xi-Q 算法结果	27
5.3.1	对 Deep Xi 系统整体性能的提升	27

---

5.3.2	对 Deep Xi 系统输出结果较差的特定带噪语音有很 好的处理结果 .....	28
<b>6</b>	<b>总结与讨论 .....</b>	<b>31</b>
6.1	总结 .....	31
6.2	讨论 .....	31
6.2.1	优化 Quality-Net .....	31
6.2.2	优化奖励函数 .....	32
6.2.3	优化先验信噪比模版 .....	32
	<b>参考文献 .....</b>	<b>33</b>
	<b>致 谢 .....</b>	<b>37</b>

# 第一章 绪论

## 1.1 研究背景

语音增强 (speech enhancement) 主要是利用音频信号处理技术来改善带噪语音的可理解性 (intelligibility) 和整体感知质量 (perceived quality)。现实中很多场景下我们需要对语音信号进行增强, 比如在嘈杂环境中的录音, 或者受到通讯信道噪声污染的语音。语音增强算法能在某种程度上减轻或抑制背景噪声, 提高人耳对带噪语音的感知度。语音增强算法按照传感器的数量可以分为单通道<sup>[1-3]</sup> 和多通道<sup>[4]</sup> 两类。

强化学习 (reinforcement learning, i.e. RL) 是机器学习中的一个领域, 强调智能体 (agent) 如何基于环境而行动, 以取得最大化的预期利益, 其灵感来源于心理学中的行为主义理论。强化学习是除了监督学习和非监督学习之外的第三种基本的机器学习方法, 其关注点在于平衡已有知识的利用和未知动作的探索<sup>[5]</sup>。

本文主要研究强化学习在单通道语音增强系统中的应用。

## 1.2 相关工作

到目前为止, 大部分单通道语音增强算法都需要估算噪声的功率谱密度 (power spectral density, i.e. PSD)。对噪声的功率谱密度过高的估计会导致还原后的语音信号失真, 而过低的估计又会使得增强后的语音中存在大量残留噪声。因此, 对噪声功率谱密度的准确估计是提升语音增强效果的关键之处, 研究者们也在一直尝试和改进不同的算法。

在早些时候, 人们利用语音活动检测器<sup>[6-8]</sup> (voice activity detector, i.e. VAD), 根据语音活动缺失时的频谱来估计噪声的功率谱密度。这些方法在噪声相当平稳的条件下是有效的, 但对于非平稳的噪声源, 语音活动检测器的跟踪效果将变得不尽人意。除此之外, 在低信噪比 (signal to noise ratio, i.e. SNR) 的条件下实现精准的语音活动检测直到现在还是一个相当困难的任务。

之后, 出现了一种利用最小统计量 (minimum statistics, i.e. MS)<sup>[9]</sup> 的方法来跟踪噪声。该方法的出发点是观察到即使在语音存在的情况下, 带噪语音的功率在每个频率段内常常会衰减到噪声的功率级。因此, 在有限的时间窗内记录每个频率段中带噪语音功率的最小值, 乘之一个修正系数, 得到最终的噪声功率谱密度估计<sup>[10]</sup>。其中, 有限时间窗长度的选择是影响 MS 方法性能的一个重要参数。如果窗长度选择过大, 系统会出现显著的跟踪延迟, 特别是在噪声的功率快速增加时。反之, 如果窗长度选择过小, 该方法可能会转而跟踪原带噪语音的功率谱密度, 导致对噪声功率谱密度的过高估计。

接着, 时间递归平均算法 (timerecursive averaging algorithms) 被作为另一种思路来估计噪声的功率谱密度。以最小控制递归平均-2 (minima controlled recursive averaging-2, i.e. MCRA-2) 方法<sup>[11]</sup> 作为该算法的一个例子, 它通过使用一个平滑参数来递归平均先前估计的噪声功率谱密度和当前噪声周期图, 实现噪声功率谱密度估计的更新。其中这个平滑参数由语音存在概率 (speech presence probability, i.e. SPP) 调整。MCRA-2 在计算 SPP 时, 首先通过连续谱最小跟踪技术<sup>[12]</sup> (continuous spectral minimum tracking technique) 得到带噪语音的局部最小值, 然后取平滑后的噪声频谱与这个局部最小值的比值, 进行阈值分割, 最终得到 SPP 估计。由于时间递归平均算法也依赖于频谱中的最小值跟踪, 所以这类算法也较难跟踪功率快速变化的噪声。

近年来, 从贝叶斯统计角度提出的一类基于最小均方误差 (minimum mean square error, i.e. MMSE) 的噪声功率谱密度估计方法被广泛应用。这类算法通过计算先验信噪比 (a priori SNR) 和后验信噪比 (a posteriori SNR), 得出噪声周期图的最小均方误差估计。在下一章的推导中我们可以看到, 此类算法对于噪声功率谱密度估计的准确度主要取决于先验信噪比估计的精准与否。一开始提出的先验信噪比的估计方法, 例如决策导向<sup>[13]</sup> (decision-directed, i.e. DD) 方法和最大似然<sup>[14]</sup> (maximum-likelihood, i.e. ML) 方法, 都假定了噪声的变化速度比语音信号慢。因此, 初期基于 MMSE 的算法能够跟踪中等非平稳噪声源, 但无法跟踪高度非平稳噪声源。最近, 随着深度学习 (deep learning) 的发展, 基于深度神经网络 (deep neural network, i.e. DNN) 的语音增强方法被不断研究和优化。Deep Xi<sup>[15]</sup> 便是其中的一种基于深度学习来估计先验信噪比的方法。Deep Xi 方法利用残差长短时记忆递归神经网络 (residual long short-term memory recurrent neural network, i.e. ResLSTM RNN), 输入带噪语音的幅度谱, 输出每一帧信号对应的先验信噪比估计。由于 Deep Xi 没有做任何关于语音或噪声特征的假设, 所以该方法对功率快速变化的非平稳噪声能及时跟踪。为了提高基于 Deep Xi 方法输出先验信噪比估计的精度, 需要大量的语音数据来训练深度神经网络。这些训练数据由纯净语音信号, 噪声信号和监督的先验信噪比组成。然而, 在实际应用中, 采集大量的训练数据往往比较困难, 会浪费大量的时间与资源。一旦训练数据的总体数量不足或者覆盖广度不够, 深度神经网络便不能得到有效的训练, 这会直接阻碍语音增强效果的提高。为了有效地使用有限的训练数据, 让 Deep Xi 中深度神经网络自优化 (self-optimize) 便成为一个很好的思路。

强化学习作为一种自优化方法, 已在众多领域得到成功应用, 包括机器人控制 (robot control), 电梯调度 (elevator scheduling), 计算机游戏等, AlphaGo 便是深度强化学习 (deep reinforcement learning) 发展的一个里程碑式的象征。在强化学习算法中, 我们需要定义奖励 (reward), 而不是定义监督的输出。奖励表示智能体 (agent) 在特定的环境 (environment) 或状态 (state) 中所采用策略 (strategy) 或行动 (action) 的有效性。一个很好的例子就是, 如果我们强化学习的目标是让智能体玩好一款计算机游戏, 那么我们通



过设计明确的奖励函数（reward function），对处于特定状态的智能体作出的下一步的动作选择给予奖励或惩罚，让智能体在不断试错的过程中学习，实现其自我优化。

Yuma Koizumi 等人针对一个以时频掩膜（time-frequency mask, i.e T-F Mask）为输出的神经网络，设计了基于语音质量感知评价（perceptual evaluation of speech quality, i.e. PESQ）的奖励函数，并从输出信号客观与主观的质量验证了该方法的有效性<sup>[16]</sup>。然而，该论文中作为对照的深度神经网络结构较为简单，仅有两层隐藏层，共 256 个神经元，与当下用来做语音增强的相关深度神经网络的规模大约差了两个数量级，因此强化学习的提升结果不具有很强的说服力。除此之外，文中 PESQ 的获取依赖于作为监督数据的纯净语音信号，因此要训练另一个深度神经网络（文中称为 *DNN - RL*）来作为强化学习动作-价值函数的映射，这将增大整个算法所需的数据量以及深度神经网络的训练时间。从这点可以看出，设计出一个合理的无参考源（non-intrusive）的客观语音质量评价能很大程度上提高该强化学习算法的有效性。

ITU-T P.563 是 ITU-T 在 2004 年提出的第一个无参考源的语音客观评价方法<sup>[17]</sup>。该方法考虑了公用电话交换网中各种失真情况，可以根据基于感知的测度 MOS-LQO（mean opinion score - listening quality objective）对语音质量进行预测。然而，ITU-T P.563 是专门为 3.1 kHz 的窄带电话所设计的，其应用有很大的局限性。

NISQA 是基于深度学习框架的语音质量预测方法<sup>[18]</sup>。该方法可用于预测通过通信系统（如电话或视频通话）发送的语音样本的质量。除了整体语音质量外，NISQA 还提供了对噪声，连续性和响度等质量维度的预测。虽然该方法相较于 ITU-T P.563 有了更广泛的应用范围，但其本质上还是基于语音通信网络来设计的，不具有很好的普适性。

最近，Szu-Wei Fu 等人提出了一种估计 PESQ 的深度学习方法，被称为 Quality-Net<sup>[19]</sup>。Quality-Net 是一种基于双向长短时记忆网络（bidirectional long short-term memory, i.e. BLSTM）的端到端（end-to-end）的无参考语音质量评价模型。该方法中对于 PESQ 的估计是帧级的，最后得出的客观语音质量评价是所有帧上的平均。帧级约束（frame constraints）和遗忘门偏差（forget gate bias）的合理初始化被应用于 Quality-Net，目的是从带 PESQ 标签的话语级别（utterance level）的监督数据中学习更加准确的帧级（frame level）的客观语音质量评估。实验结果表明，Quality-Net 的客观语音质量估计值与 PESQ 具有较高的相关性（带噪语音为 0.9，经过语音增强处理的语音为 0.84）。

