电 子 科 技 大 学

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

硕士学位论文

MASTER THESIS

论文题目 **面向复杂环境的语种识别方法研究**

学科专业	软件工程
学　　号	**202021090323**
作者姓名	陈聪
指导教师	蓝 天　研究员
学　　院	信息与软件工程学院

分类号 _____ 密级 _____

UDC [注1] _____

学 位 论 文

面向复杂环境的语种识别方法研究

（题名和副题名）

陈聪

（作者姓名）

指导教师 蓝天 研究员

电子科技大学 成 都

（姓名、职称、单位名称）

申请学位级别 硕士 学科专业 软件工程

提交论文日期 2023 年 3 月 16 日 论文答辩日期 2023 年 5 月 11 日

学位授予单位和日期 电子科技大学 2023 年 6 月

答辩委员会主席 _____

评阅人 _____

注 1：注明《国际十进分类法 UDC》的类号。

Research on Language Identification Method for Complex Environment

A Master Thesis Submitted to
University of Electronic Science and Technology of China

Discipline **Software Engineering**

Student ID **202021090323**

Author **Cong Chen**

Supervisor **Prof. Tian Lan**

School **School of Information and Software Engineering**

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

作者签名：陈聪　　日期：2023 年 5 月 25 日

论文使用授权

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定，同意学校有权保留并向国家有关部门或机构送交论文的复印件和数字文档，允许论文被查阅。本人授权电子科技大学可以将学位论文的全部或部分内容编入有关数据库进行检索及下载，可以采用影印、扫描等复制手段保存、汇编学位论文。

（涉密的学位论文须按照国家及学校相关规定管理，在解密后适用于本授权。）

作者签名：陈聪　　导师签名：[illegible]

日期：2023 年 5 月 25 日

摘　要

语音是人类与周围环境之间进行信息交互的基本载体之一，在人类的活动中扮演着重要的角色。随着社会现代化的发展，使得包括语音识别在内的智能语音技术逐渐成为研究热门。但在现实场景下，存在各种类型的噪声干扰且信噪比未知，如何有效的提高在复杂噪声场景下智能语音算法的性能是当前热点问题。语音增强技术用于抑制噪声，因此其通常会作为智能语音算法的前端。语种识别是指自动判定语音所属语言种类的过程，是多语言环境下智能语音算法的关键前端技术。

然而基于深度学习的语音增强和语种识别技术仍然存在以下挑战：（1）语音增强模型难以适应未知的噪声类型和信噪比；（2）在训练和测试条件不匹配时，语种识别算法泛化性和鲁棒性较差；（3）模型在低资源语种数据上表现较差。（4）语音增强目标和下游任务之间不匹配。本文以语音增强为切入点，研究了面向复杂环境下的语种识别技术及其应用，针对以上问题，提出了相应的解决方案，主要贡献如下：

（1）针对噪声和信噪比自适应问题，本文提出了一种基于动态选择核机制的语音增强算法，通过为具有不同感受野大小的卷积核所对应的卷积模块赋予不同的权重，来动态的调整卷积层的感受野大小，从而使得模型能够适应不同的噪声类型和信噪比环境。为了有导向性的引导模型朝着目标优化，本文引入了中继监督损失机制，进一步提升了模型性能。

（2）针对语种识别泛化性问题，本文提出了一种基于词法和语法特征的语种识别方法，实验表明相较于其他基于底层声学特征的方法具有更优的泛化性能。同时针对低资源语种识别准确率较差的问题，本文利用自监督模型在大规模无标注数据上学习到的语音表征，进一步提升了语种识别的准确率。针对语音增强引入的失真导致语种识别性能下降的问题，本文探索融合带噪语音和增强语音的方式，进一步提升了语种识别模型在噪声环境下的性能。

（3）最后在算法的基础上，本文设计了一个面向民航陆空通话的语种识别系统，在系统中，针对基于深度学习的模型部署计算效率低下问题，提出了一种消息批处理框架，有效的提高了系统的吞吐量。

关键词：语音增强，语种识别，自监督模型，多语种语音识别

ABSTRACT

Speech is one of the basic carriers of information interaction between human beings and their surroundings, and plays an important role in human activities. With the development of social modernization, intelligent speech technology, including speech recognition, has gradually become a research hotspot. However, in the real scene, there are various types of noise interference and the signal-to-noise ratio is unknown. How to effectively improve the performance of intelligent speech algorithms in complex noise scenes is a hot issue at present. Speech enhancement technology is used to suppress noise, so it is usually used as the front end of intelligent speech algorithm. Language identification refers to the process of automatically determining the language type of speech, and it is the key front-end technology of intelligent speech algorithm in multilingual environment.

However, the speech enhancement and language identification technology based on deep learning still has the following challenges: (1) The speech enhancement model is difficult to adapt to the unknown noise type and signal-to-noise ratio; (2) When the training and testing conditions do not match, the generalization and robustness of the language identification algorithm are poor; (3) The model performs poorly on low-resource language data. (4) There is a mismatch between the speech enhancement target and the downstream task. Based on speech enhancement, this thesis takes speech enhancement as an entry point to study language recognition technology and its application in complex environments. Aiming at the above problems, a corresponding solution is proposed. The main contributions are as follows:

(1) Aiming at the adaptive problem of noise and signal-to-noise ratio, this thesis proposes a speech enhancement algorithm based on dynamic selection kernel mechanism. By giving different weights to convolution modules corresponding to convolution kernels with different receptive field sizes, the receptive field size of convolution layer is dynamically adjusted, so that the model can adapt to different noise types and signal-to-noise ratio environments. In order to guide the model towards the goal optimization, this thesis introduces the relay supervision loss mechanism to further improve the model performance.

(2) Aiming at the generalization of language identification, this thesis proposes a

language identification method based on lexical and grammatical features. Experiments show that it has better generalization performance than other methods based on underlying acoustic features. At the same time, in order to solve the problem of poor accuracy of language identification with low resources, this thesis uses the speech representation learned from large-scale unlabeled data by self-monitoring model to further improve the accuracy of language identification. Aiming at the problem that the distortion introduced by speech enhancement leads to the decline of language identification performance, this thesis explores the way of combining noisy speech and enhanced speech, which further improves the performance of language identification model in noisy environment.

(3) Finally, based on the algorithm, this thesis designs a language identification system for civil aviation land-air communication. In the system, a message batch processing framework is proposed to solve the problem of low computational efficiency of model deployment based on deep learning, which effectively improves the throughput of the system.

Keywords: speech enhancement, language identification, self-supervised model, multilingual speech recognition

目 录

第一章 绪 论

本文针对复杂噪声环境下的语种识别问题展开研究，利用深度学习技术提出了噪声自适应的语音增强算法和鲁棒语种识别模型，提升了语音在噪声种类多变、信噪比未知的复杂环境下的识别准确率。本章作为本文的绪论，首先对本文研究工作的背景和意义进行了阐述；然后总结梳理了语音增强和语种识别技术的研究历史与现状；接着对本文研究内容进行了概括总结；最后给出了论文的整体架构。

1.1 研究工作的背景与意义

古往今来，语音是人类与周围环境之间进行信息交流的基本载体，具有便捷、高效、及时的特点。我们每天从身边接受包括噪声、人类语音和音乐在内的各种声音，人类大脑可以很轻松的辨别这些声音的种类，然后再对声音进行更进一步分析，例如获取到语音所表达的含义、来源等，如果是人类语音还可以进一步识别到其语种、说话人性别、情感等更多的信息。在嘈杂的环境中也能专注于特定的语音，忽略掉无关的信号。通过语音我们还能向外界直接传达有效的信息，语音在人类的活动中扮演着重要的角色。

随着人类现代化的发展和生活水平的提高，人机交互在人类与外界环境之间的信息交互中占有着越来越重要的地位。智能语音技术就是实现人机交互的关键技术，它包括语音识别（Automatic Speech Recognition, ASR）、语音合成（Text To Speech, TTS）和自然语言处理（Natural Language Processing, NLP）在内的多项技术。人机交互需求广泛，在民用领域和军事领域都有广泛的应用。例如车载助手可以帮助驾驶员更方便的控制车辆状态；智能音箱能够解放人们的双手，便捷的控制家居设备；手机助手等语音交互软件提高了使用者的沟通效率。语音情报分析能够高效的挖掘特定信息，在国际反恐中有着重要的应用前景。

随着经济全球化的高速发展，在不同语种之间的交流和碰撞愈加频繁。据统计，目前世界上各国语言种类合计 6909 种[1]，不同语种之间在发音上存在较大差异，人类难以凭借个人力量掌握所有语言，这在一定程度上提高了交流成本，同时对智能语音技术提出了更高的要求。现有的智能语音技术大部分都是针对单一的特定的语言而设计，在面对多语种的环境时，往往需要提前确定语言种类才能进一步进行分析处理。语种识别（Language Identification, LID）是一种用于判定给定语音所属的语言种类的过程，是多语言环境下智能语音算法的关键前端技术。

然而现有语种识别系统的性能仍然不能满足日益增长的需求，智能语音技术

的应用不止局限在安静无噪声的理想场景下。在现实环境中，往往存在各种噪声干扰，且信噪比多样。在面对复杂的噪声环境时，包括语种识别技术在内的各种下游语音任务的性能都会急剧下降，从而限制了智能语音技术的广泛应用。全球语种分布也极其不均衡，汉语、英语、西班牙语、印地语、阿拉伯语和俄语等占据了全球大部分人口的第一语种，其余语种使用人数较少，数据集也十分匮乏，导致难以为这些低资源语种构建高性能的智能语音处理算法模型。可以看到，语种识别技术作为人机交互的关键前置处理技术，依然面临着不小的挑战，噪声干扰导致的语音可懂度降低、低资源语种数据集匮乏导致的性能下降都是该领域亟待解决的关键问题。本文面向复杂噪声环境，针对同一时刻只有单说话人的应用场景对语种识别技术和语音增强技术展开了研究。

1.2 研究历史与现状

1.2.1 语音增强的发展历程

语音增强（Speech Enhancement, SE）旨在抑制语音中的噪声成分，提高语音的感知质量和可懂度。通常作为信号处理前端用于各种应用中，例如语音识别[2, 3]、助听器[4]以及网络电话[5]等。语音增强技术可以根据麦克风数量分为单声道语音增强和多声道语音增强，单声道语音增强技术只使用一个声道的语音数据，相比于利用麦克风阵列的多声道语音增强其成本更低，计算速度更快，因此应用也更加广泛，但多声道之间可以形成信息互补，因此多声道语音增强具有更优的性能，近年来，随着计算机算力的提升，多声道语音增强成为了新的研究热点[6-8]。

早期对语音增强的研究基于信号处理算法，以无监督方法为主，例如谱减法[9, 10]，其假设语音中的噪声为加性噪声，在计算时，将带噪语音的频谱减去估计的噪声频谱，从而获得清晰的纯净语音。然而对于非平稳噪声，因为难以准确估计其在不同时间内的频谱分布，导致其应用受限。维纳滤波[11]是另一种降噪算法，其将语音增强任务视为将含噪语音通过一种线性系统的过程，假设带噪语音为源信号，通过一个线性滤波器后得到纯净的增强语音，再通过最小化增强语音和纯净语音之间的均方误差，求得滤波器参数。然而这种方法泛化性较差，依赖语音和噪声信号的统计特征，当估计出现偏差时，会出现所谓的音乐噪声[12]，影响增强语音质量。基于非负矩阵分解[13, 14](Nonnegative Matrix Factorization, NMF)的语音增强是另一种处理方法，其通过在大量的数据集中进行训练，构建语音信号和噪声的基矩阵，再将语音的频谱矢量映射到基矩阵上，从而重建语音信号。这种方法对于非平稳噪声表现良好，然而，在噪声未知的场景下泛化能力不足[15]。其他的传统语音增强方

法还包括最小均方误差估计法[16]（Minimum Mean Square Error，MMSE）、基于统计模型的方法[17]和子空间法等。

近年来，由于计算机算力的大幅提升，特别是高并行度的 GPU 的快速发展，基于深度学习的方法[18, 19]在包括 SE 任务在内的多个领域取得了革命性的进步。DNN 在拟合非线性问题上具有显著的优势，特别是对于复杂多变的声学场景下的语音增强任务。Xu 等人[20]利用多层感知器（Multilayer Perceptron, MLP）将带噪语音对数功率谱映射到干净语音，然而其并未考虑到语音的时序依赖性。Chen 等人[21]采用长短期记忆网络（Long Short-Term Memory, LSTM）对语音上下文关系进行建模，LSTM 是循环神经网络[22]（Recurrent Neural Network, RNN）的一种变体，在推理过程中能充分融合语音上下文信息，实验结果表明相较于 MLP，在未知噪声和多说话人条件下，LSTM 具有更好的性能。然而 LSTM 参数量较大，Cui[23]等人探索使用参数量更少的门控循环单元（Gate Recurrent Unit，GRU）和简单神经单元（Simple Recurrent Unit, SRU）来降低模型复杂度，并且在 PESQ 和 STOI 指标上取得了更好的效果。

基于卷积神经网络（Convolutional Neural Networks，CNN）的语音增强也受到了学术界的广泛关注，卷积结构能够充分捕捉图像中的局部信息，而语谱图可视为一张单通道 2 维图像，在模型一层一层的迭代中，逐渐将底层特征抽象为高维表示。2016 年 Park 等人[24]首次将 CNN 引入到语音增强领域，基于 CNN 的模型结构具有更少的参数量，性能却相较于 LSTM 等循环神经网络不相上下，因此逐渐成为研究主流。然而 CNN 难以充分抓取全局信息，而 RNN 擅长捕获上下文关联信息，因此 Tan 等人[25]将 RNN 引入到卷积神经网络中，命名为 CRN，他们还通过引入门控卷积单元，进一步提升了语音的感知质量。以上方法都是基于频域进行建模，由于相位谱中缺乏清晰的结构，通常只利用幅度谱作为底层声学特征。有研究表明，相位信息在低信噪比的语音增强扮演着重要的角色[26]。Tan[27]等人又提出了一种利用复数谱映射的卷积循环神经网络，用于对噪声和说话人无关的含噪语音建模，所提出的方法优于原始的 CRN 模型。然而其仅是简单的将实部和虚部视为两个输入通道，使用共享的卷积滤波器进行卷积运算。Hu 等人[28]提出了一种用于复数运算的深度卷积循环神经网络（Deep Complex Convolution Recurrent Network, DCCRN），在计算时通过模拟复数乘法来对相位和幅度之间的相关性建模，在降低计算复杂度的同时获得了更优的性能，在 2020 年 DNS 挑战赛[29]中取得了最佳的成绩。

以上方法都是基于语音信号的频域建模，另一种能充分利用相位信息的方式是直接将时域信号作为输入，由于时域波形结构特征不明显，难以直接利用，但随

着深度学习的发展，深度神经网络强大的拟合能力使得直接使用时域信号构建端到端语音增强框架成为可能[30]。因为语音信号的短时平稳性，短时傅里叶变换的窗口通常选取低于 25ms 区间的固定值，以 10ms 作为帧移长度，在转化过程中信号会丧失平滑度。而对于时域信号，在处理过程中帧长可以选择任意大小，从而可以根据噪声信噪比水平进行调整，有研究表明，通过将帧大小设置为非常小的值时能有效提升说话人分离网络的性能[31]。使用时域信号建模能避免信号在频域之间转换导致的计算，并且基于深度神经网络训练的特征可能比频谱更具优势。Pandey 等人[32]提出了一种用于时域实时语音增强的全卷积神经网络，其基于编码器解码器架构，中间插入了一个 TCM 因果和扩张卷积模块，实验结果表明和 CRN 相比，具有更好的增强性能。随后其又提出了一种稠密卷积网络（Dense Convolutional Network, DCN）,其每一层包含一个密集块和自注意力模块，同时引入了一种基于增强语音谱和预测噪声谱的额外损失函数。扩散概率模型[33]是一类在图像生成[34]、语音合成[35, 36]任务中表现出色的一种生成式模型。其在逐步扩散过程中将干净的输入数据转换为各向同性的高斯分布，并在反向过程中，通过预测和去除在扩散过程的每一步中引入的噪声来逐渐恢复干净的输入。Lu 等人[37]将扩散模型引入到语音增强领域，由于噪声特征通常是非高斯的，其通过制定了一个广义条件下的扩散概率模型将观察到的噪声数据引入到模型中，实验证明其性能优于其他生成式模型。

1.2.2 语种识别发展历程

语种识别是多语言场景下人机交互中的重要技术。近几十年来，得益于信号处理、模式识别和机器学习领域的飞速发展，语种识别技术取得了巨大的进步。作为智能语音处理系统前端，是下游任务例如多语言语音识别[38]的关键支持技术。作为人类，我们可以通过听觉系统固有的声学和心理学认知过程来识别语言[1]。研究表明，人类进行语种判别时所利用的声学特征主要包括两方面，一种是词前（prelexical）信息，包括韵律、语调和音素等；另一种是词后（post lexical）信息，即语义内容和语法[39]。对婴儿进行听力实验的研究表明，当婴儿没有获得大量词汇知识时，他们成功地依靠词前信息区分了语种属性。随着婴儿语言知识的积累，词后信息开始发挥着越来越重要的作用。受到人类语种判别逻辑的启发，在计算机语种识别技术的研究中，将用于区分语种的特征从底层到高层分别分为声学特征、音素特征、韵律学特征、词法和语法特征。接下来将分点对这些不同基于特征的语种识别方法进行概述。