从上述研究背景可见，理论上基于 DNN 和 MMSE 的 Deep Xi 系统也可以通过如图 1-1 的强化学习模型进行自我优化。

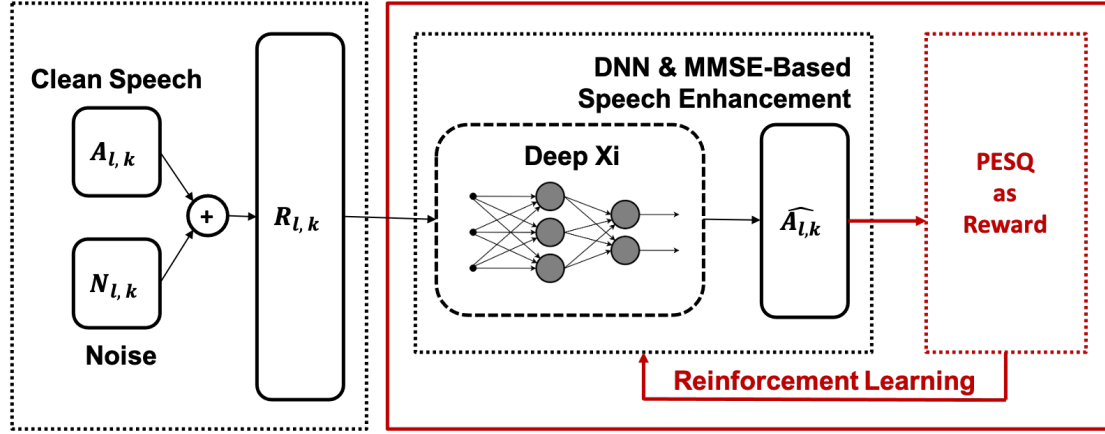


图 1-1: 强化学习模型在语音增强中的应用场景

### 1.3 本文主要工作

本文旨在通过强化学习算法实现 Deep Xi 系统语音增强结果的自优化，设计了 Xi-Q Algorithm。本文主要工作内容如下：

1. 基于 Deep Xi 系统输出的先验信噪比估计值，给定 Xi-Q 算法内的动作模板（template）；
2. 基于强化学习中的 Q-学习（Q-Learning）算法，结合 Deep Xi 系统的特征，建立适配 Deep Xi 系统的 Q-Learning 模型；
3. 基于 Quality-Net 估计客观语音质量感知评价 PESQ 的数值，加之局部错误动作惩罚，设计 Xi-Q 算法选择先验信噪比估计值的奖励函数；
4. 基于矩阵的线性映射功能，给定 Deep Xi 从输入信号到输出先验信噪比估计之间的动作-价值函数  $Q$ 。
5. 基于对数谱距离（log spectral distance, i.e. LSD），验证所提出的基于强化学习与 Quality-Net 的 Deep Xi 系统自优化算法 Xi-Q Algorithm 的有效性。
6. 基于给定准确的 PESQ 数值而设计的奖励函数，表明 Quality-Net 的可优化性，并验证理想 PESQ 数值估计下 Xi-Q Algorithm 所能达到的理论优化效果。

### 1.4 本文结构

本文的各章节组织结构如下：

第一章：绪论。简要说明了语音增强和强化学习的研究来由及相关研究，概述了这篇文章的工作，总结了本文结构。

第二章：Deep Xi 框架。简要概述和分析了基于深度学习的 Deep Xi 所使用的模型。

第三章：Quality-Net 框架。简要介绍了基于 BLSTM 的无参考源的客观语音质量评价的估计方法。

第四章：Xi-Q 算法。介绍了一个基于 Deep Xi 的系统自优化的强化学习算法。

第五章：实验。介绍了实验进行过程中环境的配置，各项参数的设置。列出了实验结果，验证了提出的 Xi-Q 算法的有效性。

第六章：总结与讨论。总结全文工作，讨论目前存在的问题和今后可以继续研究的方向。



## 第二章 Deep Xi 框架

Deep Xi 是一种基于 MMSE 的估计先验信噪比的深度学习方法<sup>[15]</sup>。

### 2.1 基于 MMSE 的噪声功率谱密度估计

在时域中，带噪语音信号由下式给出：

$$y[n] = s[n] + d[n] \quad (2-1)$$

其中  $n$  表示离散时间索引， $s[n]$  和  $d[n]$  分别表示纯净语音信号和不相关的加性噪声信号。

然后使用短时傅里叶变换 (short-time fourier transform, i.e. STFT) 对带噪语音按帧进行分析，得到：

$$Y_l[k] = S_l[k] + D_l[k] \quad (2-2)$$

其中， $l$  表示时间帧索引， $k$  表示离散频率索引， $Y_l[k]$ ， $S_l[k]$ ， $D_l[k]$  分别表示带噪语音、纯净语音和噪声的 STFT 系数。

我们应用基本假设， $S_l[k]$  和  $D_l[k]$  在时间与频率上是统计独立的，并且遵循均值为零，方差分别为  $E\{S_l(k)^2\} = \lambda_s[l, k]$ ， $E\{D_l(k)^2\} = \lambda_d[l, k]$  的条件高斯分布。其中  $E\{\cdot\}$  代表统计期望算子。

为方便起见，我们用极坐标的形式来表达  $Y_l[k]$ ， $S_l[k]$ ， $D_l[k]$ ：

$$\begin{aligned} Y_l[k] &= R_l[k]e^{j\phi} \\ S_l[k] &= A_l[k]e^{j\psi} \\ D_l[k] &= N_l[k]e^{j\theta} \end{aligned} \quad (2-3)$$

其中  $R$ ， $A$  和  $N$  分别是带噪语音，纯净语音和噪声的幅度谱， $\phi$ ， $\psi$  和  $\theta$  分别是带噪语音，纯净语音和噪声的相位谱。

先验信噪比定义为：

$$\xi_l[k] = \frac{\lambda_s[l, k]}{\lambda_d[l, k]} \quad (2-4)$$

后验信噪比定义为：

$$\gamma_l[k] = \frac{R_l^2[k]}{\lambda_d[l, k]} \quad (2-5)$$

为了用 MMSE 来估计  $N_l^2[k]$ ，定义条件期望： $E\{N^2|Y, \lambda_s, \lambda_d\}$ 。这里为了方便起见，在公式的推导中暂时先省略  $l$  和  $k$ 。我们用  $\hat{\cdot}$  符号表示一个数据的估计

值，以便与真实值区分，例如， $\widehat{N^2}$  是对  $N^2$  的估计。

利用贝叶斯定理（Bayes' Rule）， $\widehat{N^2}$  可以写为：

$$\begin{aligned}\widehat{N^2} &= E\{N^2|Y, \lambda_s, \lambda_d\} \\ &= \frac{\int_0^{+\infty} \int_0^{2\pi} n^2 f(Y|n, \theta, \lambda_s) f(n, \theta|\lambda_d) d\theta dn}{\int_0^{+\infty} \int_0^{2\pi} f(Y|n, \theta, \lambda_s) f(n, \theta|\lambda_d) d\theta dn}\end{aligned}\quad (2-6)$$

基于纯净语音和噪声的 STFT 系数服从复高斯分布的基本假设， $f(Y|n, \theta, \lambda_s)$  和  $f(n, \theta|\lambda_d)$  可以被改写为：

$$f(Y|n, \theta, \lambda_s) = \frac{1}{\pi\lambda_s} \exp\left\{-\frac{|Y - ne^{j\theta}|^2}{\lambda_s}\right\} \quad (2-7)$$

和

$$f(n, \theta|\lambda_d) = \frac{n}{\pi\lambda_d} \exp\left\{-\frac{n^2}{\lambda_d}\right\} \quad (2-8)$$

将式（2-7）和式（2-8）代入式（2-6），利用第一类零阶修正贝塞尔函数  $I_0(\cdot)$  的积分关系，可以得到：

$$\begin{aligned}\widehat{N^2} &= E\{N^2|Y, \lambda_s, \lambda_d\} \\ &= \frac{\int_0^{+\infty} n^3 \exp(-\frac{n^2}{\lambda}) I_0(\frac{2nY}{\lambda_d}) dn}{\int_0^{+\infty} n \exp(-\frac{n^2}{\lambda}) I_0(\frac{2nY}{\lambda_d}) dn}\end{aligned}\quad (2-9)$$

其中， $\lambda$  满足关系  $\lambda = \lambda_s \lambda_d / (\lambda_s + \lambda_d)$ 。

式（2-9）的解为：

$$\widehat{N^2} = \left[ \frac{1}{(1 + \xi)^2} + \frac{\xi}{(1 + \xi)\gamma} \right] R^2 \quad (2-10)$$

由式（2-10）可知，基于 MMSE 的噪声估计依赖于先验信噪比  $\xi$  和后验信噪比  $\gamma$ ，并且准确度最主要受到先验信噪比估计  $\hat{\xi}$  的影响。

## 2.2 利用 Deep Xi 估计先验信噪比

Deep Xi 对先验信噪比的估计流程包含两个阶段。

在第一个阶段中，输入带噪语音信号的幅度谱， $\mathbf{R}_l = \{R[l, 0], R[l, 1], \dots, R[l, K-1]\}$ ，经由 Deep Xi 系统内的深度神经网络，输出先验信噪比映射版本（mapped version）的估计值， $\hat{\xi}_l = \{\hat{\xi}_l[0], \hat{\xi}_l[1], \dots, \hat{\xi}_l[K-1]\}$ ，其中  $K$  是每帧内的离散频率索引的总数。

在第二个阶段中，我们从先验信噪比映射版本的估计值， $\hat{\xi}_l$ ，通过换算公式计算出先验信噪比的估计值， $\hat{\xi}_l$ 。

下面两小节描述了第一阶段先验信噪比映射版本的估计值  $\hat{\xi}_l$ ，和第二阶段先验信噪比估计值  $\hat{\xi}_l$  分别如何计算。

### 2.2.1 第一阶段，输出先验信噪比映射版本的估计值 $\hat{\xi}_l$

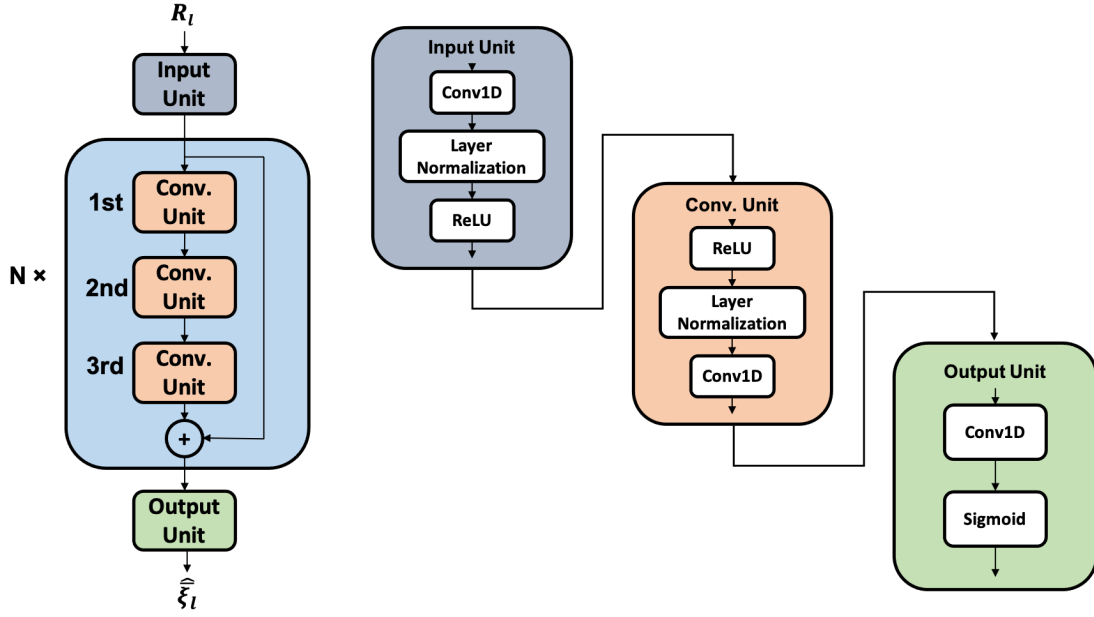


图 2-1: Deep Xi 中的神经网络框架

如图 2-1 所示，Deep Xi 内 DNN 的训练目标是瞬时先验信噪比的映射版本  $\hat{\xi}_l[k]$ <sup>[15]</sup>。对于瞬时情况，式 (2-1) 中纯净语音  $s[n]$  和噪声  $d[n]$  完全已知。这意味着可以将式 (2-4) 中的  $\lambda_s[l, k]$  和  $\lambda_d[l, k]$  分别替换为纯净语音和噪声幅度谱的平方，即  $A_l^2[k]$  和  $N_l^2[k]$ ，来得到瞬时先验信噪比  $\xi_l[k]$ ：

$$\xi_l[k] = \frac{A_l^2[k]}{N_l^2[k]} \quad (2-11)$$

为方便起见，我们将先验信噪比  $\xi_l[k]$  的单位换算成分贝，定义为  $\xi_{dB}[l, k]$ ：

$$\xi_{dB}[l, k] = 10 \log_{10}(\xi_l[k]) \quad (2-12)$$

为提高随机梯度下降算法 (Stochastic Gradient Descent algorithm, i.e. SGD) 的收敛速度，我们利用瞬时先验信噪比  $\xi_{dB}[l, k]$  的累积分布函数 (Cumulative Distribution Function, i.e. CDF)，将  $\xi_{dB}[l, k]$  映射到区间 [0,1]。这时  $\xi_{dB}[l, k]$  服从均值为  $\mu_k$ ，方差为  $\sigma_k^2$  的正态分布： $\xi_{dB}[l, k] \sim \mathcal{N}(\mu_k, \sigma_k^2)$ 。