（1）基于词法语法特征的语种识别方法

人类语言包含了一个控制单词组合的词法系统和控制词与词搭配的语法系统，每种语种都有其独特的词法语法特征，这种特征是人类判断语言种属时的首选，听力实验表明，人类在深入了解多门语言后能够快速准确的判断这些语种中的语音。然而，词法和语法作为语言高级特征难以直接抽取，通常会基于语音识别（Automatic Speech Recognition, ASR）系统，从多个不同语种的 ASR 模型中生成每一个语种的文本作为词法和语法特征[40-42]。由于受限于计算成本和 ASR 系统的性能约束，这种方式并没有受到研究者的重视。近年来，随着深度学习的发展，ASR 准确率达到了人类的水准[43]，因此可以获得准确率更高的文本特征。早期基于词法特征的语种识别采用基于 HMM 的语音识别系统作为前端，通过 N-gram 语言模型进行打分，综合 ASR 的置信度和语言模型分数作为多分类器的输入进行语种识别[41]，这种方法在语种较少时表现出较好的性能。Okamoto 等人[44]从模型延迟角度出发，探索了这种方法带来的实时性和性能的取舍。Kukk 等人[45]在无监督预训练模型 wav2vec 基础上进行迁移训练构建单 ASR 模型，随后基于 ASR 的输出构建基于卷积神经网络的语种分类模型，受益于预训练模型参数的初始化，迁移后的 ASR 模型具有较好的性能，实验表明其对于非母语的语音也具有较好的泛化性。总的来说，基于词法和语法特征的语种识别研究还未受到学者的广泛关注，然而计算机性能的提升使得基于 ASR 的语种识别应用成为了可能。

（2）基于韵律特征的语种识别方法

韵律一般是指语音中的超音段特征，例如重音、时长、节奏和语调等，不同语种之间存在显著差异。声调是语言最突出的特征，例如汉语有 4 种声调，分别为阴平，阳平，上声，去声，越南语共有 6 种声调，即平声、玄声、问声、跌声、锐声、重声，而印欧语系是无声调语系。因此韵律可以作为一种特征区分语言类别。Lin 等人[46]提出了一种遍历拓扑动态模型，利用语音音高（pitch）特征对语种进行建模。Ng 等人[47]提出了一种新颖的韵律属性模型（Prosodic Attribute Model, PAM）来捕获韵律特征，对特定语种的韵律进行共现统计，在 NIST 语种识别评估种取得了不错的成绩。Lee 等人[48]基于支持向量机（Support Vector Machine, SVM）的词袋 n-gram 方法对韵律属性建模，并将更多的属性纳入韵律特征，如基频、归一化属性、基频残差等。Koolagudi 等人[46]结合 MFCC 特征和韵律特征进行建模，最后利用主成分分析技术 PCA 对特征向量降维，实验表明比起只使用单一的 MFCC 特征性能提升 18%。虽然有不少学者使用基于韵律的方法进行语种识别的研究，然而韵律所携带的信息有限，语音的其他特征更适合对语种进行建模[49]，韵律特征只有当语音长度在特定的范围内才更有效[50]，因此主流的研究专注于其他特征。

（3）基于音素分类的语种识别方法

音节是人类发音的基本单元，而音素是构成音节的最小单元，每种语言都有其独特的词汇-发音规则来控制其音素组合，而其音素顺序和频率统计特征具有较大差异，因此通过音素之间的组合区分语种类别是一种有效的语种识别方法。通常基于音素的语种识别模型分为两个部分：音素识别器和音素语言模型[1]。音素识别器用于将音频序列转化为音素序列，由于不同语种可以共享同一套音素系统，音素识别模型可基于一种语言进行训练[51]。早期，以高斯混合模型[52]（Gaussian Mixture Model， GMM）为代表的机器学习算法成为研究主流，其基于语音的梅尔倒谱系数（Mel-scale Frequency Cepstral Coefficients, MFCC）[53]特征进行训练，在测试期间，输出 GMM 计算中得分最高的高斯分量索引，作为当前帧的音素。随着深度学习的发展，涌现出更多的音素识别算法，例如 Liu 等人[54]利用 DNN 进行特征抽取，提出了联合 DNN 和 GMM 的音素识别算法；Chen 等人[55]利用生成对抗网络进行伪标签标注，提升了识别准确率。

音素语言模型用于评价某一音素序列在特定语种下的合理程度，通常基于统计概率模型 N-gram 建模，N 表示统计的基本单元长度。在进行语种种属判定之前，会为每个语种单独构建一个 N-gram 模型，用于衡量音素序列在该语言模型下的困惑度。模型推理时，首先通过音素识别器获取到音素序列，再将音素序列通过每个提前构建好的语言模型，进而根据输出的困惑度大小判断语种类别，这种方法叫做结合音素识别的语言模型建模方法（Phoneme Recognition followed by Language Modeling, PRLM）[56]。然而基于单一语种进行音素建模存在缺陷，不同语种之间的音素在统计上存在较大差异，导致在识别其他语种时候精度下降，因此又有学者提出了一种并行的 PRLM 方法叫做 PRLM-P[56]，其本质上是一种多路的 PRLM 模型，语音通过在不同语种条件下训练的多个音素识别器，再分别由语言模型进行打分，最后综合多个评分进行判别，这种方法基于多个语种进行音素建模，不同语种之间的音素识别模型可以形成互补，从而提升语种识别准确率。由于单个语种语料构建的音素识别模型包含了语种大量的先验信息，随后又有学者提出了一种只利用音素识别模型进行语种判别的方法叫做并行音素识别方法（Parallel Phoneme Recognition, PPR）[57]，其利用音素模型输出的后验概率，通过维特比解码算法找出最有可能的音素序列，并计算其平均似然，进一步判断其语种。

（4）基于底层声学特征的语种识别方法

基于底层声学特征建模是语种识别的主流方法，其原理是通过利用底层声学特征的统计差异来实现语种的分类，这种特征可以通过语音信号处理方法较为容易的得到，从而更加方便的构建端到端语种识别模型。常见的底层特征包括梅尔频率倒谱系数(Mel-frequency Cepstral Coefficients, MFCC)、线性频率倒谱系数（Linear

Frequency Cepstral Coefficients, LFCC）[58]、线性预测编码（Linear Predictive Coding, LPC）[53]和 FBank[59]等。语种识别和说话人识别在方法和评价指标上有较多的相似之处，基于统一背景模型的高斯混合模型（GMM-UBM）[60]是一种受到说话人识别研究启发的分类模型，GMM 一大优势是能够拟合任意形状的数据分布，然而却很容易受到说话人和信道的干扰。有学者尝试了更优的训练方法来提高判别能力，如受限最大似然线性回归（Constrained Maximum-likelihood Linear Regression(CMLLR)）[61]、软边际估计（Soft Margin Estimation）[62]、和最大互信息训练（MMI training）[63]等。i-vector 同样是说话人识别领域的一种方法，I-Vector 模型为语音序列提取一个一维的特征向量，再利用线性判别分析（LDA）实现语种的分类，Najim 等人[64]将其引入到语种识别中，在 2009 年的 LRE 任务中取得了较大的性能提升。

随着深度学习的发展，基于机器学习的语种识别占据主流。Lopez 等人[65]首次将深度神经网络引入语种识别，利用 DNN 的拟合能力对特征进行进一步提取，取得了优异的性能。x-vector[66]是一种利用时间延迟网络（Time Delay Neural Network TDNN）[67]构建的模型，和 i-vector 类似，该网络将语音特征序列映射为特定维度的嵌入向量，称为 x 向量，其在 2017 年的 NIST 比赛中取得了优于 i-vector 的成绩。由于语种识别技术在实际应用中的环境较为复杂，噪声多样，提升语种识别的环境鲁棒性成为了一个研究热点。2017 年，Bartz 等人[68]使用 CNN 和双向 LSTM 的混合模型，联合基于 CNN 的特征提取和 LSTM 的时间维度信息抽取模块，适用于一系列嘈杂的场景，并且可以轻松扩展到以前未知的语言，同时保持其分类精度。2018 年，Vuddagiri 等人[69]提出了一种基于课程学习和注意力机制的噪声鲁棒性语种识别方法，减少了环境噪声导致的 LID 性能下降问题。2021 年邵玉斌等人[70]提出了一种基于滤波对数语谱图图像处理语种识别方法，作者对语音信号进行滤波后，提取其灰度对数语谱图，最后使用主成分分析提取图像的特征，比起传统的特征提取方式，在噪声环境下正确率有所提升。然而处理流程较为复杂，计算复杂度较高。

语种识别通常作为下游任务的前端，辅助在多语种场景下的语音识别等任务。为了更好的辅助下游任务，有学者提出了联合语音识别和语种识别的多任务训练方法[71, 72]，结合语言学信息不仅提高了语音识别的准确率，还大幅提高了语种识别的识别精度。然而新增加一个语种需要重新训练整个模型，难以广泛应用。近年来，无监督学习在自然语言处理(Natural Language Processing, NLP)大放光彩，在多个任务上取得了卓越的性能，不少学者在语音领域也提出了多种无监督框架[73-75]，Tjandra 等人[76]将无监督 wav2vec 模型成功的应用在语种识别领域，仅仅使用 10

分钟少量的语音，比起使用不使用预训练模型，语种识别精度从 7%提升到 87%，显示出巨大的潜力。

1.3 拟解决的关键问题

(1) 语音增强的复杂噪声适应问题

基于深度学习的语音增强学习带噪语音到干净语音的映射，在图像领域表现出色的卷积神经网络被应用于语音增强，CNN 可以通过在时间维度上进行卷积来利用上下文信息，在增强任务中表现出优秀的性能。然而在不同噪声以及说话人和语音内容不同的情况下，语谱图的特征分布将发生变化，基于卷积的端到端模型如果使用固定感受野进行特征抓取，则难以在所有条件下表现良好，难以适应复杂多样的噪声环境。

(2)语种识别环境泛化性问题

目前，基于深度学习的语种识别效果显著，但仍存在泛化性问题，即如果测试条件和训练条件不匹配，例如说话环境，说话人不同和录制设备的不同都会对识别性能产生较大影响，导致语种识别难以应用于实际的下游任务中。在噪声种类复杂多变、录制设备多种多样、说话人未知的复杂环境下，如何使语种识别在面对复杂环境时保持稳定工作是一个值得研究的问题。

(3) 低资源语种识别问题

深度学习技术往往依赖于大量的数据集，但目前大规模语音数据集主要集中分布在常见的几种语种，例如汉语、俄语、英语等，然而世界上总共有约 6900 种语言，对于低资源的语种，现有的技术往往会导致识别效果较差，如何在低资源的环境下构建高性能的语种识别系统成为当前的研究难点。

1.4 本论文的结构安排

本文的章节内容安排如下：

第一章为绪论，主要介绍了本文的研究背景和研究现状，对拟解决的关键对题做了详细说明。

第二章为相关技术背景及算法，主要介绍了语音信号处理中的相关知识和模型算法。首先从人类发声原理和听觉感知出发，引出了 FBank 声学特征提取。随后介绍了常见的神经网络模型，最后对本文用到的评价指标做了一一介绍。

第三章为语音增强算法设计，在 RCNA 端到端语音增强算法的基础上，引入了动态选择核机制核中继监督损失，有效的提升了语音增强模型在复杂场景下的自适应能力，通过详细的实验证明了模型结构的有效性。

第四章为语种识别算法设计，提出了两种分别基于有监督和无监督的多语种语音识别模型，在这个模型基础上，利用其词法和语法特征构建了一种泛化性和鲁棒性更好的语种识别方法，通过大量的实验对所提出的算法进行了详细的比较和分析。

第五章为原型系统设计，设计了一种面向民航陆空通话中英文双语的语种识别系统，用于辅助下游语音识别等任务。同时提出了一种批处理架构有效的提升了系统的吞吐量。

第六章为总结与展望，对本文的研究成果进行了概括和总结，并针对当前语种识别中存在的问题，对未来的研究方向进行了展望。

第二章 相关技术背景及算法

本章主要介绍语音增强和语种识别涉及到的语音信号处理和深度学习算法。首先对本文用到的FBank语音特征以及涉及到的短时傅里叶变换作了详细的分析，接着介绍了基本的神经网络结构，最后对本文用到的评价指标进行了简单说明。

2.1 语音特征提取

特征提取是深度学习中一个重要的环节。对于语音信号，其时域特征不够明显且容易受到噪声干扰，频域特征和时域特征相比具有更加清晰的时空结构，因此通常会将频域的特征作为建模的基本单元。本节将首先从声音发声机理出发，详细介绍人类语音产生原理和听觉感知过程，随后介绍语音信号处理中常用的特征提取技术：短时傅里叶变换，以及用于语种识别的 FBank 特征提取方法。

2.1.1 人类发声机理

人类语音的产生是一个复杂的过程，涉及包括嘴唇、舌、咽喉、气管和肺在内的多个器官参与。在发声时，通过肺部呼出气体的方式产生稳定气流，气流随气管到达咽喉，咽喉由肌肉、韧带和软骨等组织组成，能够控制声带的开闭，声带是一个类似阀门的结构，具有呼吸、发声和闭合三种不同的状态。呼吸状态不参与发声过程，是人类正常呼吸所处的状态。当处于发声状态时，声带在喉咙的控制下周期性的开闭，形成了一股气流脉冲，从而带动空气振动产生声波。而震动的周期是语音特征中的一个重要参数：基音（pitch）周期。其频率也被称为基频。人类的基频各不相同，但处于一个特定的范围，通常男性的基频在 100Hz~200Hz 范围内，平均基频 160Hz，女性基频在 200~350Hz 范围内，平均基频 297Hz，随着年龄增加，基频会逐渐降低[77]。

声门到嘴唇之间的器官：口腔、鼻腔和咽等组成声道，其形状会随着人体对声道器官的控制而变化，从而构成一个可动态调整参数的谐振腔，用于对声带产生的气流脉冲进行在不同频率上的增益控制，被放大的频率称为共振峰（formant）频率。通常在功率谱图像上的前三个共振峰是描述声道特性的关键参数。最后声道产生的信号经由嘴唇或鼻孔向外界辐射，即形成声波。

语音信号的产生可以看作一种线性模型，每个阶段用数学语言进行描述，最广泛使用的一种模型叫做激励-滤波（source-filter）线性模型，是语音信号处理的理论基础。如图 2-1 所示。

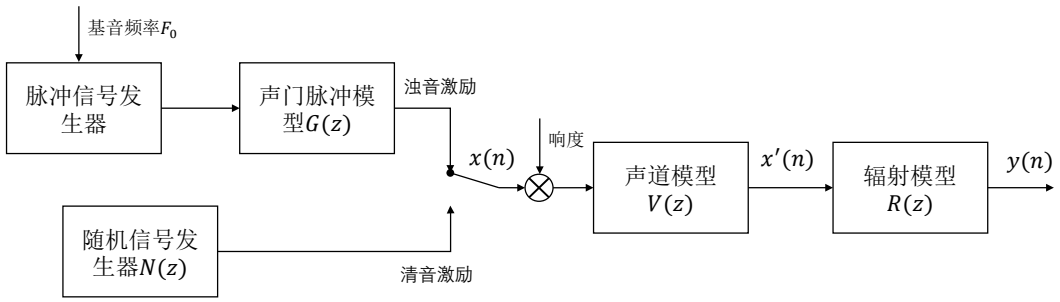


图 2-1 激励-滤波线性模型

其中，声音根据是否由声带振动产生分为清音（Unvoiced sounds）和浊音(Voiced sounds)。浊音就是由声带周期性闭合产生的脉冲气流，通过谐振腔增益发出的声音，如 a，o。脉冲信号发生器对应肺部产生的气流带动声带周期性开闭的过程，声门脉冲模型用于建模信号的采样过程。清音则不通过声带振动，而是通过收缩声道的某个部位形成狭窄通道，气流的通过会形成湍流，并激发出声音，如 t，d。另一种清音也被称为爆破音，是由声道突然的打开并释放气流而产生，如 b，p。声道模型对应声道谐振腔调制的过程，辐射模型对应嘴唇或鼻腔的辐射。

脉冲信号发生器产生的脉冲信号类型斜三角波g(n)，如式(2-1)所示：

$$\mathrm{g(n)}=\begin{cases}\dfrac{1}{2}\cos(1-cos\dfrac{n\pi}{N_1}) & 0\le n\le N_1\\ \cos(\dfrac{n-N_1}{2N_2}\pi) & N_1\le n\le N_1+N_2\\ 0 & other\end{cases} \tag{2-1}$$

其中N_1为斜三角波单调递增时间，N_2为单调递减时间。频域分析表明，其 Z 变换为一个二阶极点模型，可表示为：

$$G(\mathrm{z})=\frac{1}{(1-g_1z^{-1})(1-g_2z^{-1})} \tag{2-2}$$

其中g_1、g_2为模型参数。声门脉冲模型为可看作是一定采样间隔的单位阶跃序列对脉冲信号激励的过程。时域表达式为：

$$\mu[\mathrm{n}]=\begin{cases}1 & n\ge 0\\ 0 & n<0\end{cases} \tag{2-3}$$

其 Z 变换可表示为：

$$E(\mathrm{z})=\frac{A_v}{1-z^{-1}} \tag{2-4}$$

其中A_v表示响度系数，因此浊音激励模型可由下式给出：

$$\mathrm{U(z)} = G(z)E(z) = (\frac{A_v}{1-z^{-1}}) \cdot (\frac{1}{(1-g_1z^{-1})(1-g_2z^{-1})}) \tag{2-5}$$

声道模型通常有两种建模方法，第一种方法叫做声管模型，其假设声道是由多个具有不同横截面的管道组成，再将截面积大小作为参数对声道进行建模。另一种方法称为共振峰模型，将声道视为谐振腔，根据其谐振频率对声道进行建模，这种方式被广泛应用于语音信号分析中。通常有三种共振峰模型对具有不同特征的音素建模：级联型、并联型和混合型。对于元音音素，使用级联型模型，其假设声道由多个二阶谐振器串联构成，使用全极点模型对其进行描述，其中每个极点对应一个共振峰：

$$V(\mathrm{z}) = \frac{G}{1-\sum_{k=1}^{P} a_k z^{-k}} \tag{2-6}$$

对于鼻音和辅音，通常使用并联型模型，其为一种零极点模型：

$$V(\mathrm{z}) = \frac{\sum_{r=0}^{R} b_r\, z^{-r}}{1-\sum_{k=1}^{P} a_k z^{-k}} \tag{2-7}$$