据此，先验信噪比的映射版本  $\tilde{\xi}_l[k]$  由下式给出：

$$\tilde{\xi}_l[k] = \frac{1}{2} \left[ 1 + \operatorname{erf} \left( \frac{\xi_{dB}[l, k] - \mu_k}{\sigma_k \sqrt{2}} \right) \right] \quad (2-13)$$

其中， $\xi_{dB}[l, k]$  的统计量  $\mu_k$  和  $\sigma_k^2$  是从 Deep Xi 内 DNN 的训练集中计算出来的。

Deep Xi 输出先验信噪比映射的估计值，也即  $\hat{\xi}_l[k]$ 。

### 2.2.2 第二阶段，计算先验信噪比估计 $\hat{\xi}_l$

利用第一阶段 Deep Xi 输出的先验信噪比映射版本的估计值  $\hat{\xi}_l[k]$ ，我们不难从上一部分的计算过程倒推来得出先验信噪比估计  $\hat{\xi}_l[k]$ ：

$$\hat{\xi}_l[k] = 10^{((\sigma_k \sqrt{2} \operatorname{erf}^{-1}(2\hat{\xi}_l[k]-1) + \mu_k)/10)} \quad (2-14)$$

至此，我们使用 Deep Xi 系统得到了先验信噪比估计  $\hat{\xi}_l = \{\hat{\xi}_l[0], \hat{\xi}_l[1], \dots, \hat{\xi}_l[K-1]\}$ 。



### 第三章 Quality-Net 框架

Qulaity-Net 是一种基于 BLSTM 的端到端无参考源的语音质量评价模型<sup>[19]</sup>。

#### 3.1 Quality-Net 的基本结构

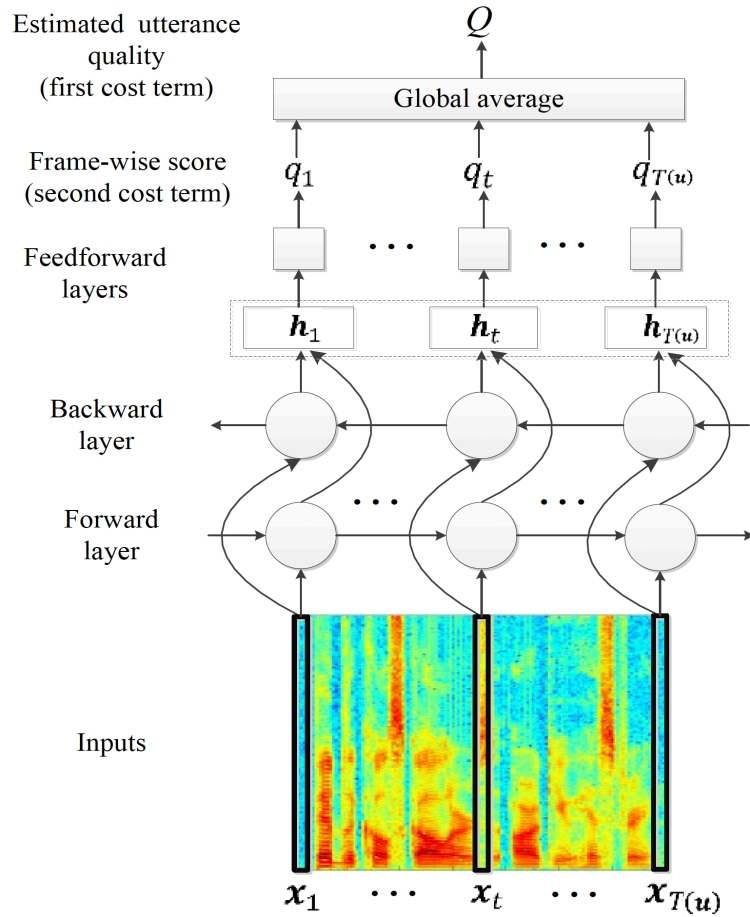


图 3-1: Quality-Net 基本结构<sup>[19]</sup>

如图 3-1，Qulaity-Net 基于 BLSTM 模型对 PESQ 值进行预测。网络的输入为带噪语音幅度谱，以每一帧作为一个单位（unit）输入 BLSTM 进行计算，生成一个帧级（frame-level）的 PESQ 估计值  $q_i$ ，最后通过全局平均得到整段语音（utterance-level）的 PESQ 估计值  $\hat{Q}$ 。

## 3.2 Quality-Net 的实现细节

### 3.2.1 基本参数

设  $\mathbf{u}$  为输入的带噪语音,  $\hat{Q}$  为估计的 PESQ 值,  $T(\mathbf{u})$  为输入语音  $\mathbf{u}$  的长度, 即帧数。由于输入带噪语音的长度不同, 该模型需要将  $\mathbf{u} \in R^{T(\mathbf{u})}$  映射到  $\hat{Q} \in R^1$ 。为了克服这种可变长度输入, 固定长度输出的映射限制, 我们利用 BLSTM 来模拟人类的听力感知系统并进行客观语音质量 PESQ 的估计。

### 3.2.2 目标函数

对于这个 BLSTM 神经网络, 我们的监督数据只有整段语音的 PESQ。当噪声不是平稳的, 逐帧的信噪比是变化时, 或者跨帧的语音失真程度不相同, 不适合直接将整段语音的客观质量评估值分配给输入信号的每一帧。然而, 当输入的语音质量较高时, 整段语音和每一帧客观质量评分的不一致性就不显著了。例如, 当我们输入纯净语音信号时, 输入幅度谱的每一帧都可以分配一个较高的 PESQ 估计值。基于这个观察, 我们在 Quality-Net 的目标函数 (objective function) 中加入了一个帧级质量评估 (frame quality assessment) 的条件约束 (conditional constraint): 当输入信号的语音质量较低时, 帧级约束将被赋予较低的权重。反之, 帧级约束将被赋予较高的权重。

据此, 我们推导出 Quality-Net 的目标函数  $O$  为:

$$O = \frac{1}{S} [(\hat{Q}_s - Q_s)^2 + \alpha(\hat{Q}_s) \sum_{t=1}^{T(\mathbf{u}_s)} (\hat{Q}_s - q_{s,t})^2] \quad (3-1)$$

其中,  $S$  为训练集中带噪语音的总数,  $\hat{Q}_s$  和  $Q_s$  分别为第  $s$  个语音的 PESQ 估计值和 PESQ 真实值,  $q_{s,t}$  是第  $s$  个语音第  $t$  帧的 PESQ 估计值,  $\alpha(\hat{Q}_s)$  为条件约束, 基于:

$$\alpha(\hat{Q}_s) = 10^{\hat{Q}_s - \hat{Q}_{max}} \quad (3-2)$$

其中,  $\hat{Q}_{max}$  是该客观语音质量度量的最大值, 对于 PESQ 而言,  $\hat{Q}_{max} = 4.5$ 。

我们注意到, 式 (3-1) 中的第一项只关注整段语音 PESQ 估计的准确性, 而不关注帧级的客观质量估计, 而第二项则使得帧级估计值遵循均匀分布。从式 (3-2) 可以看出, 对 PESQ 越高的语音, 条件约束  $\alpha(\hat{Q}_s)$  越显著。所以对于一个高质量的语音, 整段语音的 PESQ 估计值  $\hat{Q}$  由遵循均匀分布的帧级客观质量估计组成, 其帧级 PESQ 估计值与整段语音的 PESQ 值相等。

### 3.2.3 遗忘门偏置

BLSTM 的主要特点之一是在估计当前帧的 PESQ 值时会结合历史与将来的信息。然而, BLSTM 的这种特性也会使得 Quality-Net 中的帧级质量评估不能专注于当前帧的状况。例如, 在整条语音中如果只有一些区域的语音被噪声污

染，那么根据 BLSTM 的前后信息关联，纯净语音所在帧的 PESQ 值也可能会变得较低。因此，帧级质量的比较只能在一条语音的内部进行，而不能在不同语音之间进行比较。

为了解决这个问题，我们引入遗忘门（forget gate）来控制 BLSTM 中的前后信息流。Jozefowicz 等人提出，为了学习更多的前后信息依赖关系，遗忘门偏置（forget gate bias）应该初始化为一个较大的值<sup>[20]</sup>。因为我们打算降低 BLSTM 中使用前后信息关联的程度，所以遗忘门偏置被初始化为一个较小的值，使得 Quality-Net 容易忘记历史信息并更多地关注当前帧的状况。



## 第四章 Xi-Q 算法

为了最大限度地利用有限的数据集，本文基于强化学习中的 Q-Learning，设计与 Deep Xi 系统相匹配的自优化算法，称之为 Xi-Q Algorithm，目标是实现该语音增强系统的自优化。这一章将首先介绍 Q-Learning 的基本框架，然后阐述如何依据语音增强系统的特征来使得 Q-Learning 中的概念适配 Deep Xi，最后详细说明提出的 Xi-Q 算法的内容。

### 4.1 Q-Learning 的基本框架

传统的强化学习可以理解为马尔可夫决策过程（markov decision process, i.e. MDP），目的是让主体学习一个最优的或接近最优的策略，使得主体能够在特定的环境和状态下通过采取行动，获得最大的累积奖励。

Q-Learning 有以下四个要素：

- 环境和主体的状态（states）集合： $\mathcal{S}$ ；
- 主体的动作（actions）集合： $\mathcal{A}$ ；
- 状态转移矩阵，也被称之为状态转移概率（probability）函数，表示主体从状态  $s$  执行动作  $a$  转移到状态  $s'$  的概率： $\mathcal{P}_a(s, s') = \mathcal{P}(s_{t+1} = s' | s_t = s, a_t = a)$ ；
- 奖励（reward），表示从状态  $s$  执行动作  $a$  转移到状态  $s'$  所获得的反馈： $\mathcal{R}_a(s, s')$ 。

Q-Learning 是一种无模型（model-free）的强化学习算法，用于让主体学习特定状态下动作的价值。Q-Learning 在  $\mathcal{S}$  中仅需要主体的状态集合而不需要环境模型，因此被称为 model-free。

Q-Learning 与上述传统强化学习四要素中不同的是它的状态转移矩阵被重新定义为状态-动作价值函数（state-action value function），这个函数计算每个状态-动作组合（combination）的价值（quality） $Q(s, a)$ ：

$$Q : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R} \quad (4-1)$$

这个状态-动作价值函数也被称为  $Q$  函数。

在标准的 Q-Learning 算法中，学习开始之前， $Q$  函数首先被初始化。接下来，在任一时刻  $t$ ，处于状态  $s_t$  的主体选择一个动作  $a_t$ ，得到一个奖励  $r_t$  并进入下一个状态  $s_{t+1}$ ， $Q$  函数也随之被更新。Q-Learning 算法的核心是基于 Bellman 方程的动态规划，使用旧值（old value）和新信息（new information）的加权平均值来迭代更新  $Q$  函数：

$$Q^{new}(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \cdot (r_t + \gamma \cdot \max_{a \in \mathcal{A}} Q(s_{t+1}) - Q(s_t, a_t)) \quad (4-2)$$

其中  $\alpha$  被称为学习速率 (learning rate,  $0 < \alpha \leq 1$ ) ,  $\gamma$  被称为折扣因子 (discount factor) 。

学习速率  $\alpha$  决定了新获取的信息在多大程度上覆盖旧信息。 $\alpha = 0$  使主体什么也学不到, 只利用先验知识, 而  $\alpha = 1$  使主体忽略先验知识以探索可能的动作。折扣因子  $\gamma$  决定了未来奖励的重要性。如果  $\gamma = 0$ , 那么主体只考虑当前的奖励, 如果  $\gamma \rightarrow 1$  则会让主体追求未来的高奖励。

综合以上信息, 我们可以看出 Q-Learning 的基本框架重点在于四个要素的定义:  $\mathcal{S}$ ,  $\mathcal{A}$ ,  $Q$ ,  $\mathcal{R}$ 。

## 4.2 适配 Deep Xi 的 Q-Learning 模型

对于 Deep Xi 系统, 我们有如下观察:

- 输入为带噪语音信号的幅度值  $R_l[k]$ ;
- 输出为先验信噪比的估计值  $\hat{\xi}_l[k]$ ;
- Deep Xi 对一帧信号估计的先验信噪比  $\hat{\xi}_l[k]$ , 不会影响其对后续信号的估计值;
- Deep Xi 系统的语音增强效果可以通过一些客观语音质量评价标准, 如 PESQ 来评判。

基于上述特征, 我们可以对 Q-Learning 模型进行适配:

- 对于状态  $\mathcal{S}$ :  
定义为输入带噪语音信号的幅度值  $R_l[k]$ ;
- 对于动作  $\mathcal{A}$ :  
定义为选择合适的先验信噪比估计值  $\hat{\xi}_l[k]$ , 我们需要预先定义数量  $A$  的先验信噪比模版 (Template)  $T$ ;
- 对于  $Q$  函数:  
由于对于特定的状态  $s_t$ , 选择动作  $a_t$  之后,  $s_{t+1}$  会变为噪声的 PSD 估计, 对于  $s_t$  没有参考意义, 所以设置折扣因子  $\gamma = 0$ 。由于 Deep Xi 所处于一个完全确定的环境 (fully deterministic environments), 所以设置学习速率  $\alpha = 1$ , 以达到最好的学习效果。  
所以式 (4-2) 可以简化为

$$Q^{new}(s_t, a_t) \leftarrow r_t \quad (4-3)$$

因此, 此时的  $Q$  函数就是奖励函数  $\mathcal{R}$ ;

- 对于奖励函数  $\mathcal{R}$ :  
由于输入只有带噪语音的幅度值, 我们无法利用纯净语音信号作为参考计算出 PESQ 的真实值, 但 Quality-Net 给予了我们利用客观音质评价来作为

奖励函数的可能。直接使用增强信号经由 Quality-Net 输出的 PESQ 估计值作为奖励函数  $\mathcal{R}$  是一个非常直观的方式，但实际上会造成问题。PESQ 估计值的高低不仅取决于语音增强的效果，还由带噪语音信噪比的高低决定。

为避免受到这些外部因素的影响，我们通过计算利用  $Q$  函数选取的先验信噪比估计值  $\hat{\xi}_l[k]$  经由 Quality-Net 所得出的 PESQ 估计值，与 Deep Xi 中 DNN 选取的先验信噪比估计值  $\hat{\xi}_l[k]$  经由 Quality-Net 所得出的 PESQ 估计值作差，得到的差值作为该算法奖励函数的一个自变量  $P$ 。

此外，PESQ 的计算需要多帧信号，不能对每一帧信号进行分析，因此不能反映每一次动作选取的好坏。因此，对于  $Q$  函数在一帧上错误的动作选择，我们还需要给它一个惩罚系数  $E$ 。

- 对于模型的数据结构：

我们观察到  $Q$  函数其实是一个判别函数（discriminative function），用来对给定状态  $s$  判别动作  $a$  的最优选择。我们用矩阵所具有的线性映射功能来表示  $Q$  这一从状态到动作的判别函数。

### 4.3 Xi-Q 算法

#### 4.3.1 定义参数

Parameter	Caption
$s[n], A[l, k]$	纯净语音信号与其 STFT 系数幅值
$d[n], N[l, k]$	噪声信号与其 STFT 系数幅值
$y[n], R[l, k]$	带噪语音信号与其 STFT 系数幅值
$T[A]$	先验信噪比模版
$Q(\mathbf{R}[l], a)$	$Q$ 函数
$\xi[l, k]$	先验信噪比实际值
$\hat{\xi}_{xiq}[l, k]$	Xi-Q 算法先验信噪比估计值的映射版本
$\hat{\xi}_{xiq}[l, k]$	Xi-Q 算法先验信噪比估计值
$\hat{\xi}_{max}[l, k]$	$Q$ 函数按照贪婪策略选择的先验信噪比估计值的映射版本
$\hat{\xi}_{max}[l, k]$	$Q$ 函数按照贪婪策略选择的先验信噪比估计值
$batch\ size$	自优化阶段每一批次的数据量
$a_l$	在 $l$ 帧，Xi-Q 从 $T_{1,2,\dots,A}$ 中选出的一个模版 $T_{a_l}$
$a_l^{max}$	在 $l$ 帧，目前 $Q(\mathbf{R}[l], a)$ 所做的最佳决定
$\hat{Z}^{xiq}$	Xi-Q 算法语音增强结果经由 Quality-Net 所得出的 PESQ 估计值
$\hat{Z}^{max}$	当前按照 $Q$ 函数所做最佳决定经由 Quality-Net 所得出的 PESQ 估计值
$P$	奖励函数的一个自变量，与 $\hat{Z}^{xiq}$ 和 $\hat{Z}^{max}$ 有关
$E[l]$	帧级错误动作惩罚系数
$r[l]$	奖励函数

表 4-1: Xi-Q 算法参数定义

### 4.3.2 具体内容

Xi-Q 算法主要分为初始化阶段和自优化阶段。

#### 4.3.2.1 初始化阶段

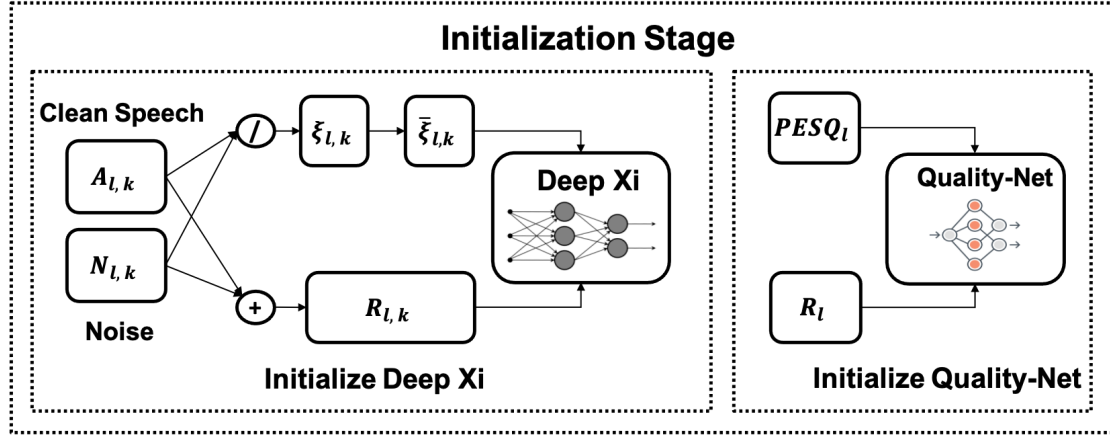


图 4-1: Xi-Q Algorithm 初始化阶段

如图 4-1，这一阶段有如下几个步骤：

1. 将容量为  $N$  的训练集中的数据  $s[n]$ ,  $d[n]$  作 STFT，得到  $A[l, k]$ ,  $N[l, k]$ ，利用式 (2-11) 得到先验信噪比实际值  $\xi[l, k]$  与其对应的映射版本  $\bar{\xi}[l, k]$ 。
2. 以  $R[l, k]$  作为输入数据，以先验信噪比实际值的映射版本  $\bar{\xi}[l, k]$  作为监督数据，利用反向传播算法 (Back-Propagation Algorithm, i.e. BP) 训练 Deep Xi 中的 DNN。
3. 利用已知的纯净语音信号  $s[l]$  作为参考源，计算带噪语音  $y[l]$  的 PESQ 真实值  $Z[l]$ 。以  $R[l, k]$  作为输入数据，以带噪语音的 PESQ 真实值  $Z[l]$  作为监督数据，利用 BP 算法训练 Quality-Net。

---

#### 算法 4.1 Xi-Q Algorithm: Initialization Stage

---

**Input:**  $s_r[n]$ ,  $d_r[n]$ ,  $s_q[m]$ ,  $y_q[m]$ , resnet-1.1n, Quality-Net

- 1: **repeat**
- 2:    $n \leftarrow n + 1$
- 3:   Calculate  $A[l, k]$ ,  $N[l, k]$  based on Equation (2-2) - (2-3)
- 4:   Calculate  $\bar{\xi}[l, k]$  based on Equation (2-11) - (2-13)
- 5:   Train resnet-1.1n based on BP
- 6: **until** Accuracy satisfied
- 7: **repeat**
- 8:    $m \leftarrow m + 1$
- 9:   Calculate PESQ based on  $s_q[m]$  and  $y_q[m]$
- 10:   Train Quality-Net based on BP
- 11: **until** Accuracy satisfied