而将两种模型进行结合便称为混合型模型。辐射模型对应声音从嘴唇发出到人耳感知到的这个过程，由于空气信道的影响，在传播的过程中，信号会随着距离衰减，通常使用一阶的高通滤波器对这个过程进行建模：

$$R(\mathrm{z}) = R_0(1-\alpha z^{-1}) \tag{2-8}$$

2.1.2 人类听觉感知

人类对语音的感知过程主要涉及的器官为耳，人耳从功能上可划分为三个部分：外耳、中耳和内耳。声波首先通过空气传播到达外耳，外耳是耳翼到鼓膜之间的这一部分。耳翼有着独特的结构，可以辅助判断出声音的方位。耳翼和鼓膜之间的通道称为耳道，其相当于一个共振腔，对 3k~4.5kHz 范围内的频率具有 15dB 的增益作用，从听觉上来说会对这部分频率更加敏感。鼓膜到耳蜗之间的部分叫做中耳，是一个立体空间，当声波由耳道到达鼓膜后，会带动鼓膜振动，而鼓膜振动会进一步引起中耳中听小骨的振动，镫骨是听小骨的一种，其紧连耳蜗的椭圆形结构上，通过镫骨将声波传递到耳蜗中的淋巴液。耳蜗内壁包含超过 15000 根毛发，淋巴液的振动刺激毛发产生神经信号，并传入大脑进行语义层面的分析。

通常情况下，人类能够感知的频率范围为 20~20kHz，随着年龄递增，听觉会出现退化。人耳可以看作是一个频谱分析器，对不同频率的语音具有不同的感知特性，这种感知特性主要包括两个方面：响度和音高。响度是衡量声音强弱大小的一

个主观物理量，和声压具有较强的相关性，因此通常使用声压作为指标进行量化分析。然而响度和声压之间并非线性关系，其还和频率有很大关系。通常情况下，在2k~4kHz 范围内，产生相同的响度其需要的声压更低，表明人类对这个范围内的语音更加灵敏。

对音高的感知同样是非线性的，音高反应了语音中频率的高低，通过对音高的研究表明人类对音高的主观感知和频率呈对数关系，一般使用梅尔频率来表征频率的主观大小，放缩后的频率和音高是线性的。计算公式如下：

$$f_m = 2595log_{10}(1 + \frac{f}{700}) \tag{2-9}$$

其中f表示原始频率，f_m表示梅尔尺度频率。图 2-2 为梅尔频率-语音频率对应关系图。

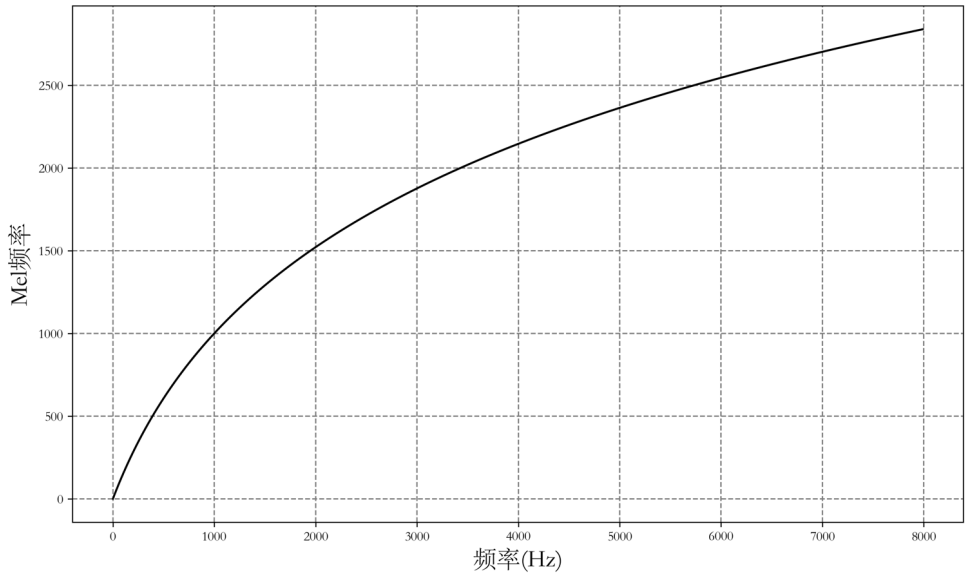


图 2-2 梅尔频率-信号频率映射

人耳还存在一种叫做掩蔽效应的特性，又分为频率掩蔽和时间掩蔽，频率掩蔽指的是当某一频率的声音声压超过临界值时，其附近频率的声音必须高于一定声压才能被感知到，否则将会被忽略。而时间掩蔽指的是某一声音结束后的一小段时间内其他声音会被直接屏蔽。这种掩蔽效应在语音编码中有着重要的应用。

2.1.3 FBank 特征提取

FBank（Filter Bank）是一种根据人耳听觉感知特性设计的特征，广泛用于语音识别、声纹识别和语种识别等领域。FBank 特征的提取流程主要包括以下几个阶段：预加重、分帧和加窗、短时傅里叶变换、梅尔滤波器组滤波和对数尺度转换,

处理流程如图 2-3 所示：

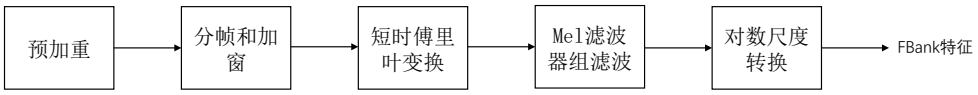


图 2-3 FBank 特征提取流程

（1）预加重

语音信号在传输过程中，高频部分的能量损耗相较于其他频率较为明显，通常使用一个高通滤波器来对高频部分进行补偿，这个过程也叫做预加重（Pre-emphasis），假设$x(n)$是语音信号，预加重后的信号可表示为：

$$y(n) = x(n) - \alpha x(n-1) \tag{2-10}$$

其中α是预加重系数，通常取 0.9。

（2）分帧和加窗

人在说话过程中，声学参数会随着内容变换，因此可以把语音信号看着是一个线性时变信号，为了方便分析，通常假定语音在 30ms 内的区间是平稳的，这就是语音的短时平稳性。离散傅里叶变换（Discrete Fourier Series, DCT）假设所截取的序列能够进行无混叠的周期延拓，然而实际的分帧不可能满足此条件，直接进行截取也就是加矩形窗会导致所谓的频谱泄露，即某些频率分量泄露到其他周期里。因此为了降低频谱泄露带来的影响，选择一个合适的窗函数是关键的一环。常见的窗函数有如下几种：

海明（Hamming）窗：

$$w(n) = \begin{cases} 0.54 - 0.46cos\dfrac{2\pi n}{N-1}, & 0 \le n \le N-1 \\ 0, & other \end{cases} \tag{2-11}$$

汉宁（Hanning）窗：

$$w(n) = \begin{cases} 0.5 - 0.5cos\dfrac{2\pi n}{N-1}, & 0 \le n \le N-1 \\ 0, & other \end{cases} \tag{2-12}$$

布莱克曼（Blackman）窗：

$$w(n) = \begin{cases} 0.45 - 0.5cos\dfrac{2\pi n}{N-1} + 0.08cos\dfrac{2\pi n}{N-1}, & 0 \le n \le N-1 \\ 0, & other \end{cases} \tag{2-13}$$

其中 N 表示窗口大小。衡量窗函数通常考察主瓣宽度、幅值误差、最高旁瓣等参数，主瓣主要影响信号的频率分辨率，带宽越宽分辨率越低。旁瓣影响信号频谱泄

露程度，旁瓣越低表示泄露越少。下表是几种窗函数的对比。

表 2-1 窗函数对比

窗类型	主瓣宽度	主瓣 3dB 带宽	幅值误差/dB	最高旁瓣/dB
矩形窗	1.0	0.89	-3.92(36.3%)	-13.3
海明窗	1.36	1.30	-1.78(20.6%)	-43.2
汉宁窗	1.50	1.44	-1.42(15.1%)	-31.5
布莱克曼窗	2.0	1.68	-1.10(12.5%)	-92.2

（3）短时傅里叶变换

语音信号在频域特征更加明显，通常会通过离散傅里叶变换将时域转化到频域进行分析。快速傅里叶变换（Fast Fourier Transform, FFT）是离散傅里叶变换的快速实现，因此在进行转化时候一般使用 FFT。为了凸显时域上的变化特性，本文选取 25ms 的窗口，以 10ms 作为帧移进行迭代的 FFT 方式将时域信号转换到频域，相比于直接对整个语音信号进行 FFT，其频谱的分辨率更高，这种方式也叫做短时傅里叶变换。其可表示为：

$$f_m[k] = \sum_{n=0}^{N-1} x[mT+n]e^{-j\frac{2\pi kn}{N}}, \quad k = 0,1,\cdots,N-1 \tag{2-14}$$

其中$f_m[k]$即为 FFT 后的结果，N表示窗口大小，$x[n]$表示时域离散信号，T为帧移长度，m表示第m个音频帧。对于语音增强任务，通常有两种方式来利用频域信息，第一种直接使用短时傅里叶变换后的复数谱信号作为特征；另一种方式是基于幅度谱特征，这种方式使用实部和虚部的模值进行建模，在逆傅里叶变换时利用原始带噪信号频域的夹角进行恢复。

（4）梅尔滤波器组

人耳听觉对音高具有非线性的感知特征，通常使用梅尔频率替代实际频率。为了消除语音信号中的谐波以便更好的突出共振峰特征，在提取到的频域特征基础上，使用三角带通滤波器组再进行一次变换，变换后的特征维度更低，有效的降低了计算量。通常梅尔特征维度为 80 维，而 FFT 后的频谱通常是 512 维。三角带通滤波器定义如下：

$$H_m(k) = \begin{cases} \dfrac{k-f(m-1)}{f(m)-f(m-1)}, & f(m-1) \le k < f(m) \\ \dfrac{f(m+1)-k}{(f(m+1)-f(m)}, & f(m) \le k \le f(m+1) \\ 0, & other \end{cases} \tag{2-15}$$

其中$f(m)$表示中心频点的频率，这个频率间隔由滤波器数量决定，通过在梅尔尺度上进行均匀划分得到。$f(m)$由式(2-16)给出：

$$f(m) = 700(10^{\frac{F_0 + m \cdot \frac{F_{max} - F_0}{M+1}}{2596}} - 1) \tag{2-16}$$

其中F_0表示最小梅尔频率，F_{max}表示最大梅尔频率，M 为滤波器个数。梅尔滤波器组如图 2-4 所示。最后取对数即得到了 FBank 特征。

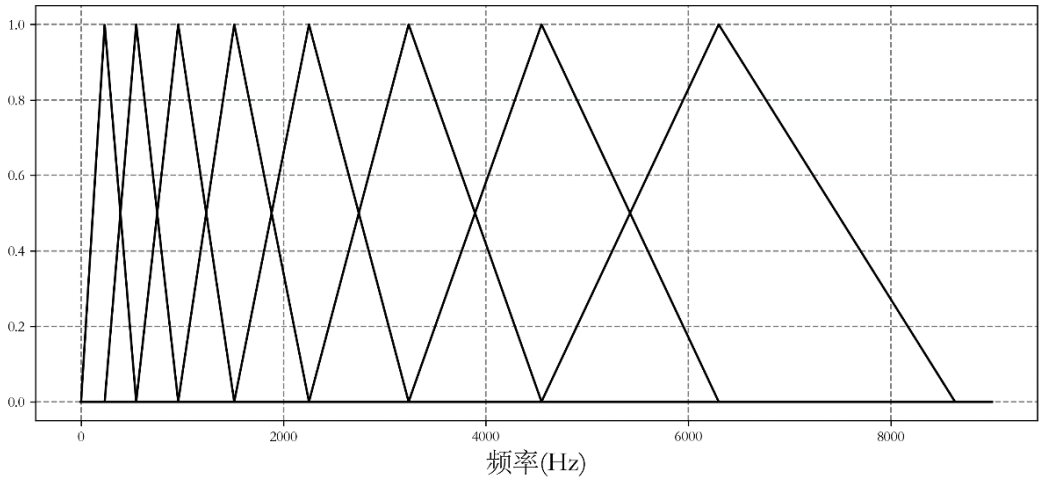


图 2-4 梅尔滤波器组

2.2 神经网络模型

人工神经网络（Artificial Neural Networks，ANN）是受到生物神经系统启发而构建的数学模型，用于模拟生物大脑的行为。对神经网络的研究最早可追溯到上个世纪 40 年代，早期的研究局限在单层感知机上，其难以对非线性问题建模。在 80 年代，LeCun 等人[78]首次提出了卷积神经网络并应用在手写字符识别中，揭开了神经网络研究的序幕，然而构建的模型参数量较大，其训练共花费了超过 3 天的时间，受限于计算机算力的限制，并没有成为研究的热门。随着 GPU（Graphics Processing Unit）算力的巨幅提升，2012 年，Hinton 团队在 ImageNet 图片分类比赛上取得了远超第二名的好成绩，其使用的卷积神经网络方法受到了学者们广泛的关注，标志着深度学习的研究正式进入爆发期。接下来将分点介绍常用的几种神经网络模型。

2.2.1 卷积神经网络

卷积神经网络（Convolution Neural Network, CNN）是一种基于卷积运算的模型结构，广泛用于计算机视觉、NLP 和语音领域。其核心思想是通过多个卷积核

在图像上进行卷积运算，从而得到多个特征图，进一步通过池化层、全连接层输出最终的期望结果。卷积核是一个矩阵结构的参数组，一张图像可以看作是一个二维数组，卷积核的大小通常为 3*3。在计算时，卷积核参数和图像上对应的点相乘并求和，根据步幅（stride）大小在图像上进行移动，从而完成一次卷积运算。卷积核数量和输出通道数有关，通常情况下，卷积核数等于输出通道数。而每个卷积核内包含多层，其层数和输入通道数相关，当 group 参数为 1 时，层数等于输入通道数。卷积运算如式(2-17)所示：

$$S(i,j)=\sum_{m}\sum_{n}I(i+m,j+n)\cdot K(m,n) \tag{2-17}$$

其中，$S(i,j)$表示输出的特征图上第(i,j)坐标的值，$I(i,j)$表示输入的第(i,j)个点，$K(m,n)$表示卷积核上第(m,n)个位置。池化层用于汇聚卷积层的输出结果，降低数据维度，有研究表明，池化还具有防止过拟合的作用。常用的池化方法包括最大池化（Max Pooling）、平均池化（Mean Pooling）和随机池化（Stochastic Pooling）等。最大池化就是选取滑动的矩形窗中最大的值作为输出，其计算过程和卷积类似，如图 2-5 所示：

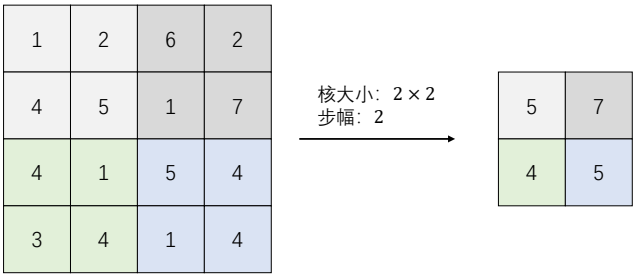


图 2-5 最大池化

全连接层是一个线性映射，输出数据的维度大小为分类类别。一个简单的 CNN 模型示意图如图 2-6 所示：

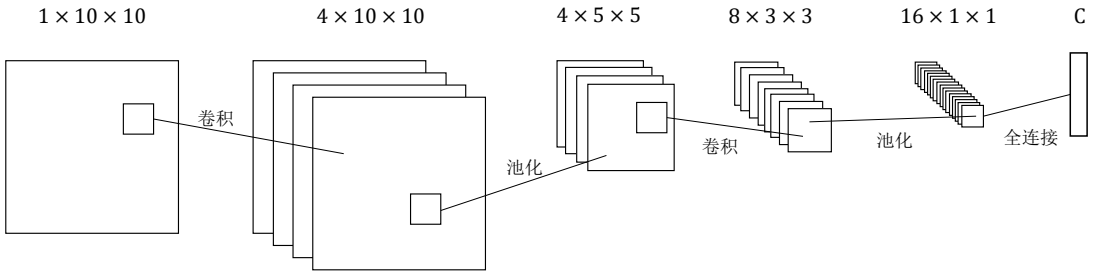


图 2-6 卷积神经网络模型

其中一张 10*10 大小的二维图片通过一层一层的卷积和池化计算最终输出维度为 C，其代表了分类类别。对于语音，其语谱图也可看作一张二维图片，因此可以很轻易的将卷积神经网络引入到对语音的建模中。

2.2.2 循环神经网络

循环神经网络（Recurrent Neural Network, RNN）是一种用于对序列建模的网络模型。CNN 由于其卷积运算的独特方式，善于捕获局部信息。而 RNN 则是利用以前帧的输出结合当前帧的信息迭代的进行计算，适合于捕获长距离的上下文信息，广泛用于语音识别、语音增强以及 NLP 领域中的机器翻译等任务中。一个简单的 RNN 模型可以通过如下方程进行描述：

$$h_t = g(Uh_{t-1} + Wx_t) \tag{2-18}$$

$$y_t = f(Vh_t) \tag{2-19}$$

其中h_t表示 t时刻隐藏状态，U为矩阵，V、W为 RNN 参数，x_t表示当前时刻输入，$f(\cdot)$表示激活函数。RNN 结构如图 2-7 所示。

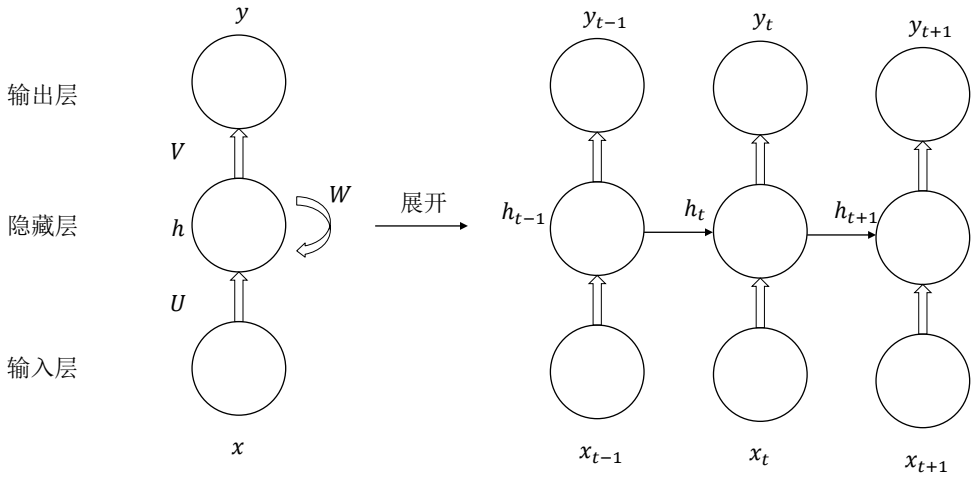


图 2-7 循环神经网络结构

由于 RNN 特殊的构造形式，常规的反向传播（Back Propagation, BP）算法无法进行梯度回传，一般使用 BPTT（Back Propagation Through Time）算法。RNN 虽然能够有效的对长序列问题进行建模，然而随着序列长度的递增，RNN 经过多次的递归计算容易导致梯度消失和梯度爆炸。有学者提出了 RNN 的变体 LSTM 和 GRU 来解决这些问题，相比于 RNN，其额外使用遗忘门来控制以前状态的信息进入下一个状态。