**Output:** trained model of resnet-1.1n and Quality-Net

---



### 4.3.2.2 自优化阶段

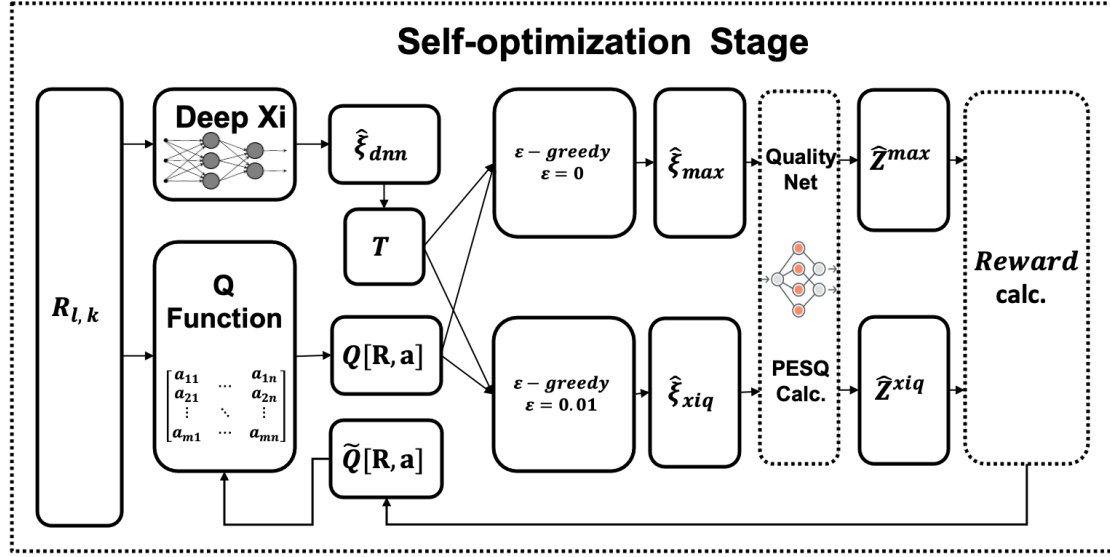


图 4-2: Xi-Q Algorithm 自优化阶段

如图 4-2，这一阶段有如下几个步骤：

1. 首先，将 *batch size* 个带噪语音幅度谱  $R[l, k]$  输入 Deep Xi 系统，输出先验信噪比的估计值  $\hat{\xi}_{dnn}[l, k]$ 。以  $\hat{\xi}_{dnn}[l, k]$  生成先验信噪比的模版  $T$ 。
2. 向  $Q(R[l], a)$  函数中输入带噪语音幅度谱  $R[l, k]$ ，利用  $\epsilon - greedy$  策略，首先设定一个大于零的  $\epsilon$  的数值（常取 0.01），并生成一个随机数  $rnd$ ，，确定动作  $a_l$ ，并选择相对应的先验信噪比模版  $\hat{\xi}_{xiq}[l, k]$ ：

$$a_l \leftarrow \begin{cases} \arg \max_{a \in T} Q(R[l], a) & (rnd > \epsilon) \\ \forall a \in T & (rnd \leq \epsilon) \end{cases} \quad (4-4)$$

$$\hat{\xi}_{xiq}[l, k] \leftarrow T_{a_l} \quad (4-5)$$

3. 令第 2 步中的  $\epsilon = 0$ ，即采用贪婪策略，选出当前  $Q(R[l], a)$  下预计奖励最多的动作  $a_l^{max}$ ，并确定相对应的先验信噪比模版  $\hat{\xi}_{max}[l, k]$ ：

$$a_l^{max} \leftarrow \arg \max_{a \in T} Q(R[l], a) \quad (4-6)$$

$$\hat{\xi}_{max}[l, k] \leftarrow T_{a_l^{max}} \quad (4-7)$$

4. 已有  $\hat{\xi}_{xiq}[l, k]$  和  $\hat{\xi}_{max}[l, k]$ ，结合  $\hat{\gamma} = \hat{\xi} + 1$ ，带入式 (2-10)，分别得出  $\widehat{N_{xiq}^2}$  和  $\widehat{N_{max}^2}$ 。
5. 从加性噪声的假定出发，利用  $\widehat{N_{xiq}^2}$  和  $\widehat{N_{max}^2}$ ，得到  $\widehat{A_{xiq}^2}$  和  $\widehat{A_{max}^2}$ ，乘上对应的相位谱，并运用 ISTFT 恢复出增强信号  $s_{xiq}$  和  $s_{max}$ 。

6. 将基于 Xi-Q 算法输出的增强信号  $s_{xiq}$  输入 Quality-Net, 计算出 Xi-Q 算法经过  $\epsilon - greedy$  策略在这次的选择中所得出的 PESQ 估计值  $\hat{Z}^{xiq}$ 。
7. 将基于 Xi-Q 算法输出的增强信号  $s_{max}$  输入 Quality-Net, 计算出目前为止  $Q(\mathbf{R}[\mathbf{I}], a)$  所决定的最大的 PESQ 估计值  $\hat{Z}^{max}$ 。
8. 将  $\hat{Z}^{xiq}$  和  $\hat{Z}^{max}$  带入  $P$  函数:

$$P = \tanh\{\alpha(\hat{Z}^{xiq} - \hat{Z}^{max})\} \quad (4-8)$$

其中  $\alpha > 0$  是一个调整系数。

这个相对值  $P$  的设定思路来源于计算机游戏的胜利或失败。如果  $\hat{Z}^{xiq}$  高于  $\hat{Z}^{max}$ , 即 Xi-Q 算法这次选择的表现比上次迭代结果生成  $Q(\mathbf{R}[\mathbf{I}], a)$  所做的贪婪决定的要好, 则给予 Xi-Q 算法这次的选择一个正值奖励。反之, 如果  $\hat{Z}^{xiq}$  低于  $\hat{Z}^{max}$ , 则说明 Xi-Q 算法这次所做的尝试没有好于先前的最佳决定。对于第二种情况, 我们可以给这次 Xi-Q 所做的选择一个负值奖励, 即迭代时 Xi-Q 算法动作的  $Q$  值加上  $P$ ; 也可以给上次迭代的最好结果一个正值奖励, 即迭代时目前最佳动作的  $Q$  值减去  $P$ 。函数  $\tanh$  限制了  $P$  的值域, 也防止了梯度爆炸 (gradient exploding) 情况的发生。

9. 为了惩罚帧级错误动作, 正如前文提过, 我们提出了一个系数  $E[\mathbf{I}]$ , 在这里我们如是定义:

$$\hat{E}[\mathbf{I}] = \sum_{k=0}^{K-1} |\ln(\widehat{A_{xiq}^2}) - \ln(\widehat{A_{max}^2})|^2 \quad (4-9)$$

$$E[\mathbf{I}] = \frac{\hat{E}[\mathbf{I}]}{\max_{l \in L}(\hat{E}[\mathbf{I}])} \quad (4-10)$$

可见  $0 < E[\mathbf{I}] < 1$ , 是一个正则化的误差系数。当算法在某一帧做出局部错误决定时, 与平方误差成正相关的  $E[\mathbf{I}]$  就会变得很大, 所以  $E[\mathbf{I}]$  能作为一个很好的惩罚系数。

10. 有了  $P$  和  $E[\mathbf{I}]$ , 我们定义最终的奖励函数  $r[\mathbf{I}]$ :

$$r[\mathbf{I}] = \begin{cases} (1 - E[\mathbf{I}])P & (P > 0) \\ E[\mathbf{I}]P & (other) \end{cases} \quad (4-11)$$

11. 根据式 (4-3), 有了  $r[\mathbf{I}]$ , 我们便能更新  $Q(\mathbf{R}[\mathbf{I}], a)$  函数。  
首先更新  $Q(\mathbf{R}[\mathbf{I}], a_l)$  的值:

$$\tilde{Q}(\mathbf{R}[\mathbf{I}], a_l) \leftarrow \begin{cases} r[\mathbf{I}] + \max_{a \in \mathcal{T}} Q(\mathbf{R}[\mathbf{I}], a) & (P > 0) \\ Q(\mathbf{R}[\mathbf{I}], a_l) & (other) \end{cases} \quad (4-12)$$

接着，如果  $a_l \neq a_l^{max}$ ，我们继续更新  $Q(\mathbf{R}[\mathbf{I}], a_l^{max})$  的值：

$$\tilde{Q}(\mathbf{R}[\mathbf{I}], a_l^{max}) \leftarrow \begin{cases} Q(\mathbf{R}[\mathbf{I}], a_l^{max}) & (P > 0) \\ Q(\mathbf{R}[\mathbf{I}], a_l^{max}) - r[l] & (other) \end{cases} \quad (4-13)$$

式（4-12）设计的目的是，当 Xi-Q 算法做出的动作  $\mathbf{a}$  比之前做出的最佳动作  $\mathbf{a}^{max}$  整体要好时，我们给予  $\mathbf{a}$  所包含的所有动作一个正值奖励  $r[l]$ ，使其成为所在帧上具有最大  $Q$  值的动作。

式（4-13）设计的目的是，这次 Xi-Q 算法做出的动作  $\mathbf{a}$  整体不如之前做出的最佳动作  $\mathbf{a}^{max}$  好，这时有  $r[l] < 0$ 。按照上述步骤 9 所提到的第二种惩罚方法，我们让  $Q(\mathbf{R}[\mathbf{I}], a_l^{max})$  的值减去  $r[l]$ ，相当于给予目前最佳动作  $\mathbf{a}^{max}$  一个正值奖励。

12. 用更新的  $\tilde{Q}(\mathbf{R}[\mathbf{I}], a)$  作为新的动作-价值函数  $Q$ 。

13. 重复步骤 1 ~ 12  $M$  次，记为  $M$  epochs。

---

#### 算法 4.2 Xi-Q Algorithm: Self-optimization Stage

---

**Input:**  $R_i[l, k]$  ( $i = 0, 1, \dots, \text{batch\_size}$ )

- 1: Generate template  $T$  based on  $\mathbf{R}[l, k]$
- 2: **repeat**
- 3:    $cnt \leftarrow cnt + 1$
- 4:   Select  $\hat{\xi}_{xiq}[l, k]$  based on Equation (4-4) and (4-5)
- 5:   Select  $\hat{\xi}_{max}[l, k]$  based on Equation (4-6) and (4-7)
- 6:   Update  $\widehat{N}_{xiq}^2$  and  $\widehat{N}_{max}^2$  based on Equation (2-10)
- 7:   Restore signal  $s_{xiq}$  and  $s_{max}$  based on ISTFT
- 8:   Update  $\hat{Z}^{xiq}$  and  $\hat{Z}^{max}$  based on Quality-Net
- 9:   Update  $P$  based on Equation (4-8)
- 10:   Update  $E[l]$  based on Equation (4-9) and (4-10)
- 11:   Update reward  $r[l]$  based on Equation (4-11)
- 12:   Update  $\tilde{Q}(\mathbf{R}[\mathbf{I}], a_l)$  based on Equation (4-12) and (4-13)
- 13: **until** Converge or  $cnt = M$
- 14: Select  $\hat{\xi}_{xiq}[l, k]$  based on Equation (4-4) and (4-5)
- 15: Restore signal  $s_{xiq}$  based on  $\hat{\xi}_{xiq}[l, k]$

**Output:**  $s_{xiq}$

---



## 第五章 实验

### 5.1 实现细节

代码运行于 Python 3.7.6 和 TensorFlow 2.4.1 环境下。

#### 5.1.1 信号处理

在 STFT 过程中，采用平方根-hann 窗函数（square-root-Hann window function），帧长为 32 ms（512 samples），帧移为 16ms（256 samples）。