2.3 评价指标

通常情况下，为了得到一个公平的实验结论，有多种评估指标用于衡量模型性能的好坏。对于语音增强任务，评价指标分为主观方法和客观方法，主观方法是通过人的听觉感受作为判别标准，通常以平均意见分数（Mean Opinion Score, MOS）作为主观评价方法，在语音合成领域广泛使用，客观方法是通过对语音信号进行量化计算得到，包括感知语音质量评估（Perceptual Evaluation of Speech Quality, PESQ）和短时客观可懂度（Short-Time Objective Intelligibility, STOI）等。对于语种识别任务，其本质上是一种分类任务，通常使用等错误率（Equal Error Rate, EER）、准确率和平均检测代价C_{avg}作为评价指标。接下来将分点进行概述。

2.3.1 短时客观可懂度 STOI

短时客观可懂度是语音增强中常用的评估指标，其用于衡量人类对语音内容的可懂度，取值为 0 到 1 之间，值越高表明语音的质量越好。具体的，STOI 指标通过如下步骤计算得到：（1）声音活性检测，用于去除能量较低的无意义帧，（2）短时傅里叶变换，将语音信号转化到频域，（3）三分之一倍频分析,用于对时频点进行划分，（4）归一化，（5）相关系数计算，计算干净语音谱和增强语音谱之间的相关系数，最后取均值得到 STOI 指标。

2.3.2 感知语音质量评估 PESQ

感知语音质量评估是另一个常用的语音质量评估指标，由国际电信联盟(International Telecommunication Union，ITU)提出，其和 MOS 值之间有很强的相关性，因此从某种意义上反应了语音的感知质量。其值在-0.5 到 4.5 之间，值越高越好。PESQ 的计算较为复杂，包含预处理、时域对齐、滤波等。PESQ 应用广泛，在电话通信质量评估中也是常用的指标之一。

2.3.3 等错误率 EER

等错误率是分类问题中的常用指标，是通过调节阈值使得错误拒绝率(False Rejection Rate，FRR)和错误接受率(False Acceptance Rate，FAR)相等时的值，其表现为在 ROC 曲线上与斜率为-1 且过（0，1）点的斜线上的交点。EER 越小表明算法识别率越高。

2.3.4 平均检测代价 Cavg

平均检测代价是 NIST 提出的语种测试标准，其在计算时对每个语种计算的代

价取均值，因此在样本不平衡的测试集中其更有说服力。计算方法如式(2-20)所示。其中，N_L表示将要建模的语种数量，L_T和L_N各表示目标语种和非目标语种，P_{Target}表示目标语种的先验概率，P_{NT}表示非目标语种的先验概率，通常取 0.5，C_{Miss}和C_{FA}分别表示漏判和误判的代价值，通常为 1，$P_{Miss}(L_T)$表示预测L_T语种错误的概率，$P_{FA}(L_T, L_N)$表示L_T预测为L_N语种的概率，P_{OT}表示预测为域外未知语种的概率，通常数据集中不包含域外语种，因此该项通常为 0。

$$C_{avg} = \frac{1}{N_L}\sum_{L_T}\Big\{C_{Miss}P_{Target}P_{Miss}(L_T) + \sum_{L_N} C_{FA}P_{NT}P_{FA}(L_T, L_N) + C_{FA}P_{OT}P_{FA}(L_T, L_O)\Big\} \tag{2-20}$$

2.4 本章小结

本章主要对语音增强和语种识别相关技术的背景与算法进行介绍。首先，介绍了人类听觉感知原理，接下来结合听觉原理详细描述了本文用到的 FBank 特征提取方法，然后再对深度学习的基础模型进行了简要的概括，最后对本文中将要使用的评估指标做了说明。

第三章 基于动态选择机制和中继监督优化的语音增强

本章内容结构安排如下：首先详细的剖析了当前主流端到端语音增强方案中存在的缺陷，并给出了相应的解决方案（见 3.1 节）。接下来详细的介绍了语音增强网络的基本构成，同时将动态选择机制以及中继损失引入到增强模型中（见 3.2 节）。最后在公开语音数据集上对所提出的模型进行测试，通过对评测结果进行深入分析与探讨，证明了本章所提语音增强方案的有效性（见 3.3 节）。

3.1 问题描述

（1）固定大小卷积核的局限性

通常情况下，从时域信号中很难挖掘出比较明显的音频特征，往往通过短时傅里叶变换将音频信号从时域变换到频域，从而获取到语音更加清晰的层次结构，然后再对频谱图进行建模。研究表明，卷积神经网络（Convolutional Neural Networks, CNN）非常适合捕获语音信号中的特定结构[24]，因此基于 CNN 的端到端模型在语音增强领域具有广泛应用。由于语音信号受到说话人语速、说话人年龄、说话人性别、环境噪声种类以及语音内容等多个条件影响，频谱图变化波动较大。在基于 CNN 的模型结构中，卷积核被设定为一个恒定参数，会导致模型在推理时其每一层的感受野大小固定，固定感受野大小的网络很难在所有条件下的频谱图上都表现良好。因此采用固定的卷积核大小在一定程度上限制了模型的性能。近年来，有学者提出了 SKconv (Selective Kernel Convolution)网络，它使用多个具有不同感受野大小的卷积层，并根据所有获得的信息为每一个分支分配不同的权重，从而让模型自适应的调整感受野大小。其实验表明这种动态选择机制能够提高语音增强的模型的性能。

（2）训练目标与模型优化方向不匹配

损失函数的构建是语音增强算法在训练过程中的关键一环。作为回归任务的语音增强通常采用均方误差（Mean Squared Error，MSE）作为优化目标，以模型的输出值和干净语音谱作为函数的输入计算得到损失值，并通过随机梯度下降进行模型优化。然而这种建模方式存在着以下问题：一方面它仅专注于模型最后一层输出的干净语音，却忽略了模型中间层的作用，模型中间层可能朝着未知方向优化，不同层的卷积核很难有效的提取到干净语音和噪声特征。另一方面，由于优化目标只是干净语音的一一映射，噪声特征可能被完全忽略，导致模型难以在复杂的噪声环境下进行更好的预测。如果希望模型能够更好的预测干净语音谱，模型的预测就

不应该只局限在最后一层。受文献[79]的启发，本文用干净语音谱和噪声谱近似每个模块的输出，并提出了一种多损失机制，包括最终的映射损失和用于特征校正的中间损失，从而迫使卷积层提取干净语音和噪声分量特征。

3.2 算法描述

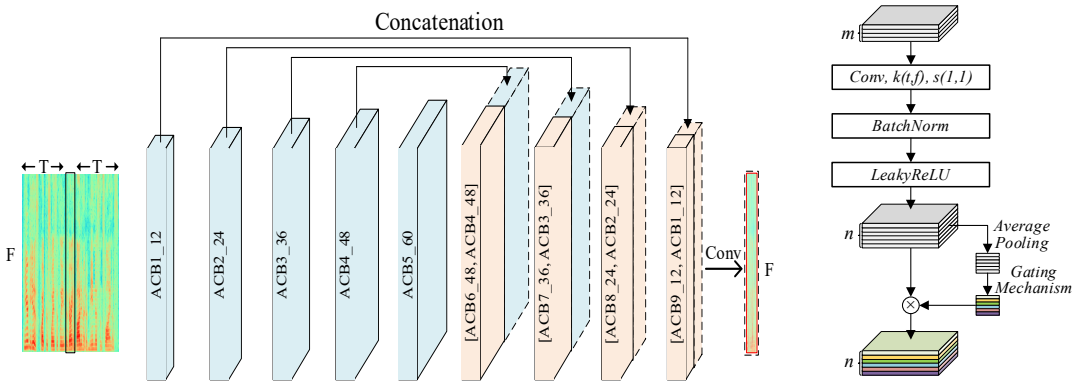
本文的算法是在 RCNA 模型的基础上作出的改进与优化：（1）使用动态选择核机制使得模型能够自适应的调整感受野大小，弥补了传统卷积网络在语音增强建模过程中难以适应各种噪声的缺陷。（2）引入了中继监督优化，从而有导向性的引导模型的每一层学习干净语音和噪声的特征，提升了模型对带噪语音到干净语音的映射能力，得到清晰度和可懂度更优的增强语音。

3.2.1 基于 RCNA 的端到端语音增强

RCNA 作为本文算法的基本架构如图 3-1 所示，是一个基于编码器-解码器的网络，由多个 ACB 模块组成。所有模块都包含一个卷积层，然后是一个批量归一化 (Batch Normalization, BN) 层和一个 Leaky ReLU（Leaky Rectified Linear Unit）激活单元层。Leaky ReLU 的数学表达式如式(3-1)所示：

$$y_i = \begin{cases} x_i & x_i \geq 0 \\ \dfrac{x_i}{A} & x_i < 0 \end{cases} \tag{3-1}$$

其中 A 是一常数，x_i表示第i帧输入，相比于 ReLU，其保留了负数部分的信息。在卷积层之后使用通道注意力机制 SE（Squeeze-and-Excitation），其通过对特征图赋予不同的权重来对通道之间的相互依赖性进行建模，提升了模型对不同特征重要性的区分能力。每个 ACB 模块示意图如图 3-1(b)所示。其中 T 表示在目标帧的每一侧拼接的帧数。F 是每帧的频点数。图 3-1(a)中每个模块对应一个 ACB 块，$ACB\#i_m_n$中的参数i表示第i层 ACB 块，m表示输入通道个数，n 表示输出通道个数。相比于 ResNet，由于语音增强任务是基于序列到序列的建模，RCNA 丢弃最大池化层和相应的下采样层。因此，这些模块中每一层的特征图大小保持一致。最后一个信息聚合块只包含一个卷积层来聚合前面操作中获得的信息，同时降低通道数和语音谱保持一致。模型的前半部分可以当作编码器，其由卷积核数量逐渐递增的卷积层构成，卷积层输出的每个通道可以看作一种特征类型。由于在逐渐迭代的过程中，可能导致部分信息丢失，这里借鉴了 UNet 网络结构，将编码器的前几层输出附加到解码器的每一层输入上，从而有效地补偿了底层特征丢失带来的损失，模型具体参数如表 3-1 所示。



(a) RCNA 端到端语音增强模型　　(b) RCNA 模块

图 3-1 RCNA 语音增强模型

表 3-1 RCNA 模型参数表

模块名	输入通道	输出通道	卷积核大小	填充
ACB#1_1_12	1	12	(11, 9)	(5, 4)
ACB#2_12_24	12	24	(11, 9)	(5, 4)
ACB#3_24_36	24	36	(11, 9)	(5, 4)
ACB#4_36_48	36	48	(11, 9)	(5, 4)
ACB#5_48_60	48	60	(11, 9)	(5, 4)
ACB#6_60_48	60	48	(11, 9)	(5, 4)
ACB#7_96_36	96	36	(11, 9)	(5, 4)
ACB#8_72_24	72	24	(11, 9)	(5, 4)
ACB#9_48_12	48	12	(11, 9)	(5, 4)
1x1Conv	24	1	(1, 1)	(0, 0)

在 RCNA 中除最后一层之外，其余每个模块结构是类似的，都是由卷积层、批量归一化（BN）、LReLU 和 SE 层组成，编解码器职责不同，编码器负责语音信息的抽象与特征抓取，提取高质量的信息供解码器选择，而解码器负责干净语音信号的还原。为了能够获得具有更好表征的语音特征，本文在编码器中用 SKconv[80] 替换了传统的卷积层，其通过动态的调节卷积核大小，提升了模型适应不同噪声类型和不同的信噪比的能力。由于原始的 SKconv 的输入和输出通道数相同，因此在模块最后添加一个卷积核大小为 1×1 的卷积层对输出通道进行更改，以满足通道数逐渐递增的模型架构。最终，编码器中的每个块都由一个 SKconv、BN、LReLU

和一个 1×1 卷积层组成。

语音幅度谱在编码器逐层的迭代中逐渐由低维映射到更高的维度，而噪声成分也逐渐降低，解码器的功能用于将编码器提取的高维特征逐渐恢复为干净语音谱，为了减少高维特征提取过程中带来的信息损失，通过跳连接将编码器中间输出连接到解码器输入上。解码器中的每个模块由卷积层、BN、LReLU 和 SE 层组成。为了帮助 SE 层实现更好的特征选择，本文在解码器的每一层中对 SE 的输出添加中继监督损失，以改进 SE 在特征选择时的结果。本文所提出模型的体系结构如图 3-2 所示。

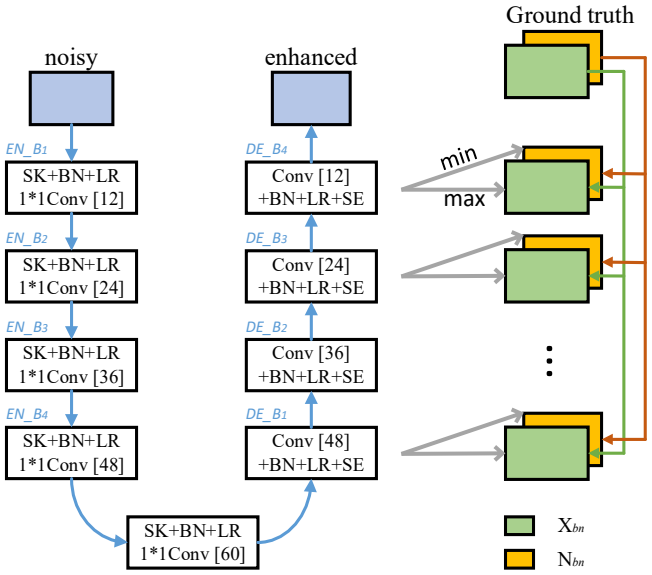


图 3-2 改进的 RCNA 端到端语音增强模型

3.2.2 动态选择卷积核

SKconv 根据输入自适应调整感受野大小，即卷积层可以在多个不同大小的卷积核之间进行选择，使得模型能够更好的适应不同的噪声类型。其结构如图 3-4 所示。它分三个步骤实现：拆分、融合和选择。

1）拆分

拆分操作会生成多个具有不同感受野的神经元分支。设$\mathrm{U} \in \mathbb{R}^{H\times W\times C}$为 SK 层的输入，其中$H, W,$和$C$分别表示输入特征图的高度、宽度和通道数。然后使用 4 个具有不同扩张率（1，2，4 和 6）的扩张卷积[81]（Dilated Convolution）来控制感受野大小并捕获不同尺度的上下文信息，其表示为$\mathrm{Conv}i(i=1,2,3,4)$，卷积核大小为$3\times 3$。SK 层的输出可以表示为$\mathrm{RF}_i \in \mathbb{R}^{H\times W\times C}(i=1,2,3,4)$。在最原始的 SK 网络中，$\mathrm{Conv}i$包含分组卷积、BN 和 ReLU，在本文中，为了降低模型参数量，将分组卷积改为了扩张卷积。扩张卷积的计算如式(3-2)所示：

$$(x * \omega)(i) = \sum_{k=1}^{K} x[i + lk]\omega[k] \tag{3-2}$$

其中$x[i]$表示输入特征，ω[k]表示卷积核，$*$表示卷积运算，l表示扩张率。通过控制扩张率大小可以有效的控制卷积核计算的感受野大小。

2）融合

融合操作聚合了多个分支获得的特征以进行权重选择。为了实现这一目标，首先将所有RF_i分支进行求和：

$$\mathrm{RF}_{global} = \sum_{i}^{4} \mathrm{RF}_i \tag{3-3}$$

然后再利用全局平均池化生成通道统计量$\mathrm{s} \in \mathbb{R}^C$，其中的每个分量代表了相对应通道的信息，由式(3-4)给出：

$$s_k = \frac{1}{H \times W} \sum_i^H \sum_j^W \mathrm{RF}_{global_k}(i,j) \;\; k = 1,2,\dots,C \tag{3-4}$$

其中$\mathrm{RF}_{global_k}(i,j)$表示第$k$个通道上点$(i,j)$的值。上一步得到的统计信息向量$s_k$包含了所有通道的信息，接下来通过一个全连接层基于$s_k$学习计算不同分支的关联关系，也就是分支权重，$s_k$向量的大小降低到 4，提高了计算效率。通过全连接层后得到了一个紧凑的特征$\mathrm{z} \in \mathbb{R}^d$，如式(3-5)所示。其中$\mathcal{F}_{fc}(\cdot)$表示在维度$\mathbb{R}^{d \times C}$上的全连接操作。

$$\mathrm{z} = \mathcal{F}_{fc}(s) \tag{3-5}$$

3）选择

选择操作将会根据为每个分支赋予的不同权重重新计算特征图。在这一步中，共创建了 4 个不同的全连接层$\mathrm{A, B, C}$和$\mathrm{D} \in \mathbb{R}^{C \times d}$，上一步骤获取到的特征 z 并行的通过这些全连接层，然后使用一层 softmax 得到每一个分支的权重值，softmax 计算如式(3-6)所示：

$$a_k = \frac{e^{A_k z}}{e^{A_k z} + e^{B_k z} + e^{C_k z} + e^{D_k z}} \tag{3-6}$$