#### 5.1.2 训练集

纯净语音训练集：来自 Librispeech 语料<sup>[21]</sup> 共 28539 条数据，来自 CSTR VCTK 语料<sup>[22]</sup> 共 42 015 条数据，来自 TIMIT 语料<sup>[23]</sup> 共 3 696 条数据。总共有 74 250 个纯净的语音信号。

噪声训练集：来自 QUT-NOISE 数据集<sup>[24]</sup>，Environmental Background Noise 数据集<sup>[25]</sup> 和 MUSAN 语料库<sup>[26]</sup> 的噪声集，总共有 2382 条噪音记录。

所有纯净语音和噪声都是单通道的，采样频率为 16khz。

#### 5.1.3 测试集

纯净语音测试集：来自 TSP 语料库<sup>[27]</sup> 随机抽取的 30 条纯净语音信号。

噪声测试集：包含 8 种不同噪声源的录音。一类是计算机产生的噪声源，包括调制的（modulated）高斯白噪声，它由调制高斯噪声得到：

$$f(n) = 1 + \sin\left(2\pi n \frac{f_{mod}}{f_s}\right) \quad (5-1)$$

其中  $f_{mod}$  和  $f_s$  分别表示调制频率和采样频率， $n$  为帧数。在实验中，我们将调制频率设置为  $f_{mod} = 0.5Hz$ 。

另一类包括 5 种真实世界的非平稳噪声源，其中有火车驶过的噪声，汽车驶过的噪声，交通拥挤处的噪声，来自 Urban Sound 数据集<sup>[28]</sup> 中的露天播放的音乐，和 RSG-10 噪声数据集<sup>[29]</sup> 中的 voice babble。

最后两种是真实世界的有色噪声源，包括来自 RSG-10 噪声数据集<sup>[29]</sup> 的工厂噪声和 F16 飞机发动机的噪声。

所有纯净语音和噪声都是单通道的，采样频率为 16khz。

#### 5.1.4 训练 Deep Xi 中的 resnet-1.1n

- 生成带噪语音：

我们按照  $batch\_size = 32$  分批次生成带噪语音。在每一批中，我们首先从纯净语音训练集中选取 32 条数据，从噪声训练集中选取 960 条数据。对每一条纯净语音，我们以  $-10dB \sim 20dB$ ， $1dB$  为步长的信噪比与噪声数据的随机部分混合。这样，有 890 批次，每一批有 960 条数据，总共 854400 条 Deep Xi 的训练数据。

- 配置 DNN 拟合参数：

Parameter	Value
batch_size	32
epoches	200
optimizer	Adam
loss_function	binary_crossentropy
metrics	accuracy

表 5-1: DNN 拟合参数

- 保存训练结果：

将 resnet-1.1n 模型保存在 h5 文件中。

#### 5.1.5 训练 Quality-Net

- 生成带噪语音：

我们按照  $batch\_size = 128$  分批次生成带噪语音。在每一批中，我们首先从纯净语音训练集中选取 128 条数据，从噪声训练集中选取 1024 条数据。对每一条纯净语音，我们以  $-20dB \sim 15dB$ ， $5dB$  为步长的信噪比与噪声数据的随机部分混合。这样，有 15 批次，每一批有 1024 条数据，总共 15360 条 Quality-Net 的训练数据。

- 配置 Quality-Net 拟合参数：

Parameter	Value
batch_size	128
epoches	20
optimizer	RMSprop
loss_function of Average_Score	mse
loss_function of Frame_Score:	frame_mse
metrics	loss

表 5-2: Quality-Net 拟合参数

- 保存训练结果：  
将 Quality-Net 模型保存在 h5 文件中。

### 5.1.6 执行 Xi-Q 算法

- Xi-Q 算法基本参数选取：

Parameter	Value	Caption
$\epsilon$	0.01	$\epsilon - greedy$ 策略中的阈值
$\alpha$	20.0	$P$ 函数的调整系数
$batch\_size$	40	每一批处理的带噪语音数量
$max\_eposide$	2000	Xi-Q 算法最大迭代次数
$eps$	0.03	每 1000 次迭代判断是否收敛的标准
$A$	$\sum_{i=0}^{batch\_size} length(R[i])$	每一批的模版数量

表 5-3: Xi-Q 参数

- 具体流程：  
首先，将 40 组带噪语音的幅度谱  $\mathbf{R}[i](i = 0, 1, \dots, 39)$  输入到 Deep Xi 系统，输出对应的先验信噪比估计值  $\hat{\xi}_i[l, k]$  和增强后的语音信号  $\tilde{s}[i]$ ，并以  $\hat{\xi}_i[l, k]$  作为 Xi-Q 的动作模版  $T[A]$ 。依据 resnet-1.1n 的训练结果初始化  $Q_i(\mathbf{R}[I], a)$  函数。  
接着，计算出  $\tilde{s}[i]$  的幅度谱  $\tilde{\mathbf{R}}[i]$ ，并依据  $Q$  函数从  $T[A]$  中按照  $\epsilon - greedy$  策略和贪婪策略分别选择先验信噪比模版  $\mathbf{a}_i^{xiq}$  和  $\mathbf{a}_i^{max}$ 。将按照动作选择增强后的语音输入 Quality-Net 进行客观语音质量评估，得出奖励  $r_i[m]$ 。  
最后，依据  $r_i[m]$  更新动作-价值函数为  $\tilde{Q}_i(\mathbf{R}[I], a)$ 。若收敛或者达到最大迭代次数，停止。

## 5.2 验证基于 Quality-Net 的 Xi-Q 算法结果

我们首先构造测试数据。共 3 批数据，每一批从纯净语音测试集中随机选取 8 条数据。对每一条纯净语音，从噪声测试集中随机选取 5 条数据，分别以  $[-15, -10, -5, 0, 5]dB$  作为信噪比混合。总共 120 条测试数据。

我们用对数谱距离 (log spectral distance, i.e. LSD) 作为语音增强效果的评判标准。两个频谱  $P(k)$  和  $\hat{P}(k)$  之间的对数谱距离被定义为：

$$D = \sqrt{\frac{1}{2\pi} \int_{-\pi}^{\pi} [10 * \log_{10} \frac{P(k)}{\hat{P}(k)}]^2 dk} \quad (5-2)$$

其中， $P(k)$  和  $\hat{P}(k)$  是功率谱。

在验证过程中，我们对测试数据中的每个带噪语音信号计算出四个结果：

1.  $D^{dnn}$ : 用 Deep Xi 的 resnet-1.1n 作为映射函数输出先验信噪比的估计值  $\hat{\xi}_{dnn}$ , 以此  $\hat{\xi}_{dnn}$  进行语音增强输出结果的 LSD 值;
2.  $D^{xiq}$ : 用 Xi-Q 算法, 以  $Q(\mathbf{R}[\mathbf{l}], a)$  作为映射函数来做决定, 得出先验信噪比的估计值  $\hat{\xi}_{xiq}$ , 以此  $\hat{\xi}_{xiq}$  进行语音增强输出结果的 LSD 值;
3.  $\delta_D = D^{dnn} - D^{xiq}$ : 用来表示 Xi-Q 算法对 LSD 的绝对降低量;
4.  $\Delta_D = \frac{D^{dnn} - D^{xiq}}{D^{dnn}}$ : 用来表示 Xi-Q 算法对 LSD 的相对降低量。

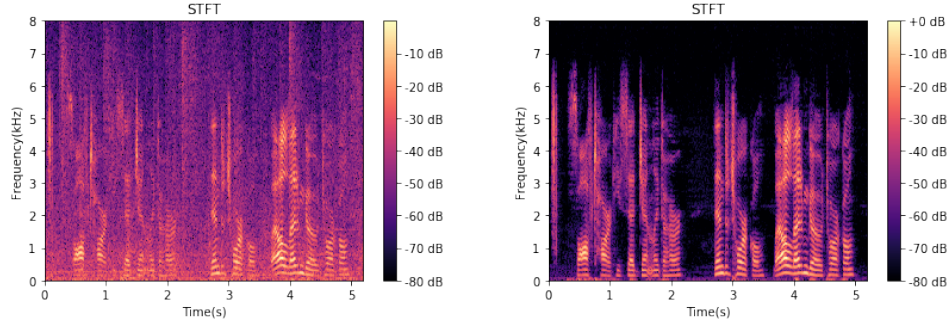


图 5-1: (a) 一个带噪语音

(b) 对应的纯净语音谱

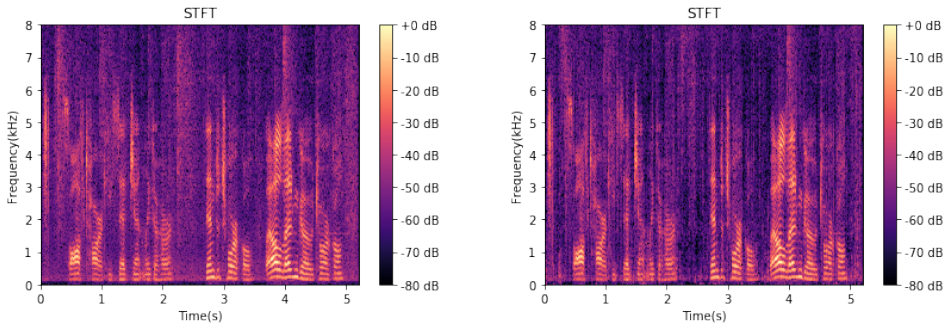


图 5-2: (a) Deep Xi 增强结果

(b) Xi-Q 增强结果

图 5-1 和 5-2 展示的是测试集的一个例子, 我们可以看出基于 Quality-Net 的 Xi-Q 算法对降低 LSD 的效果。在图 5-2 (b) 中  $0 \sim 0.5s$ ,  $2.1 \sim 2.8s$ ,  $3.4 \sim 3.7s$  以及  $4.9 \sim 5.1s$  内 Xi-Q 的处理结果明显优于图 5-2 (a) 中 Deep Xi 的处理结果。

SNR	$\delta_D$	$\Delta_D$
-15 dB	0.1413	5.85 %
-10 dB	0.0979	4.55%
-5 dB	0.1017	5.32%
0 dB	0.0109	0.92%
5 dB	-0.0245	-1.61%

 表 5-4: 验证结果: 以 SNR 分类, 输出的  $\delta_D$  与  $\Delta_D$



$\delta_D$	0.0655
$\Delta_D$	3.01%

 表 5-5: 验证结果:  $\delta_D$  与  $\Delta_D$  在整个测试集上的平均值

整体验证结果如表 5-4 和表 5-5。从表中我们可以观察到, 基于 Quality-Net 对 PESQ 值估计的情况下, Xi-Q 算法迭代 2000 次将 LSD 平均降低 0.0655, 相对优化 3.01%。

### 5.3 验证基于真实 PESQ 数值的 Xi-Q 算法结果

目前 Quality-Net 训练结果的测试集误差为 0.3664, 并且在 80% 的情况下估计的 PESQ 值与真实的 PESQ 值满足线性相关性。但这种准确度的 Quality-Net 对于 PESQ 的估计效果可以说仍不够理想。由于 PESQ 估计的准确度会对 Xi-Q 算法的自优化结果带来不小的误差, 因此为了排除 Quality-Net 目前 PESQ 值的预测效果对最终增强结果所带来的影响, 我们研究在 PESQ 值估计理想的情况下 Xi-Q 算法所能达到的理论效果。

#### 5.3.1 对 Deep Xi 系统整体性能的提升

测试数据与 5.2 一致。数据以  $[-5, 0, 5, 10, 15]dB$  作为信噪比混合。

我们用 PESQ 作为语音增强效果的评判标准。在验证过程中, 我们对测试集中的每个带噪语音信号计算出四个结果:

1.  $Z^{dnn}$ : 用 Deep Xi 的 resnet-1.1n 作为映射函数输出先验信噪比的估计值  $\hat{\xi}_{dnn}$ , 以此  $\hat{\xi}_{dnn}$  进行语音增强输出结果的 PESQ 值;
2.  $Z^{xiq}$ : 用 Xi-Q 算法, 以  $Q(\mathbf{R}[\mathbf{I}], a)$  作为映射函数来做决定, 得出先验信噪比的估计值  $\hat{\xi}_{xiq}$ , 以此  $\hat{\xi}_{xiq}$  进行语音增强输出结果的 PESQ 值;
3.  $\delta_Z = Z^{xiq} - Z^{dnn}$ : 用来表示 Xi-Q 算法对 PESQ 的绝对增加量;
4.  $\Delta_Z = \frac{Z^{xiq} - Z^{dnn}}{Z^{dnn}}$ : 用来表示 Xi-Q 算法对 PESQ 的相对增加量, 也即 Deep Xi 系统的优化量。

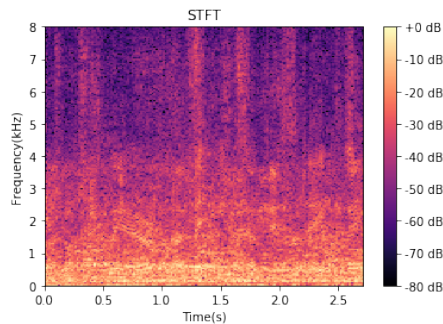
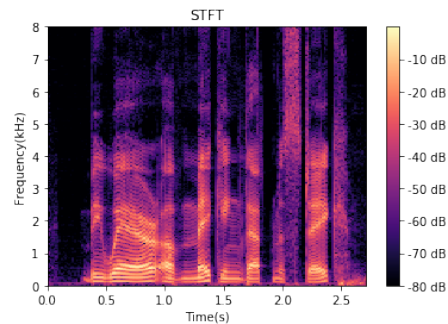


图 5-3: (a) 一个带噪语音



(b) 对应的纯净语音谱

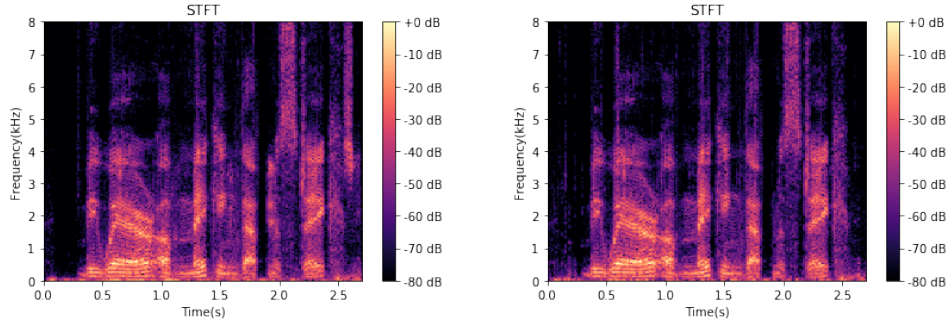


图 5-4: (a)Deep Xi 增强结果

(b)Xi-Q 增强结果

图 5-3和 5-4展示的是测试集的一个例子，我们可以看出理想情况下 Xi-Q 算法对于 Deep Xi 的优化。在图 5-4 (b) 中 1.8 ~ 2.0s 和 2.6 ~ 2.8s 内 Xi-Q 的处理效果明显优于图 5-4 (a) 中 Deep Xi 的处理结果。

SNR	$\delta_Z$	$\Delta_Z$
-5 dB	0.2623	14.36 %
0 dB	0.3095	14.73%
5 dB	0.3318	13.90%
10 dB	0.4737	19.71%
15 dB	0.4341	15.78%

 表 5-6: 验证结果：以 SNR 分类，输出的  $\delta_Z$  与  $\Delta_Z$ 

$\delta_Z$	0.3613
$\Delta_Z$	15.66%

 表 5-7: 验证结果：  $\delta_Z$  与  $\Delta_Z$  在整个测试集上的平均值

整体验证结果如表 5-6和表 5-7。从表中我们可以观察到，在 PESQ 值估计理想的情况下迭代 2000 次：

- Xi-Q 算法能将 Deep Xi 系统语音增强结果的 PESQ 平均提高 0.3613；
- Xi-Q 算法能将 Deep Xi 系统整体性能提升 15.66%。

### 5.3.2 对 Deep Xi 系统输出结果较差的特定带噪语音有很好的处理结果

在测试集中有这样一段带噪语音与它对应的纯净语音谱，如图 5-5所示：可以看出，给这段纯净语音加的是频谱成倍数的噪声分量。

图 5-6 (a) 为 Deep Xi 处理的结果，可以看出很不理想。对于这段带噪语音，Deep Xi 只能将在时域内频率一定的噪声去除，而对某一时刻这种频谱垂

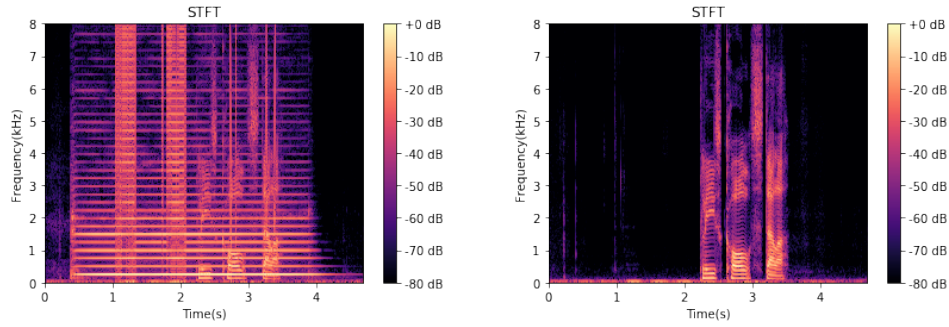


图 5-5: (a) 一个带噪语音

(b) 对应的纯净语音谱

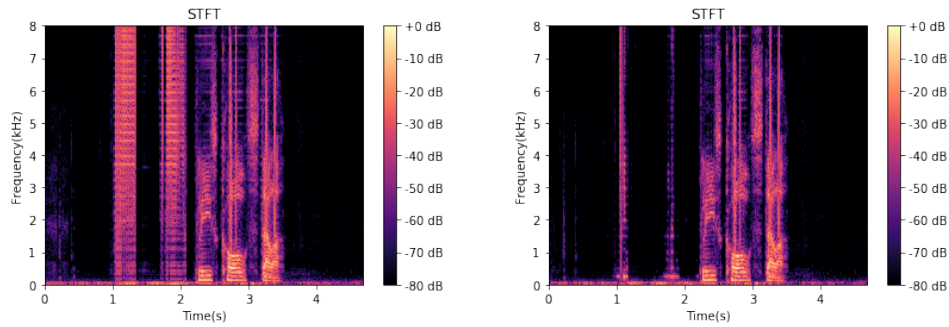


图 5-6: (a)Deep Xi 增强结果

(b)Xi-Q 增强结果

直分布的噪声可以说几乎没有处理。由于人的声纹的频谱也是成倍数的，与图中所加的噪声很类似，所以在语音增强的过程中，Deep Xi 系统误把这类噪声当成了语音，导致处理结果很差。

实验结果表明，在 PESQ 值估计理想的情况下，Xi-Q 能很好地改善这种情况。图 5-7 展示了 Xi-Q 算法大概在 15000 次迭代后收敛。

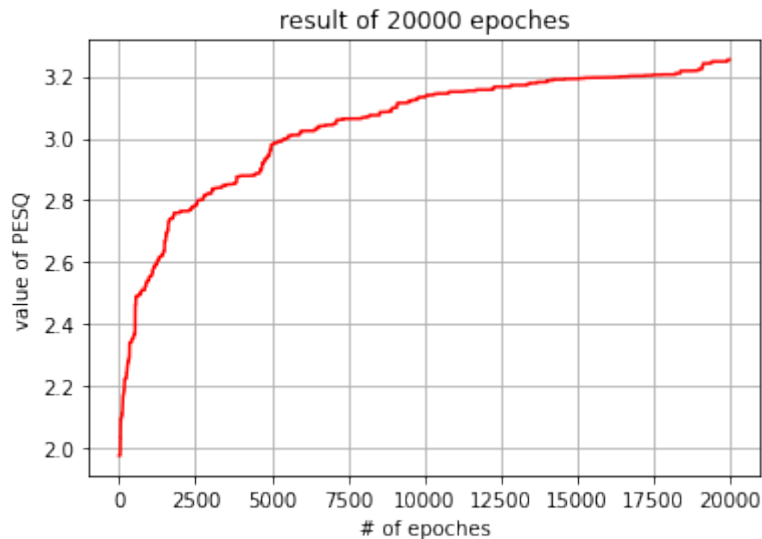


图 5-7: Xi-Q 收敛图

图 5-6 (b) 为 Xi-Q 算法在迭代了 20000 次后的结果，相比较于 Deep Xi 的输出，Xi-Q 算法的增强结果得到了很明显的改善。Xi-Q 不仅大幅度降低

1.0 ~ 2.2s 内的噪声分量，而且改善了 2.0 ~ 3.5s 秒内信号的高频分量处理结果。

## 第六章 总结与讨论

### 6.1 总结

在本文中，我们以 Deep Xi 系统为例，研究强化学习在单通道语音增强系统中的应用，设计了基于 Deep Xi 系统的 Xi-Q 算法。该算法基于强化学习中的 Q-Learning 概念，依据  $\epsilon - greedy$  策略，让系统在不断的尝试中获得奖励或者惩罚，实现系统的自我学习。实验结果表明，Xi-Q 算法能够有效地提高 Deep Xi 系统的输出质量，改善系统性能，实现 Deep Xi 系统的自优化。