其中$\mathrm{a, b, c}$ 和 d分别表示$\mathrm{U}_i(i = 1,2,3,4)$的软注意力向量。下标 k 表示对应第 k 个通道。b_k, c_k, d_k依次类推。并且$a_k + b_k + c_k + d_k = 1$。最终得到的特征V便包含了多个层级的感受野信息，模型从而可以通过控制权重值大小选择合适的卷积核权重，聚合多个感受野尺度的信息。

$$\mathrm{V}_c = a_k \times \mathrm{RF}_1 + b_k \times \mathrm{RF}_2 + c_k \times \mathrm{RF}_3 + d_k \times \mathrm{RF}_4 \tag{3-7}$$

其中$\mathrm{V}=[\mathrm{V}_1,\mathrm{V}_2,\ldots,\mathrm{V}_c]$，$\mathrm{V}_c\in\mathbb{R}^{H\times W}$。最终 SKconv 模块示意图如图 3-3 所示。

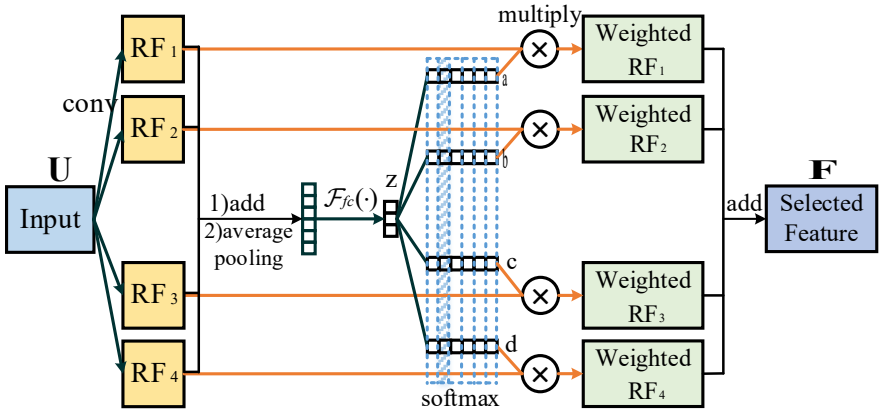


图 3-3 SKconv 结构示意图

3.2.3 中继监督损失函数

在 RCNA 模型中，使用了一种 SE 机制用于自适应的调节通道的权重，使得模型能够利用全局信息对通道局部信息进行重新校正。在SE机制中，包含两个部分，压缩（Squeeze）和激励（Excitation）。压缩操作相当于在通道维度上的一个全局平均池化，能够获取到输入特征图在通道维度上的统计特征z_k，计算公式如式(3-8)所示：

$$z_k=\frac{1}{H\times W}\sum_i^H\sum_j^W v_k(i,j)\ \ k=1,2,\ldots,C \tag{3-8}$$

其中，$v_k(i,j)$表示第k个特征图上坐标(i,j)上的值。由于得到的特征z_k只能表明特征图在通道维度上的分布特性，无法得到通道之间的关系，因此还需要通过激励操作对刚刚得到的统计特征进行非线性映射。具体来说，首先通过一层线性映射层W_1，将z_k的维度放缩，从而降低计算量，然后再通过 ReLU 激活单元，将值限制在区间[0,1]内。为了恢复z_k的维度，再通过一层线性映射层W_2，最后经过 sigmoid 函数，得到最终的通道权重s。其计算公式如式(3-9)所示：

$$s=\sigma(W_2\delta(W_1z)) \tag{3-9}$$

通过 SE 机制，使得模型具备为特征图中的噪声成分和纯净语音成分赋予不同的权重的能力，从而提高语音增强的性能。然而，在端到端的训练过程中，模型的中间输出是一个黑盒，这可能导致模型为某些语音成分分配低权重而为某些噪声成分分配高权重。为了解决这个问题，本文引入了一种新的损失机制，叫做中继监督损失[79]，为了帮助卷积核更加精确的提取到干净语音和噪声的信息，在 SE 层获取到的特征图权重中，选取权重最大的特征图近似为干净语音谱，而权重最小的特

征图近似为噪声谱，通过最小化近似的语音谱和真实语音谱之间的距离，从而有导向的使得 SE 层输出的权重更加准确的朝着目标进行，loss 计算公式如(3-10)所示：

$$L_{sv}(DE\_B_i) = \|\mathrm{X}_{bn} - \mathrm{SE}_{max}(DE\_B_i)\|_{1,1} + \|\mathrm{N}_{bn} - \mathrm{SE}_{min}(DE\_B_i)\|_{1,1} \tag{3-10}$$

其中，DE_B_i表示第i个解码层，X_{bn}和N_{bn}分别表示归一化的真实干净语音谱和真实噪声谱，这里归一化的目的在于 SE 层的特征图也是通过归一化处理，为了保持计算损失时单位的一致性。L_{sv}即为中继监督损失。由于有 4 层解码层，越靠近模型尾部的模块得到的特征图会更加清晰，因此其重要性不同，在本文中，为每个模块设置了一个权重值，最终的中继监督损失计算公式如(3-11)所示：

$$L_{sv} = \sum_{i}^{4} \alpha_i \cdot L_{sv}(DE\_B_i) \tag{3-11}$$

在这里$\alpha_1, \alpha_2, \alpha_3$和$\alpha_4$是模型的超参数，分别选取 0.25，0.5，0.75 和 1.0 作为它的值，最后总的损失函数如式(3-12)所示，它由两部分组成：预测损失和中继监督损失。

$$L_{total} = \alpha L_{pre} + \beta L_{sv} \tag{3-12}$$

其中L_{total}表示总的损失函数，L_{pre}表示预测损失，这里使用L_1损失函数，如式(3-13)所示：

$$L_{pre} = \left\|\mathrm{X} - \hat{\mathrm{X}}\right\|_{1,1} \tag{3-13}$$

其中X 和$\hat{\mathrm{X}}$分别表示干净语音谱和预测谱。

3.3 实验与讨论

3.3.1 实验环境

本章实验代码基于 PyTorch 深度学习框架编写，编程语言为 Python。因为在模型的训练过程中，存在大量的相似重复运算，例如基本的线性代数运算矩阵乘法和加法等，比起 CPU，这些计算更加适合并行度较高的 GPU 进行处理，而 PyTorch 框架便能较为简便的将计算放在 GPU 中进行处理，从而降低了训练的时间成本。并且 PyTorch 中包含大量现成的神经网络库，包括基础模型、训练优化器以及其他辅助工具等，可以帮助共快速实现复杂的模型结构。

实验开发的操作系统是 Ubuntu18.04，基于 Linux 的系统在开发时具有更好的兼容性和稳定性。开发 IDE 使用 PyCharm 和 VSCode。CPU 型号为 i7-9700，GPU 使用具有 12GB 的显存的 NVIDIA GeForce GTX 1080ti，硬盘大小为 2TB，内存大

小为 32GB。

3.3.2 实验数据及设置

本章实验所用到的干净语音数据集为 TIMIT，其由美国国防部资助，主要在德州仪器（TI）和麻省理工学院（MIT）的协作下构建，因此命名为 TI-MIT。TIMIT 数据集被广泛用于语音增强和语音识别领域中，其总共包含 6300 条语音，采样率为 16K。说话人来自于美国 8 个具有不同方言的地区，一共 630 人，每人录制 10 个句子。其中女性 30%。男性占比 70%。TIMIT 对数据集进行了划分，共分为三个：训练集 3696 条语音共 3.14 小时、核心测试集 192 条共 0.16 小时和完全测试集 1344 条共 0.81 小时，在划分时保证测试集中的说话人不会出现在训练集中，同时测试集覆盖所有的音素。本章中选择其核心测试集作为最终的测试集，训练集保持不变。

噪声数据集使用 Noisex92[82]，其由英国荷兰感知 TNO 研究所语音研究单位(SRU)在 1990 年到各个噪声源现场录制，广泛用于语音增强的研究。本章中使用其中的餐厅嘈杂噪声（babble）、驱逐舰引擎噪声（destroyer engine）、驱逐舰操作舱噪声（destroyer-ops）和工厂车间噪声（Factory1）作为训练集混合噪声源，以-5、0 和 5dB 的信噪比混合在训练集中。对于测试集，选择非平稳的工厂车间噪声（factory2）和平稳白噪声（white）作为噪声源，并额外选择-10dB 和 10dB 两个训练集外的噪声信噪比添加到测试集中。

对于特征提取，本文使用具有 256 个点的汉宁窗提取时频特征，帧移为 128 个点，使用短时傅里叶变换（Short-Time Fourier Transform，STFT），STFT 变换长度为 256，因为变换后的特征是对称的，因此取变换后一半的特征向量，最后特征向量维度为 129。在模型训练方面，本文使用 Adam 优化器[83]，批大小为 8，学习率为 0.0002。当模型 loss 不再明显下降时候，采取将学习率缩小到原来的 1/10 的策略进行训练。

3.3.3 对比方法及评价指标

本文实验对比的模型如下：

（1）RCNA[84]：RCNA 由九个构建块组成。前八个块包含一个卷积层、LReLU、BN 和 SE 层，最后一个块包含一个卷积层。RCNA-SK：在 RCNA 的编码器中去除 SE 并用 SKconv 替换传统的卷积层。

（2）RCNA-loss：RCNA 在解码器中使用所提出的损失机制。

（3）RTNet-3[85]：一种基于编码器-解码器的最新模型，其使用了一种动态注

意力机制和递归学习。在模型的中间阶段，通过噪声估计模块生成注意力，具有较好的噪声抑制性能。在各项指标上都显著的超过了其对比模型。

3.3.4 实验结果与分析

在本文中，选择 PESQ 和 STOI 作为语音质量的评估指标。首先本文测试了在 5 种信噪比条件下（-10，-5，0，5 和 10dB），模型对于可见噪声的性能。实验结果如表 3-2 所示。为了验证模型在未知环境下的泛化性，本文也测试了模型在不可见噪声下的性能指标，如表 3-3 所示。

表 3-2 噪声匹配条件下的平均 STOI 和 PESQ

指标	模型	Babble	Destroyer engine	Destroyer-ops	Factory1
STOI (%s)	Noisy	65.51	67.98	68.82	65.14
	RCNA	73.46	81.74	79.68	76.28
	RCNA-loss	74.21	82.90	81.00	76.96
	RCNA-SK	74.42	83.84	81.36	77.61
	Proposed	75.20	**83.90**	81.47	77.83
	RTNet-3	**75.90**	83.59	**81.72**	**78.57**
PESQ	Noisy	1.89	1.84	1.91	1.81
	RCNA	2.28	2.51	2.50	2.37
	RCNA-loss	2.35	2.64	2.61	2.46
	RCNA-SK	2.35	2.67	2.62	2.47
	Proposed	**2.39**	**2.68**	**2.65**	**2.50**
	RTNet-3	2.29	2.53	2.52	2.38

如表 3-2 所示，通过对 RCNA 和 RCNA-loss 模型的对比，本文所提出的 SK 机制在各种噪声条件下 STOI 和 PESQ 指标均优于基线模型，表明将 SK 引入卷积网络中有效的提高了语音增强的性能。通过将中继监督引入到 RCNA 模型中，即表 3-2 中的 RCNA-SK 模型，增强语音的各项指标也同样优于基线模型，证明了中继监督机制的有效性。将本文所提出的两种方法均应用在模型中，性能有进一步提升，STOI 和 PESQ 都大幅优于原始的 RCNA 基线模型，证明本文所提出的方法能够有效的提升带噪语音在各种环境下的语音感知质量和可懂度。相较于 RTNet-3，本文所提出的模型在 STOI 指标上给出了相当或稍差的结果，然而在 PESQ 指标上，模型仍具有显著的优势。

为了测试模型在未知环境下的泛化性，本文选取了训练集中未见过的噪声来进行测试，如表 3-3 所示，在不可见噪声 Factory2 和 White 下，所提出的方法相较

于基线模型均有较大的性能提升，STOI 分别从 82.58%和 79.31%提升到 84.12%和 80.02%，PESQ 从 2.59 和 2.44 提升到 2.73 和 2.56，并且 RCNA-loss 和 RCNA-SK 在 STOI 和 PESQ 指标上也均优于 RCNA。该实验表明，本模型在噪声不匹配的条件下仍具有较好的性能。相较于 RTNet-3，通过对 STOI 和 PESQ 指标的综合比较，所提出的模型在未知环境下更具有优势，其中在 Factory2 噪声下，本文提出的模型 STOI 和 PESQ 分别为 84.12%和 2.73，高于 RTNet-3 的 84.10%和 2.59，在白噪声下，虽然 STOI 低于 RTNet-3 模型，但是在 PESQ 指标上仍具有显著的优势。通过表 3-2 和表 3-3 的实验，一定程度上证明了本文提出的 SK 机制和中继监督能够适应不同的噪声，在各种噪声环境下，能够有效的改善语音的质量和可懂度。

表 3-3 噪声不匹配条件下的平均 STOI 和 PESQ

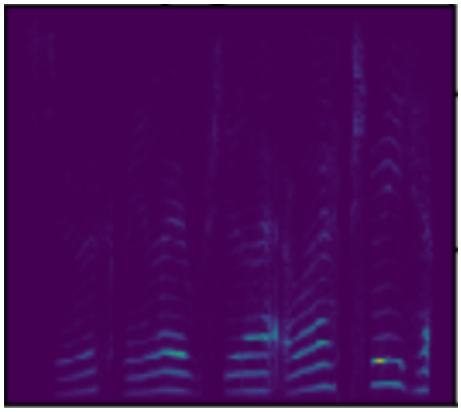
模型	STOI(%)		PESQ	
	Factory2	White	Factory2	White
Noisy	73.52	71.14	2.05	1.74
RCNA	82.58	79.31	2.59	2.44
RCNA-loss	83.27	79.56	2.68	2.55
RCNA-SK	83.80	79.91	2.72	2.53
Proposed	**84.12**	80.02	**2.73**	**2.56**
RTNet-3	84.10	**80.94**	2.59	2.40

为了进一步验证所提出的损失函数是否对语音增强任务起作用，本文对 RCNA 和 RCNA-loss 的解码器做了可视化分析，其中取解码层第一层（DE_B_1）和第二层（DE_B_2）。本文随机选择验证集中的一条带噪语音，将其输入模型中，同时对解码器第一和第二个模块中具有最大注意力权重的特征图进行了绘制。其特征图如图 3-4 所示。

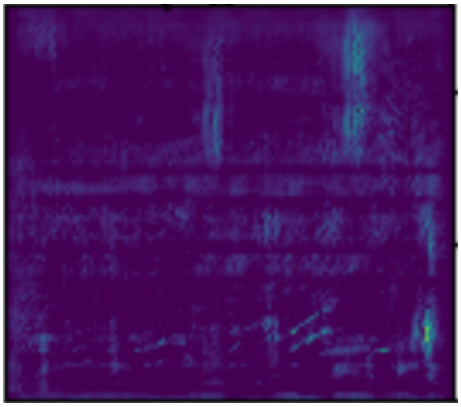
可以观察到到(b)中的特征图相当混乱，而(c)与(a)相似并且出现了谐波图案。尽管在(d)中可以看到语音特征纹理，但整体模式仍然存在一些噪声，并且(e)包含清晰可见的谐波结构。这表明本文提出的的监督损失机制可以有效地引导卷积核更加专注的提取到干净语音特征，从而提高最后增强语音的清晰度。此外，通过比较DE_B_1和DE_B_2的特征图，即(b)和(d)、(c)和(e)证明了 SE 机制在语音增强模型中的原理，即通过为干净语音特征更为清晰的特征图赋予更大的权重，而噪声特征赋予更低的权重，无论是否使用中继监督，具有最大权重的通道所指向的特征图结构和真实语谱图相似。

为了进一步了解 SKconv 是否起作用以及其作用原理，本文对注意力权重进行了数值分析。测试语音为 3 条与 factory1 噪声混合的嘈杂语音，其信噪比分别为-

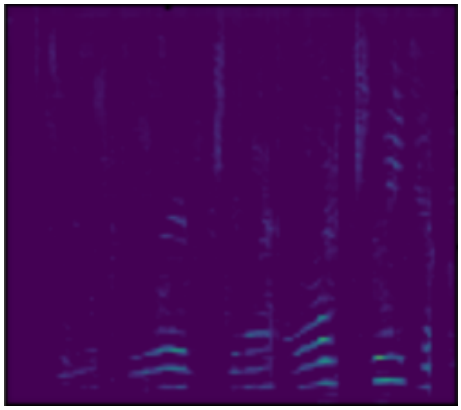
5、0 和 5dB。首先，从DE_B_4中随机选择一个通道，提取模型赋予 4 个具有不同的卷积核大小的分支的权重值，如图 3-5 所示。发现通道分配给每个分支的权重是不同的。例如，分配给RF_4的权重约为 0.4，而分配给RF_1的权重约为 0.1。并且在每个 SKConv 分支中，虽然分配给所有 SNR 的权重略有不同，但它们都固定在一定范围内，证明了 SK 机制对于噪声信噪比的选择作用，使得模型可以根据噪声类型自适应的调整卷积核。



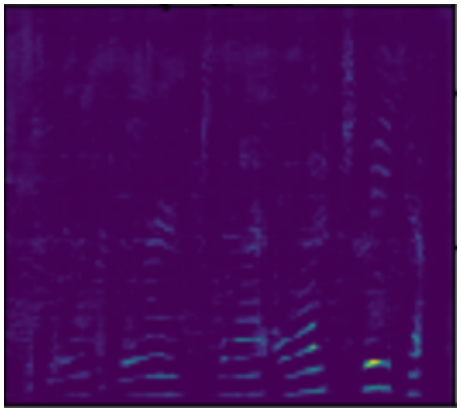
(a) X_{bn}



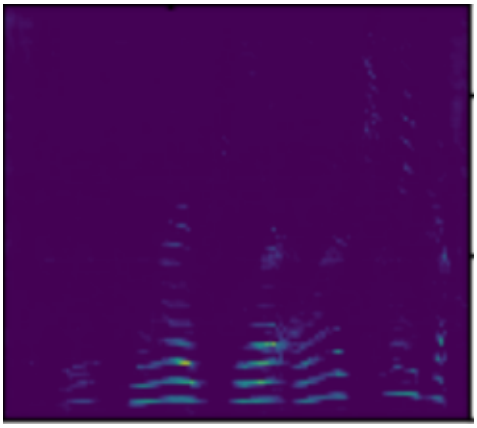
(b) DE_B_1 (RCNA)