### 6.2 讨论

在这一节中，我们讨论此研究未来的优化与发展方向。

#### 6.2.1 优化 Quality-Net

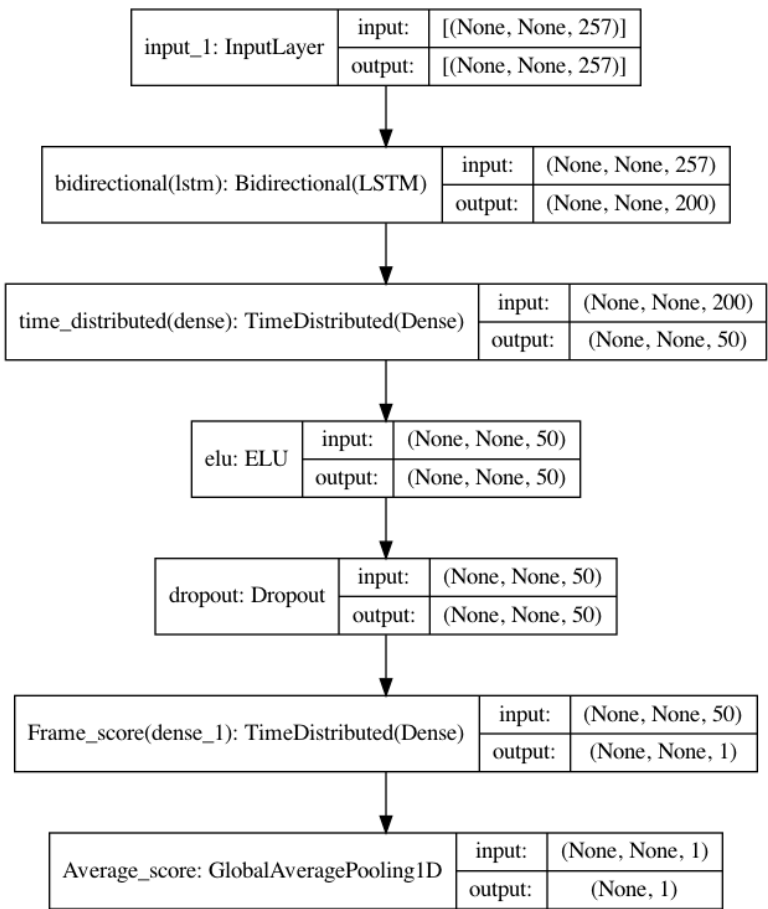


图 6-1: Quality-Net 的网络结构

如图 6-1 所示, 这是本文中 Quality-Net 的网络结构, 其隐藏层个数较少。目前 Quality-Net 的测试集方差为 0.3664, 对带噪语音 PESQ 的线性相关性为 0.9, 对 Deep Xi 增强后的语音 PESQ 的线性相关性为 0.84, 可见 Quality-Net 对 PESQ 的估计值没能达到较理想的效果。

训练过程中可以发现, 当 epoches 大于等于 10 之后, 验证集的 loss 就基本收敛了, 说明这个网络的性能已经达到饱和状态。如此少的 epoch 就将网络训练到收敛, 从侧面也反映了目前 Quality-Net 的网络结构比较简单。基于这些观察, 我们可以从如下几点研究来提升 Quality-Net 的预测准确度:

- 加入一维卷积层 (Conv1D layer)。目前 Quality-Net 用的是全连接层 (Dense layer), 几乎忽略了语音信号前后帧之间的联系, 并且所需的参数量也较大。加入卷积层后, 通过发挥卷积神经网络的特性, 能将语音信号前后帧的关联性考虑进去, 作出更加合理的预测, 同时也能降低所需参数量;
- 调整隐藏层层数和每层的神经元个数。目前这个网络规模还比较小, 每层神经元个数也相对较少。适度调整 Quality-Net 的整体规模可以优化其映射性能。

### 6.2.2 优化奖励函数

提升奖励函数的合理性是 Xi-Q 算法改善其自优化效果的关键。

本文所设计的奖励函数的更新源于两个部分, 一是整体 PESQ 值的高低, 二是局部错误先验信噪比的估计。在未来的工作中, 我们可以加入更多的评判标准 (如 STOI 和 PEASS) 来对语音增强的结果作出判断, 从而设计出更加合理的, 更贴近于主观语音质量评价的奖励函数。

### 6.2.3 优化先验信噪比模版

本文中定义 Xi-Q 算法的模版为同一批数据所有先验信噪比的估计值, 实验过程中发现已经达到了几万个。模版的合理选择对 Xi-Q 算法作出合理的动作选择十分重要, 过大的模版数量会使得作出错误决定的概率增大, 相应的学习速率也会降低。在未来的研究中, 我们可以尝试平均, 线性映射或者聚类算法来降低模版数量, 在减少学习时间的同时作出更好的决定。

## 参考文献

- [1] EPHRAIM Y, MALAH D. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator[J/OL]. IEEE Transactions on Acoustics, Speech, and Signal Processing, 1984, 32(6): 1109-1121. DOI: 10.1109/TASSP.1984.1164453.
- [2] ERKELENS J S, HENDRIKS R C, HEUSDENS R, et al. Minimum mean-square error estimation of discrete fourier coefficients with generalized gamma priors[J/OL]. IEEE Transactions on Audio, Speech, and Language Processing, 2007, 15(6): 1741-1752. DOI: 10.1109/TASL.2007.899233.
- [3] DIONELIS N, BROOKES M. Phase-aware single-channel speech enhancement with modulation-domain kalman filtering[J/OL]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2018, 26(5): 937-950. DOI: 10.1109/TASLP.2018.2800525.
- [4] WANG Z Q, LE ROUX J, HERSHEY J R. Multi-channel deep clustering: Discriminative spectral and spatial embeddings for speaker-independent speech separation[C/OL]//2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2018: 1-5. DOI: 10.1109/ICASSP.2018.8461639.
- [5] HU J, NIU H, CARRASCO J, et al. Voronoi-based multi-robot autonomous exploration in unknown environments via deep reinforcement learning[J/OL]. IEEE Transactions on Vehicular Technology, 2020, 69(12): 14413-14423. DOI: 10.1109/TVT.2020.3034800.
- [6] CHANG J H, KIM N S, MITRA S. Voice activity detection based on multiple statistical models[J/OL]. IEEE Transactions on Signal Processing, 2006, 54(6): 1965-1976. DOI: 10.1109/TSP.2006.874403.
- [7] HUGHES T, MIERLE K. Recurrent neural networks for voice activity detection[C/OL]//2013 IEEE International Conference on Acoustics, Speech and Signal Processing. 2013: 7378-7382. DOI: 10.1109/ICASSP.2013.6639096.
- [8] GELLY G, GAUVAIN J L. Optimization of rnn-based speech activity detection[J/OL]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2018, 26(3): 646-656. DOI: 10.1109/TASLP.2017.2769220.
- [9] MARTIN R. Spectral subtraction based on minimum statistics[J]. power, 1994, 6(8).

- [10] MARTIN R. Noise power spectral density estimation based on optimal smoothing and minimum statistics[J/OL]. IEEE Transactions on Speech and Audio Processing, 2001, 9(5): 504-512. DOI: 10.1109/89.928915.
- [11] RANGACHARI S, LOIZOU P C. A noise-estimation algorithm for highly non-stationary environments[J]. Speech communication, 2006, 48(2): 220-231.
- [12] DOBLINGER G. Computationally efficient speech enhancement by spectral minima tracking in subbands[C]//Fourth European conference on speech communication and technology. [S.l.: s.n.], 1995.
- [13] HENDRIKS R C, HEUSDENS R, JENSEN J. Mmse based noise psd tracking with low complexity[C]//2010 IEEE International Conference on Acoustics, Speech and Signal Processing. [S.l.]: IEEE, 2010: 4266-4269.
- [14] EPHRAIM Y, MALAH D. Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator[J]. IEEE Transactions on acoustics, speech, and signal processing, 1984, 32(6): 1109-1121.
- [15] NICOLSON A, PALIWAL K K. Deep learning for minimum mean-square error approaches to speech enhancement[J]. Speech Communication, 2019, 111: 44-55.
- [16] KOIZUMI Y, NIWA K, HIOKA Y, et al. Dnn-based source enhancement self-optimized by reinforcement learning using sound quality measurements[C/OL]//2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2017: 81-85. DOI: 10.1109/ICASSP.2017.7952122.
- [17] MALFAIT L, BERGER J, KASTNER M. P.563—the itu-t standard for single-ended speech quality assessment[J/OL]. IEEE Transactions on Audio, Speech, and Language Processing, 2006, 14(6): 1924-1934. DOI: 10.1109/TASL.2006.883177.
- [18] MITTAG G, NADERI B, CHEHADI A, et al. Nisqa: A deep cnn-self-attention model for multidimensional speech quality prediction with crowd-sourced datasets[J]. arXiv preprint arXiv:2104.09494, 2021.
- [19] FU S W, TSAO Y, HWANG H T, et al. Quality-net: An end-to-end non-intrusive speech quality assessment model based on blstm[C]//Interspeech. [S.l.: s.n.], 2018.
- [20] JOZEFOWICZ R, ZAREMBA W, SUTSKEVER I. An empirical exploration of recurrent network architectures[C]//International conference on machine learning. [S.l.]: PMLR, 2015: 2342-2350.



- [21] PANAYOTOV V, CHEN G, POVEY D, et al. Librispeech: An asr corpus based on public domain audio books[C/OL]//2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2015: 5206-5210. DOI: 10.1109/ICASSP.2015.7178964.
- [22] VEAUX C, YAMAGISHI J, MACDONALD K. Cstr vctk corpus: English multi-speaker corpus for cstr voice cloning toolkit[C]//[S.l.: s.n.], 2017.
- [23] GAROFOLO J S, LAMEL L F, FISHER W M, et al. Darpa timit acoustic-phonetic continuous speech corpus cd-rom. nist speech disc 1-1.1[J]. NASA STI/Recon technical report n, 1993, 93: 27403.
- [24] AUTHOR A A, AUTHOR B B, AUTHOR C. The qut-noise-timit corpus for the evaluation of voice activity detection algorithms[J]. Scopus, 2005, 10(2): 49-53.
- [25] SAKI F, SEHGAL A, PANAHI I, et al. Smartphone-based real-time classification of noise signals using subband features and random forest classifier[C/OL]//2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2016: 2204-2208. DOI: 10.1109/ICASSP.2016.7472068.
- [26] SNYDER D, CHEN G, POVEY D. Musan: A music, speech, and noise corpus[J]. arXiv preprint arXiv:1510.08484, 2015.
- [27] KABAL P. Tsp speech database[J]. McGill University, Database Version, 2002, 1(0): 09-02.
- [28] SALAMON J, JACOBY C, BELLO J P. A dataset and taxonomy for urban sound research[C]//Proceedings of the 22nd ACM international conference on Multimedia. [S.l.: s.n.], 2014: 1041-1044.
- [29] STEENEKEN H J, GEURTSSEN F W. Description of the rsg-10 noise database[J]. report IZF, 1988, 3: 1988.



## 致 谢

感谢南京大学物理学院声科学与工程系卢晶教授对本课题从论文选题，到开题报告，再到最终定稿的悉心指导。

特别感谢实验室的雷桐学姐一直以来对此论文撰写的耐心帮助。论文起步阶段，雷桐学姐编写了 GPU 服务器操作手册，让我很快熟悉了实验室服务器的相关内容。之后，每次向学姐请教论文的相关问题，例如 Deep Xi 系统的模型选择和客观语音质量评估标准等，学姐都会作出非常细致的解答，大幅提高了我的工作效率。此外，和学姐进行的多次关于算法改进的交流也让我受益匪浅。

感谢同系的侯仲舒同学对实验室 GPU 服务器相关问题的解答。