(c) DE_B_1 (RCNA-loss)



(d) DE_B_2 (RCNA)



(e) DE_B_2 (RCNA-loss)

图 3-4 RCNA 和 RCNA-loss 模型中最大注意力权重的特征图

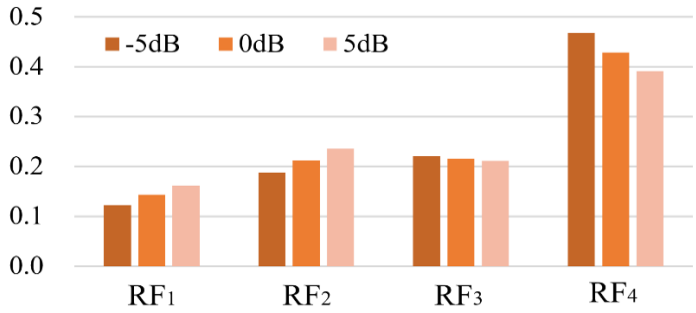


图 3-5 在 factory1 噪声下 SKConv 每个分支的平均权重

其次，本文还对 SKConv 对噪声类型的适应进行了验证。测试数据使用 4 种不同的噪声，分别为 babble、destroyer engine、factory1 和 destroyer-ops。本文计算了不同噪声类型的语音在不同 SKConv 分支上对每个通道的平均权重值如图 3-6 所示。可以清楚地看到每个分支的权重值存在明显差异。例如，在处理 babble 和 destroyer engine 噪声时，模型分别为RF_4分配 0.3 和 0.5 的权重，而RF_3的权重分别为 0.15 和 0.3。这表明在不同的噪声条件下需要不同的感受野。因此，自适应调整卷积核大小对于语音特征提取是必要的。这种机制对语音增强任务有效的原因很可能是动态改变感受野可以帮助提取具有不同轮廓的声学特征，从而更为精确的将带噪语音映射到纯净语音上。

表 3-4 各个模型参数量对比

模型	RCNA	Proposed	RTNet-3
参数量（百万）	1.17	**1.04**	3.98

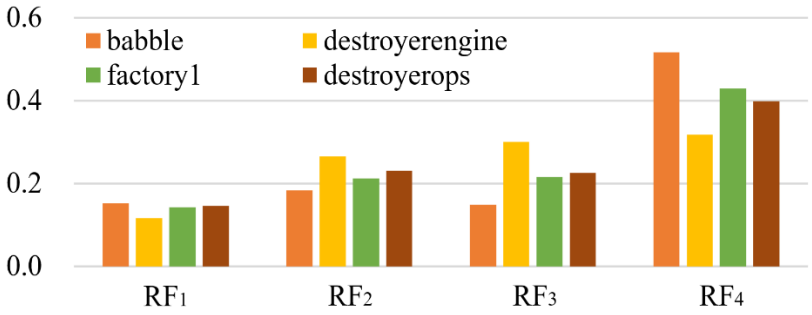


图 3-6 在不同噪声类型下 SKConv 每个分支的平均权重

本文所提出的模型参数与基线模型之间的比较如表 3-4 所示。因为原始的 RTNet 基于时域建模，而本文的方法基于频域，为了公平比较，在 RTNet-3 具体实现时，对其进行了改动，使得特征提取方法相同。可以看到本文提出的方法的参数量更小，同时，实现了更好的性能。表明所提出的模型性能提升并非来源于参数量

与计算复杂度的增加。与 RCNA 基线模型相比，SK 机制并没有增加计算量，反而有更好的效果。

3.4 本章小结

本章探索了一种基于编码器-解码器的语音增强模型，首先对卷积层进行改进，在编码器中使用具有动态感受野大小的 SKConv 进行有效的特征提取，并在解码器中使用中继监督损失，模型以多任务的方式进行学习，从而实现更好的特征选择。实验结果表明，所提出的方案优于基线模型。对于不同类型噪声和不同信噪比噪声，具有较好自适应性。

第四章 基于词法和语法特征的语种识别

本章的内容结构安排如下：首先对当前主流语种识别方案进行了分析，指出了存在的问题以及缺陷。然后介绍了本文提出的算法架构。在本章中，提出了一种多语种语音识别框架，基于 WavLM 和 Conformer 分别探索了利用无监督表征学习特征进行建模和端到端的底层声学特征建模的方案。在这个框架的基础上，基于语音识别的后验概率，设计了一种鲁棒的语种识别方法。考虑到利用语言学信息，本文同时将语言模型引入到语种识别中，进一步提升了识别精度。最后结合第三章提出语音增强方案，提升了语音在噪声环境下的语种识别准确率。

4.1 问题描述

(1)语种识别环境泛化性问题

目前，基于深度学习的语种识别效果显著，但仍存在泛化性问题，即如果测试条件和训练条件不匹配，例如说话环境，说话人不同和录制设备的不同都会对识别性能产生较大影响，导致语种识别难以应用于实际的下游任务中。在噪声种类复杂多变、录制设备多种多样、说话人未知的复杂环境下，如何使语种识别在面对复杂环境时保持稳定工作是一个值得研究的问题。考虑到语言内容是判断语音所属语种的关键高层特征，本文设计了一个多路多语种识别框架，因为在语音识别的训练过程中，与语言内容无关的噪声（加性噪声影响、说话人差异和录制环境差异）会被逐渐弱化，同时具有较好的泛化性。本文利用其后验输出，结合语言模型，构建了一个更加鲁棒的语种识别系统。

(2) 低资源语种识别问题

深度学习技术往往依赖于大量的数据集，但目前大规模语音数据集主要集中分布在常见的几种语种，例如汉语、俄语、英语等，然而世界上总共有约 6900 种语言，对于低资源的语种，现有的技术往往会导致识别效果较差，如何在低资源的环境下构建高性能的语种识别系统成为当前的研究难点。近年来，无监督预训练模型成为研究热点，基于大规模语音上无监督训练的模型往往具有更好的特征表示，应用于下游任务时往往具有更好的性能，特别是在监督数据较少时。本文所提出的方案利用了无监督预训练模型 WavLM，实验证明了其有效性。

（3）增强语音失真问题

有研究表明经过语音增强的语音可能存在一定的失真，虽然在一定程度上降低了噪声成分，然而由于失真可能会损害语音中的重要成分，降低下游任务例如语

音识别的性能[86]。针对该问题有学者提出在增强语音上重新训练的策略或联合语音增强前端和语音识别后端联合训练的方式，然而这些方式时间成本太高，联合训练的策略在一定程度上限制了模型大小和应用范围，在实际的场景中无法落地。如何利用语音增强对噪声的抑制优势来有效的提升下游任务例如语种识别的性能是亟待解决的关键问题。本章探索了如何利用语音增强提升语种识别在带噪场景下的性能。

4.2 算法描述

4.2.1 WavLM 无监督预训练模型

WavLM[87]是一种基于判别式的自监督模型，其通过预测被掩蔽语音的离散化标签进行训练，离散化标签由在训练时上一次迭代产生的模型直接生成，初始化标签是通过 k-means 聚类算法产生。WavLM 在 SUPERB 评测[88]上取得了最佳的成绩。模型的总体结构如图 4-1 所示，其由一个卷积编码器和 Transformer 编码器[89]组成。卷积编码器由七个卷积块构成，每个卷积块包含一维卷积、Dropout 层、层归一化和 GELU 激活函数。每个一维卷积有 512 个输出通道，其步幅（stride）分别为（5, 2, 2, 2, 2, 2, 2），卷积核宽度为（10, 3, 3, 3, 3, 2, 2）。卷积层的每个输出大约表征 25ms 的音频信息，帧移为 20ms。Transformer 是 2017 年谷歌提出的一种基于多头自注意力机制的网络，用于对序列到序列问题建模，其改变了传统的基于 CNN 或 RNN 建模方式，在多项任务中表现出良好的性能。在 WavLM 中并没有大幅更改 Transformer 模型结构，只是在注意力计算中引入了门控相对位置编码（gated relative position bias），使得模型能够更好的建模具有上下文关联关系的语音。

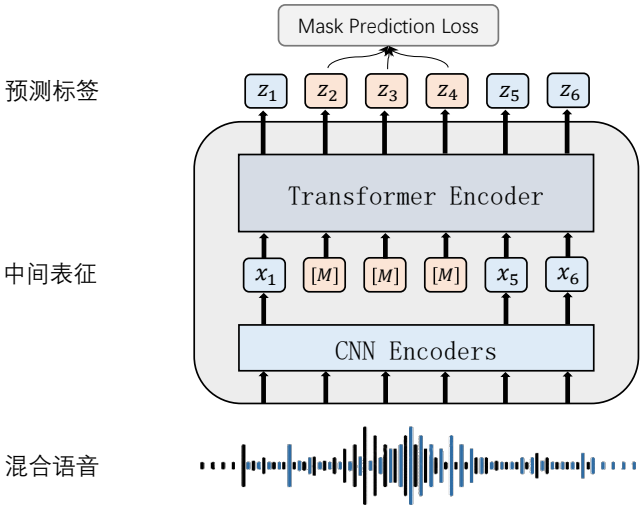


图 4-1 WavLM 模型结构

为了提高自监督模型的鲁棒性，使用了随机模拟噪声和重叠语音的方式来进行数据增强。具体的，对于训练过程中的每一批次数据$\mathrm{U}=\{u^i\}^B$，其中批大小（batch size）为B，u^i表示第i条语音，以概率p随机选择S条语音。对于这些语音，以在两种方法中随机的方式进一步处理。第一种是从批数据中随机选取一条语音，再从均匀分布$U(-5,5)$中选取一个能量比率γ；另一种方式是从 DNS 挑战中的噪声数据集随机选择一条噪声，再从均匀分布$U(-5,20)$中选取一个能量比率γ。选取的语音用u^{sec}表示，原始语音用u^{pri}表示。接下来开始将选取的语音混合到原始语音中，具体的，首先从均匀分布$U(1,\frac{L}{2})$中选取一个值l作为混合长度，其中L为原始音频长度。接下来，再从均匀分布$U(1,L-l)$中选取一个值作为原始音频起始位置s^{pri}，同样的方式对所选取的语音取一个起始位置s^{sec}。然后计算两个音频的平均能量，分别用E^{pri}和E^{sec}表示，通过能量比率γ计算得到一个放缩尺度：

$$scl=\sqrt{\frac{E^{pri}}{E^{sec}10^{\frac{\gamma}{10}}}} \tag{4-1}$$

最终，混合语音可表示为：

$$u^{pri}\left[s^{pri}:s^{pri}+l\right]=u^{pri}\left[s^{pri}:s^{pri}+l\right]+scl\cdot u^{pri}[s^{sec}:s^{sec}+l] \tag{4-2}$$

训练时，首先基于 HuBERT 模型[90]，将所有原始语音输入到模型中进行推理，得到其中 Transformer 的第 9 层特征，并通过 k-means 聚类算法将这些特征分为多个离散标签z。然后将混合的语音输入到 WavLM 中，并在卷积层选取 20%的特征进行掩蔽，经过 Transformer 层后，输出预测标签。损失函数由式(4-3)给出:

$$\mathcal{L}=-\sum_{l\in K}\sum_{t\in M}\log(z_t|h_t^l) \tag{4-3}$$

其中M表示被掩蔽特征的下标集合，z_t表示第t帧对应的标签，h_t^l表示第t帧第 l 层 Transformer 的中间输出，K表示 Transformer 的中间层。通常使用最后一层的 Transformer 输出作为下游任务的特征向量。

根据模型参数和训练集大小，WavLM 共分为三种类型： Base、Base+, Large。其中 Base 模型只使用 960 小时的 Librispeech 数据集进行训练，Transformer 模型为 12 层，其中隐藏层维度为 768，训练集语音加噪概率为 0。Base+模型和 Base 模型结构相同，其区别在于训练集大小达到了 94000 小时，训练集语音加噪概率为 0.1。Large 模型和 Base+模型的区别在于其使用了更深的 Transformer，其层数达到了 24 层，且隐藏层维度为 1024。三种模型的对比如表 4-1 所示。受限于计算机算力限制，本文只使用了 Base+模型。

表 4-1 WavLM 模型种类对比

模型名称	Transformer 层数	隐藏层维度	训练集大小	加噪概率
WavLM Base	12	768	960h	0
WavLM Base+	12	768	94000h	0.1
WavLM Large	24	1024	94000h	0.1

4.2.2 基于词法特征的语种判别

人类在说话人未知、声学环境未知的条件下依然能够准确的识别所熟悉的语种，其主要依靠对内容的深入理解，因此语言知识有助于提高语种识别的泛化性。受到这种现象的启发，本文提出了一种基于词法特征的语种识别方法，具体的，首先通过一个多语种语音识别模型作为语种先验知识，然后利用其在每个语种上的置信度作为语种判定分数，最后输出语种类别。相较于现有的研究方法，其在两个方面具有显著优势，首先是未知环境泛化性，语音识别的训练是序列到序列的过程，目标是帧级别的特征到字符的映射，在进行推理时，可解释性更强，而基于底层声学特征的方法是段级别特征到语种类别的映射，因此将词法特征用于语种识别的判定中具有更好的泛化性。另一个优势是本文所提出的方法计算复杂度更低，早期对使用词法特征进行语种识别的研究基于多路的语音识别，受限于其倍速增长的计算量消耗，无法实际的应用，本文探索了一种基于共享层的多语种识别模型，模型是一个树结构的神经网络模型，主干参数为所有语种共享，每个语种使用一个轻量化的模块来完成最后的识别，和主干相比，每个语种单独的模块参数量小，在语种有限的条件下，模型整体的复杂度较低。接下来从多语种识别模型的具体细节和如何利用其词法知识进行语种判定两个方面对语种识别方法进行阐述。

4.2.2.1 基于有监督的多语种语音识别模型构建

为了利用语言词法信息，本文构建了一个多语种语音识别模型。2020 年，Gulati 等人[91]提出了一种全新的 ASR 结构叫做 Conformer，其由多层 Conformer 块构成，比起基线模型，在 Librispeech 上实现了最优的性能。受到 Conformer 模型启发，为了实现一个较优的语音识别系统，同时使得模型参数量尽可能低，本文提出的多路多语种语音识别模型基于 Conformer 架构，其主要分为两个部分：共享特征提取器和特定语种识别器。共享特征提取器由 14 层 Conformer 块构成，为所有语种共享，为了进一步提高模型的鲁棒性，在特征提取器前引入数据增强模块，其从两个维度对音频进行增强，首先是时域增强，包括对音频加混响和 0.9、1.1 倍随机调速，这种方法直接作用于原始语音波形。其次是频域增强，音频转化到 FBank 特

征后，使用 SpecAugment 技术[92]对特征选择随机大小区域进行 mask。特定语种识别器由一个 Conformer 块和一个线性映射层组成，每个语种都有一个单独的模块，线性映射层的目的在于将输出大小和每个特定语种的词典大小匹配。模型在优化时，每个语种特有的模块都由单独的 CTC 损失函数约束。最终构建出一个多语种语音识别模型，如图 4-2 所示。

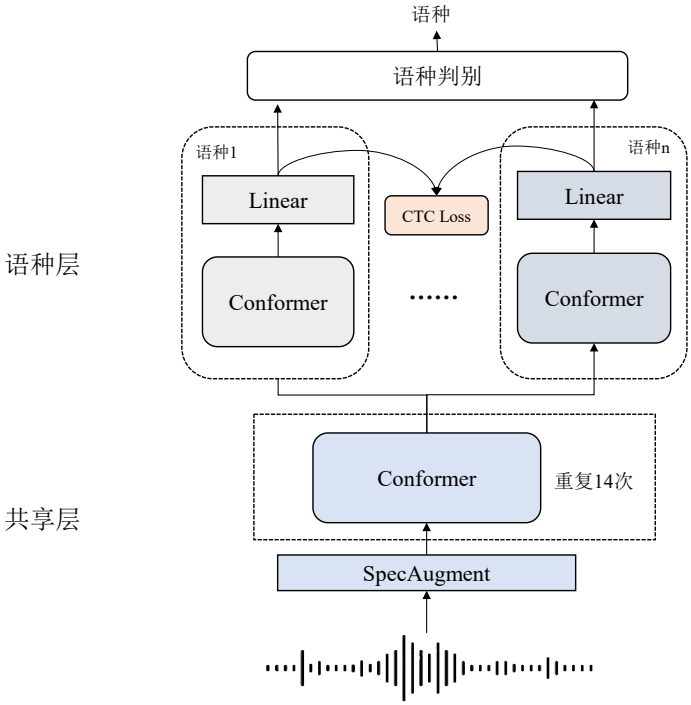


图 4-2 多语种语音识别模型

其中 Conformer 块的结构如图 4-3 所示。其由两层前馈模块，一层多头自注意力模块和卷积模块构成，最后通过一个层归一化层。前馈模块由层归一化层、线性映射层、Swish 激活函数、Dropout、线性激活层和 Dropout 构成，其中线性映射层输出大小为 144，提取的 FBank 特征维度为 80。多头自注意力模块使用了相对位置编码来建模位置信息，其中注意力 head 数为 4，维度为 64。卷积模块使用深度可分离卷积，其中卷积核大小为 31，同时激活函数使用了 GLU 和 Swish 激活函数。Conformer 的输入表达式如下式所示：

$$\widetilde{x_i} = x_i + \frac{1}{2}\mathrm{FFN}(x_i) \tag{4-4}$$

$$x_i' = \widetilde{x_i} + \mathrm{MHSA}(\widetilde{x_i}) \tag{4-5}$$

$$x_i'' = x_i' + \mathrm{Conv}(x_i') \tag{4-6}$$

$$y_i = \text{Layernorm}(x_i'' + \frac{1}{2}\text{FFN}(x_i'')) \tag{4-7}$$

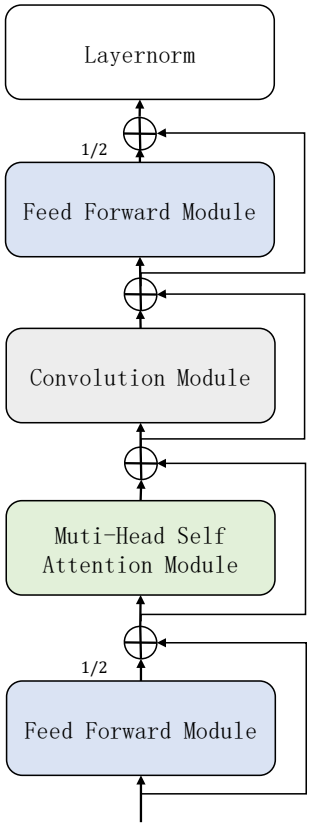


图 4-3 Conformer 块结构

本文使用 NovoGrad 优化器[93]，学习率为 0.01，权值衰减系数为 0.0001，同时为了充分训练模型，本文使用了一种三阶段学习率调度策略，前 10%的轮次学习率从 0.0001 预热到 0.01，随后 40%的轮次保持固定，最后以指数级降低到 0.0001，总的训练轮次为 100 轮。对于频谱增强，在时间维度上从均匀分布$U(1,0.05 \times L)$上选择一个时域掩蔽长度l，其中L表示音频的实际长度，并在音频上随机选择一个起点s_t，在特征维度，从均匀分布$U(1,12)$中选择特征掩蔽长度m，同样选择一个特征的起点s_f。最终对如下矩形区域进行掩蔽：

$$\text{feature}[s_t : s_t + \text{l}, s_f : s_f + \text{m}] = 0 \tag{4-8}$$

本文采用全连接时序分类（Connectionist temporal classification, CTC）[94]作为损失函数，每个语种的损失函数之间相互独立。CTC 用于时序类数据的分类问题，其最大的优势在于不需要音频与文本的对齐，极大的方便了语音识别的端到端建模。CTC 中引入了空白字符概念，其表示静音帧或相邻帧之间的分割字符，用于解码和损失函数的计算。对于任一模型，其输出为$\text{y} = (y_1, \ldots, y_T)$，其中$T$表示音频

帧长度，$y_i = (x_1, \dots, x_N)$其中$x_i \in W$，N为词典W的大小。

在解码时，通常选取当前帧概率最大的字符作为当前帧的输出字符，接下来是 CTC 解码算法。由于同一字符可能由多个相邻帧构成，因此在 CTC 解码过程中，相邻帧输出相同时只会输出一个字符（若相邻字符相同，通过空白帧进行分割来保证解码的准确性），最后得到的序列中去掉多余空白帧即完成了 CTC 解码。

对于模型的输出，可以看作是一$N \times T$的概率矩阵，显然，这个矩阵可能存在多条路径，使得对于其中任意一条路径，使用 CTC 解码方法即可将其转化为目标序列。而 CTC 的目标是在这个概率矩阵中找到所有可能的路径，计算每个路径的概率并求和，CTC 损失函数由下式给出：

$$L = -\log\left\{\sum_{P\in\mathcal{D}}\prod_{t=0}^{T} p_t(a_t|X)\right\} \tag{4-9}$$

其中P表示所有可能的路径，$p_t(a_t|X)$表示路径中第t时刻字符a_t的后验概率，X表示音频序列。然而直接找到所有的路径是非常复杂的，其时间复杂度达到了$O(N^T)$。在 CTC 中提出了一种动态规划算法，能够以$O(N \times T)$的时间复杂度完成路径搜索，大幅提高了 CTC 的计算效率。接下来将结合一个例子对 CTC 动态规划进行解析，如图 4-4 所示：

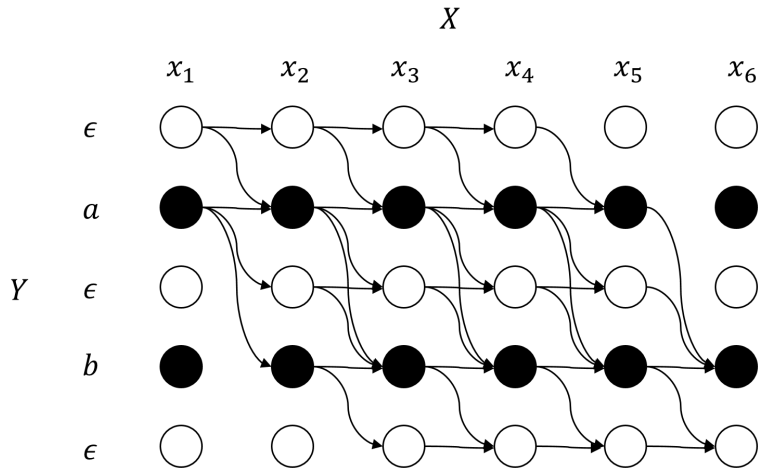


图 4-4 CTC 动态规划示意图

其中，X为模型输入，Y表示目标序列"ab"，插入空白字符ϵ后，目标序列变为"$\epsilon a\epsilon b\epsilon$"，对于图 4-4 中的每一点，表示序列$X_{1:t}$和目标序列$Y_{1:s}$对齐的概率，箭头向左移动一个单位表示当前帧解码为空白帧，箭头向下可移动一个或两个单位到并到下一个时间点，则当前帧解码为Y轴上对应的下一个字符。用$\alpha_{s,t}$表示图中每一个点的对齐概率，那么根据是否可以移动两个字符得到如下递推公式：

$$\alpha_{s,t} = (\alpha_{s-1,t-1} + \alpha_{s,t-1}) \cdot p_t(z_s|X) \tag{4-10}$$

$$\alpha_{s,t} = (\alpha_{s-2,t-1} + \alpha_{s-1,t-1} + \alpha_{s,t-1}) \cdot p_t(z_s|X) \tag{4-11}$$

如图 4-4 所示，若上一帧已经解码为ϵ空白字符，则只能向左移动或向下移动一个单位到下一个时间点。若为其他情况，则才有可能向下移动两个单位。在解码过程中可进行剪枝降低计算量，如$\alpha_{1,4}$点不会再向左移动，因为剩余帧数量限制，移动后其不可能再和目标序列对齐。最后得到多个完全和目标序列对齐的路径，节点值即为路径的概率。

4.2.2.2 基于无监督的多语种语音识别模型构建

全世界有超过 6000 种的语言，对于常见的语言例如英语、汉语、俄语等有大量的数据集用于语音识别的训练，然而大部分其他语种都缺乏充足的标注语料，导致基于有监督学习的语音识别模型性能较差甚至无法收敛。在常见的语种例如汉语中，也存在各种方言或特殊领域，其受限的训练数据无法将语音识别推向实用。基于自监督的预训练模型在大量的无标注数据上进行训练，其能学习到语音更好的表征，因此本文结合自监督模型 WavLM 和上文提出的多语种语音识别模型架构，提出了一种基于自监督的多语种语音识别模型。相较于上文提出的模型，将主干中的 Conformer 直接替换为 WavLM，利用其输出的中间表征作为语种层的输入。由于 WavLM 直接基于时域语音建模，因此数据增强模块去掉了频域掩蔽部分，只在时域进行增强。模型结构如图 4-5 所示：

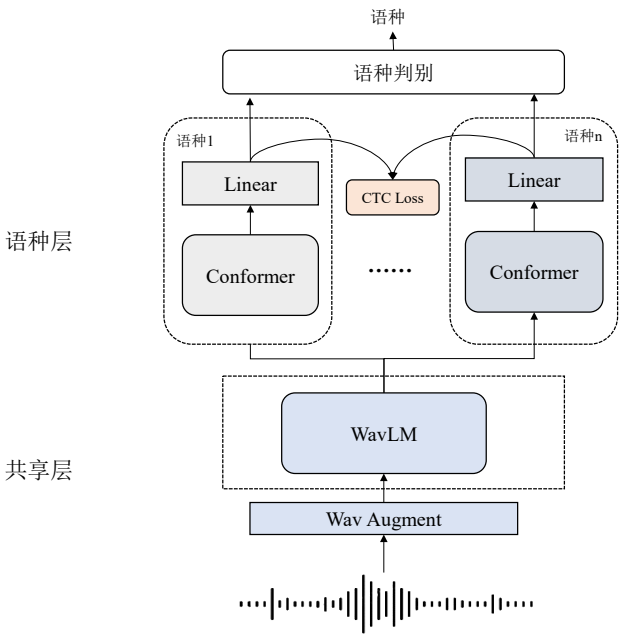


图 4-5 基于 WavLM 的多语种语音识别模型

训练过程中使用 Adam 优化器，学习率为 0.0001，和上文一样，使用三阶段的学习率调度策略，损失函数同样基于 CTC。WavLM 的特征已经具有较好的分类能力，因此在训练的前两个轮次，保持 WavLM 的参数固定，只训练语种层的参数，2 个轮次后，loss 趋于稳定，再将共享层和语种层参数一起进行训练。由于 WavLM 参数量较大，这里选择训练的批大小为 8。为了提升模型了泛化性能，对 WavLM 中的卷积层输出的特征以 0.15 的概率随机掩蔽。在训练过程中，本文使用一种特殊的 batch 采样方式，使得 batch 中的所有语音都同属于一个语种，有效提高了训练效率。

4.2.2.3 基于词法特征的语种判别模块

多语种语音识别模型输出每个音频帧对应的字符，这个过程对应着语言学中的词法。假设音频序列$X = (x_1, x_2, x_3, \dots, x_T)$，对于多语种语音识别模型，其在每一个语种上都有一个后验输出$\mathrm{O} \in \mathbb{R}^{T \times N}$，其中$T$表示音频帧长度，$N$表示该语种的词典大小。首先使用 softmax 将输出限制在 0-1 范围内。

$$\mathrm{softmax}(o_i) = \frac{\mathrm{e}^{o_i}}{\sum_{k=0}^{N} \mathrm{e}^{o_k}} \tag{4-12}$$

再使用贪心算法对每个语种下的输出进行解码，除去无意义的空白帧字符，得到序列$\mathrm{Z} \in \mathbb{R}^{L}$，其中$L$为该语种 CTC 解码后文本长度。然后计算解码字符的几何平均概率：

$$\mathrm{P} = log\left\{\prod_{t=1}^{L} p(z_t|X)\right\}^{\frac{1}{L}} = \frac{1}{L} \cdot \sum_{t=1}^{L} \log\big(p(z_t|X)\big) \tag{4-13}$$

其中对概率取对数进行计算。将这个分数作为语音在当前语种上的置信度。但由于各语种词表大小不一致，当 softmax 进行归一化计算后，会导致各语音识别模块计算的置信概率分布不一致，从而无法进行有效的比较。因此引入一个放缩因常数μ。置信分数可表示为$\mu\mathrm{P}$。

放缩因子μ在一定程度上保证了不同语种之间对置信分数比较的公平性，然而其只是线性的近似，例如当不同语种模块计算的概率趋近 1 时，表明该语音在所有语种上都应该具有同样的高置信度，但是在μ因子的影响下，会使得计算得到的置信分数偏向于词典空间最小的语种。经过实验验证，这种线性近似也具有较好的性能，因为经过 softmax 后计算得到的概率值大部分在一个较小的范围内，在这个范围内，不同语种的置信度可近似为线性关系。接下来通过实验验证其近似线性关系的猜想。基于本文提出的多语种识别模型，测试了两个语种：阿拉伯语和斯瓦西

里语中的大约 650 条语音，分别使用该语音所属的语种模块计算得到置信分数，对其排序，得到的分布曲线如图 4-6 所示：

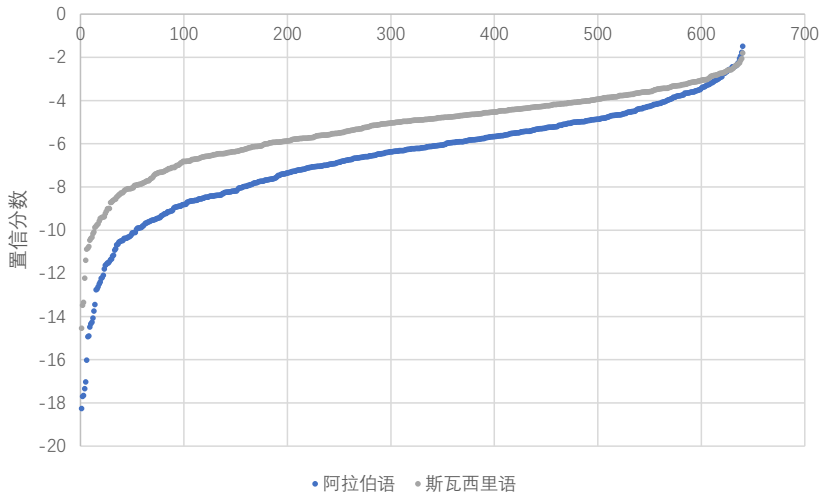


图 4-6 阿拉伯语和斯瓦西里语置信分数散点图

其中横坐标为排序后的下标，纵坐标为置信分数，对于拥有相同横坐标排名的置信分数，可以认为两者的置信度是等价的。进一步求得两个语种置信分数的比值曲线，如图 4-7 所示：

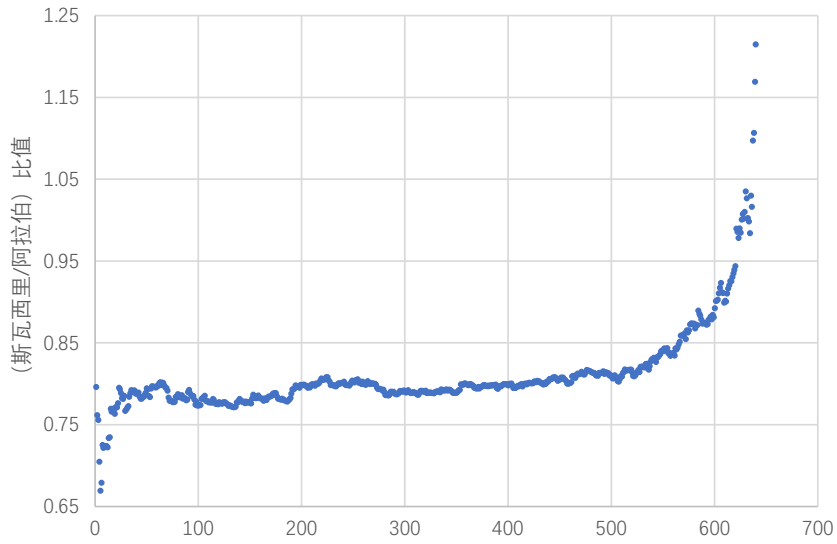


图 4-7 阿拉伯语和斯瓦西里语置信分数比值

可以看到，在 20 到 550 区间范围内，其比值可认为是一常数，表明在一定范围内，不同语种模块所估计的置信度关系可近似为线性的。同理，本文测试了更多的语种对之间的置信分数比值，在特定范围内，曲线斜率较小，波动范围不超过 0.1。

对于偏差较高的部分，本文通过实验进行了进一步的探索。针对阿拉伯语语音，获取到其在阿拉伯语和斯瓦西里语这两个语种模块输出的置信度分布曲线，如图 4-8 所示：

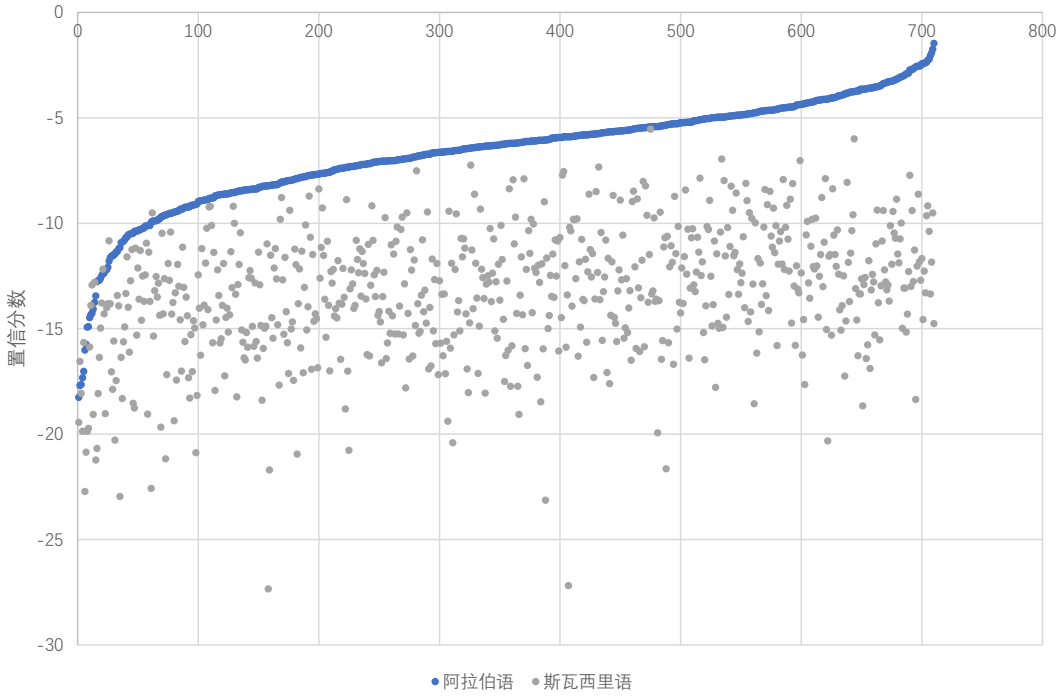


图 4-8 阿拉伯语语音在阿拉伯语模块和斯瓦西里语模块下置信分数散点图

其中对于超过 550 区间的范围，阿拉伯语语种下的置信度分数远高于斯瓦西里语，即使较低的放缩因子不能准确的得到两个语种等价的置信度，也能根据其大小准确判定语种类型。接下来将对放缩因子进行估计。

一种较为精确的估计方式是直接对验证集输出的置信分数比值曲线进行拟合，并作为μ值，然而由于过度拟合验证集对于未知环境可能表现较差。本文采用另一种直接估计的简便方法，假设存在一条音频，使得模型对其音频帧的输出概率在所有字符上均等，表明对于所有语种下的语音识别模块都无法正确识别语音的内容，对于音频的每一帧都是同等困惑，那么各语音识别模块最后的得分应该等价。由此对于任意两个语种，选择一个语种作为基准，另一个语种的分数进行放缩可得到如下恒等式：

$$-\frac{1}{S_1}\sum_{i=0}^{S_1}\log\left(\frac{1}{N_1}\right) = -\mu\frac{1}{S_2}\sum_{i=0}^{S_2}\log\left(\frac{1}{N_2}\right) \tag{4-14}$$

其中等式两边分别表示两个不同的语种模型对语音给出的置信分数，μ即为放缩因子。进一步，求解μ可得：

$$\mu = \frac{\log(N_1)}{\log(N_2)} \tag{4-15}$$

最后本文选择一个基准语种，对其他语种得到的置信分数进行放缩比较，选择置信分数最高的语种模型作为当前语音的语种属性。

4.2.3 基于语法特征的语种判别

词与词之间的搭配关系称为语法，不同语种的语法规则各不相同，因此可以利用语法差异区分不同的语种。语言模型是一种用于衡量单词序列合理程度的概率模型，对于一给定单词序列$\mathrm{W} = (w_1, w_2, \ldots, w_n)$，语言模型可以输出单词序列的概率值P(W)，概率越大，表明单词序列在该语种中越合理。因此语言模型在某种程度上反映了句子语法的合理性，从而可以通过不同语言模型的概率分数识别语种。本文使用了一种基于统计的语言模型 n-gram，首先为不同语种构建独立的语言模型，再通过对多语种识别模型的多个输出进行贪心解码获取到相应语种下的文本，接下来对文本进行打分，通过比较相互之间的困惑度的大小，完成语种的识别。n-gram 语言模型的构建主要分为两个部分：概率统计和平滑。

对于一条文本，在其开始和结尾添加标记符号，得到的序列可用$(w_1, w_2, \ldots, w_n)$表示，显然其概率可由下式给出：

$$\begin{aligned} P(w_1, w_2, \ldots, w_n) &= P(w_1)P(w_2|w_1)\cdots P(w_n|w_1, w_2, \cdots, w_{n-1}) \\ &= \prod_{i=1}^{n} P(w_i|w_1, w_2, \cdots, \ w_{i-1}) \end{aligned} \tag{4-16}$$

其中$P(w_i|w_1, w_2, \cdots, \ w_{i-1})$表示第$i$个字符在已知前$i-1$个字符下的条件概率。通常情况下，对于文本序列的预测可以看作为一个马尔可夫过程，即当前文本只与前 N 个单词有关，与超过 N 距离的单词无关，N 即为 n-gram 单词中的“n”，通常 N 的大小为 2 或 3，其对应的语言模型分别叫做 bigram 和 trigram。因此条件概率可简化为下式：

$$P\left(w_i|w_1, w_2, \cdots, \ w_{i-1}\right) \approx P\left(w_i|w_{i-N}, w_{i-N+1}, \cdots, \ w_{i-1}\right) \tag{4-17}$$

进一步，条件概率可通过极大似然估计求出：

$$\begin{aligned} P(w_i|w_{i-N}, w_{i-N+1}, \cdots, w_{i-1}) &= \frac{P\left(w_{i-N}, w_{i-N+1}, \cdots, \ w_{i-1}, w_i\right)}{P\left(w_{i-N}, w_{i-N+1}, \cdots, \ w_{i-1}\right)} \\ &= \frac{C\left(w_{i-N}, w_{i-N+1}, \cdots, \ w_{i-1}, w_i\right)}{C\left(w_{i-N}, w_{i-N+1}, \cdots, \ w_{i-1}\right)} \end{aligned} \tag{4-18}$$

其中$C(\mathrm{W})$表示序列W在所有组合中出现的频数。显然，如果我们知道所有字符组合出现的频数即可计算出任意序列的概率，而语言模型训练的目标即得到所

有可能的组合概率。在训练阶段，本文为每个语种构建一个训练文本数据集，对其中的文本进行清洗，去除无意义符号。然后统计其中各个组合出现的频数。可以看到，随着训练文本的增加，其统计得到的条件概率越接近于真实分布，但统计成本和储存成本也会随之提升。然而这种建模方式存在一个显著缺点，即数据稀疏问题，若测试的句子中包含训练集中未出现的词或组合时，会出现概率为零的情况，通常通过平滑来解决这个问题。

常用的平滑技术包括折扣法、插值法和回退法。其中折扣法是从现有的概率中分配一部分给未出现的组合。插值法是混合不同 N 的语言模型通过线性插值得到训练中未出现的组合的概率。回退法是利用低阶 n-gram 模型的概率替代高阶模型中未出现的组合概率。本文中使用 Kneser-Ney 平滑，其是一种广泛使用的平滑技术，它结合了绝对折扣法和回退法。绝对折扣法使用一个固定值对频数进行折扣，如下式所示：

$$P\left(w_i\middle|w_{i-n+1}^{i-1}\right)=\frac{\max\left(C\left(w_{i-n+1}^{i}\right)-D,0\right)}{C\left(w_{i-n+1}^{i-1}\right)}+\lambda(w_{i-n+1}^{i-1})P\left(w_i\middle|w_{i-n+2}^{i-1}\right) \tag{4-19}$$

其中w_{i-n+1}^{i-1}表示第$i-n+1$到$i-1$之间的字符，D 即为折扣值，通常在 0-1 之间。$\lambda(w_{i-n+1}^{i-1})$为折扣系数，由下式给出：

$$\lambda(w_{i-n+1}^{i-1})=\frac{D}{C\left(w_{i-n+1}^{i-1}\right)}\left|\{v:C\left(w_{i-n+1}^{i-1}v\right)>0\}\right| \tag{4-20}$$

其中$\left|\{v:C\left(w_{i-n+1}^{i-1}v\right)>0\}\right|$表示字符串$w_{i-n+1}^{i-1}$后出现新词的频数。对于式(4-19)中等式右边的低阶部分，其只是使用最大似然估计w_{i-n+1}^{i-1}字符串的概率，然而其存在一个较大的问题。考虑一场景，例如构建一基于成电校园文本的语言模型，“大气大为”词组出现的频率较高，因此对于一个 bigram，P(为|大)为一较大的值，考虑一任意字符h，那么在计算高阶的字符串例如“h 大为”的条件概率$P\left(\text{为}\middle|h\ \text{大}\right)$时，低阶部分将会给这个条件概率赋予一个较大的值，显然这是不合理的。Kneser-Ney 平滑认为低阶 n-gram 模型的概率应该和词能组成的新词种类数成正比关系，其种类越多，表明该词和词典中其他词组合的概率更大。因此对低阶模型进行修正：

$$P_{continuation}(w_i)=\frac{|\{v:C(vw_i)>0\}|}{\sum_w|\{v:C(vw)>0\}|} \tag{4-21}$$

该概率也被称作连续性概率。结合式(4-19)一个 bigram 的 Kneser-Ney 平滑为：

$$P(w_i|w_{i-1}) = \frac{\max(C(w_{i-1}w_i) - D, 0)}{C(w_{i-1})} + \lambda(w_{i-1})P_{continuation}(w_i) \tag{4-22}$$

最后，计算该句子在本文所构建的语言模型上的困惑度。困惑度用于衡量一个句子在该语种对应语言模型中的合理程度，其通常作为语言模型的评价指标。其计算公式如式(4-23)所示：

$$\begin{aligned} PPL(W) &= P(w_1 w_2 \dots w_N)^{-\frac{1}{N}} \\ &= \sqrt[N]{\frac{1}{P(w_1 w_2 \dots w_N)}} \\ &= \sqrt[N]{\frac{1}{\prod_{i=1}^{n} P(w_i|w_{i-n+1}^{i-1})}} \end{aligned} \tag{4-23}$$

通过比较语音识别结果在不同语种下的困惑度大小，以困惑度最小的语言模型对应的语种作为当前语音的语种。

4.3 实验与讨论

本章涉及语种识别相关实验、语音增强和语种识别整合实验，验证了在未知环境、复杂噪声环境下的语种识别性能。为了方便代码的移植，本章实验的其他硬件和软件条件和第三章保持一致。

4.3.1 实验数据及设置

本章采用 Common Voice 数据集和科大讯飞多语种数据集作为语种识别数据集。Common Voice 从世界各地收集语音，因此采集环境和采集设备多种多样，其包含 13,905 小时录音，76 种语言的语音数据，并且还在不断更新中，这些数据中同时包含了年龄、性别、口音等人口统计元数据。科大讯飞多语种数据集来源于科大讯飞举办的竞赛，其包含 3 个语种，斯瓦西里语、越南语和阿拉伯语，每个语种大约只有 10 小时有标注训练数据，用于模拟低资源语种，其在统一的录音室进行录制，因此无环境噪声，和 Common Voice 数据存在较大的分布差异。

为了测试模型在带噪环境下的性能，本章使用第三章中的 Noisex92 噪声数据集，作为噪声来源。选取 0、5、10、15dB 常见信噪比对语种数据集进行混合，使用的噪声为餐厅嘈杂噪声（babble）、驱逐舰引擎噪声（destroyer engine）、驱逐舰操作舱噪声（destroyer-ops）和工厂车间噪声（Factory1）一共 4 种噪声。降噪模型使用第三章提出的最优的增强模型，由于只在 TIMIT 英文数据集上进行了训练，

在非英语的语音上不能完全发挥出其性能优势，因此在本章中使用了多语种语音数据集对增强模型进行了重新训练，训练噪声和方法与第三章保持一致。

4.3.2 对比方法及评价指标

本文采用以下的基线模型：

（1）X-Vector[66]：X-Vector 是一种基于 DNN 的端到端语种识别模型，通过池化层将帧级别的特征处理成句级别特征，具体来说就是通过 TDNN 层提取特征向量，将这些特征向量的均值和方差拼接作为句级别特征，最后通过一层前馈神经网络输出分类概率。该模型广泛用于语种识别模型的基准模型。

（2）Resnet[95]：Resnet 即残差网络，在图像识别领域应用较为广泛，由于音频在通过短时傅里叶变换后的特征图和图像类似，因此也可用于语种识别中。相较于原始的残差网络，本文采用时间统计池化（Temporal Statistics Pooling, TSTP），其通过将均值和方差拼接作为最后线性映射层的输入。

（3）XLSR[45, 95]：基于 Wav2Vec2 自监督预训练模型，在其上进行迁移的语种识别方法。其属于基于底层声学特征的分类方法。

本文提出的两个模型符号表示如下：

（1）Conformer：本文提出的基于 Conformer 的多语种识别模型，如 4.2 节所述。AM 表示使用 ASR 声学模型的置信输出进行语种识别，LM 表示使用 ASR 的文本输出对语种进行判别。

（2）WavLM：本文提出的基于 WavLM 自监督模型的语种识别模型，如 4.2 节所述。其中 AM 和 LM 含义和 Conformer 模型相同。

4.3.3 实验结果与分析

4.3.3.1 未知环境泛化性

为了测试所提出方法在未知环境下的泛化性能，本文在科大讯飞数据集上进行了语音识别的训练，并使用 Common Voice 中对应的语种进行测试，需要注意的是 Common Voice 和科大讯飞的录制环境和内容具有较大差别。为了更好的对比各个模型性能之间的差异，本文让训练的超参数保持一致，对于基于预训练的多语种语音识别模型，由于预训练的参数可以加速模型的收敛，一共训练 20 轮，其余有监督模型训练 100 轮。实验结果如表 4-1 所示。

对于本文中提到的基于多语种语音识别的语种识别方法，其中用 AM 表示直接利用语音识别模型的置信输出进行语种判别，如本章 4.2.2 节所述。LM 表示利用各自模块输出的文本进行困惑度打分来进行语种识别，如本章 4.2.3 节所述。为

了能够更加充分利用词法和语法特征，将 AM 和 LM 的输出进行了融合，具体的，对于 AM 输出的每个语种的分数，当不同语种之间的差值小于一个固定的阈值时，可以认为 AM 对这条语音的结果置信度偏低。因此这部分语音交给 LM 进行语言模型的打分，这个阈值可自由调节。

表 4-2 训练条件不匹配条件下 EER 和C_{avg}

指标	方法	模型	科大讯飞测试集	Common Voice 测试集
EER (%s)	有监督	X-Vector	16.97	20.84
		Resnet	0.64	20.78
		Conformer +AM	0.69	**11.34**
		Conformer +LM	4.80	21.53
		Conformer+AM+LM	**0.64**	11.73
	无监督	WavLM+AM	0.39	6.30
		WavLM+LM	0.07	5.37
		WavLM+AM+LM	**0.02**	**4.27**
		XLSR	0.37	5.40
C_{avg} （%s）	有监督	X-Vector	16.55	30.27
		Resnet	**0.60**	23.38
		Conformer +AM	0.92	9.40
		Conformer +LM	4.40	23.04
		Conformer+AM+LM	0.75	**9.80**
	无监督	WavLM+AM	0.37	5.03
		WavLM+LM	0.07	7.80
		WavLM+AM+LM	**0.01**	**3.40**
		XLSR	0.35	10.61

从实验中可以看出，本文所提出的方法在 EER 和C_{avg}指标上均优于基准模型，证明了该方法的有效性，结合声学模型和语言模型的分数进一步提升了语种识别的性能。通过将在科大讯飞数据集上训练得到的模型直接在 Common Voice 测试集上进行测试，本文的方法有一定程度的性能下降，然而 X-Vector 和 Resnet 这种端到端建模的算法性能下降较为明显，EER 从 16.97%和 0.64%下降到 20.84%和 20.78%，C_{avg}从 16.55%和 0.6%下降到 30.27%和 23.38%。对比 Conformer 和 Resnet 可以发现，在域匹配的条件下，两者性能相当，然而，在未知环境下，Resnet 性能下降更加明显，两者的 EER 分别是 11.73%和 20.78%，C_{avg}分别为 9.8%和 23.38%，Conformer 多语种模型下降幅度更低，证明本文所提出的方法具有更好的泛化性。

基于自监督的方法相较于有监督方法具有更好的性能，表明自监督模型在大规模语音数据集学习到更好的特征表示，特别是对于只有少量训练数据的低资源语种识别任务，其有效的提升了下游任务的性能。

4.3.3.2 噪声环境鲁棒性

为了验证所提出的方法在噪声环境下的表现，本文使用 Noisex92 中的 4 种噪声（Factory1、Babble、White 和 Factory2），其中 Factory1 和 Babble 是训练集中出现的噪声，而 White 和 Factory2 是训练集未出现的噪声，用于模拟噪声不可见的场景。本文选取的噪声中既有平稳噪声（White）也有非平稳噪声（Factory2），能够覆盖常见的应用场景。随后以 4 种不同的信噪比（0dB、5dB、10dB、15dB）在科大讯飞测试集上进行了测试，实验结果如下表所示：

表 4-3 SNR=0 时语种识别 EER 和C_{avg}指标

噪声类型	可见噪声				不可见噪声			
	factory1		babble		white		factory2	
指标	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}
X-Vector	50.00	47.57	47.66	44.81	41.85	40.53	51.33	46.17
Resnet	**42.32**	**40.09**	**39.04**	45.99	40.84	36.83	34.88	45.32
Conformer	43.60	42.95	40.42	**39.49**	**35.52**	**34.80**	**22.76**	**22.97**
WavLM+AM	**7.17**	**8.07**	**10.99**	**11.73**	**4.48**	**5.19**	1.18	1.15
WavLM+LM	18.77	17.49	21.08	20.50	13.08	12.35	1.97	1.90
WavLM+AM+LM	10.20	9.10	13.74	12.14	6.40	5.95	**1.13**	**0.97**
XLSR	31.60	30.45	28.23	27.65	21.03	20.12	5.27	5.28

表 4-4 SNR=5 时语种识别 EER 和C_{avg}指标

噪声类型	可见噪声				不可见噪声			
	factory1		babble		factory1		babble	
指标	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}
X-Vector	47.98	43.72	42.73	39.58	40.10	38.91	45.76	40.95
Resnet	39.66	37.98	35.64	39.58	31.63	31.01	33.72	35.39
Conformer	**28.35**	**27.68**	**22.41**	**21.83**	**18.55**	**18.20**	**10.22**	**9.58**
WavLM+AM	**1.95**	**1.94**	2.29	2.22	**2.32**	**2.62**	0.74	0.80
WavLM+LM	4.21	4.07	3.45	3.33	5.12	4.95	0.44	0.42
WavLM+AM+LM	2.17	2.01	**1.67**	**1.57**	2.56	2.48	**0.22**	**0.24**
XLSR	8.87	8.65	10.34	10.20	6.45	6.17	1.11	1.12

表 4-3 到 4-6 中展示了在信噪比为 0dB、5dB、10dB 和 15dB 时语种识别的性

能，从表中可以看出，带噪语音相较于干净语音在语种识别指标上有明显下降。值得注意的是本文所提出的方法相较于其他基线模型下降幅度较小。对于有监督模型，在大于 5dB 的信噪比条件下，Conformer 模型性能远高于 Resnet 和 XVector，在 0dB 条件下，几种模型都表现较差，原因在于噪声对语音成分的干扰导致模型失效，训练集有限的条件下难以适应低信噪比的噪声。对于基于自监督学习的方法，在 0dB 的场景下，基于分类的 XLSR 模型 EER 在某些噪声条件下达到了 31.6%，C_{avg}为 30.45%，几乎完全失效，而基于 WavLM 的多语种模型 EER 最低只下降到 10.2%，C_{avg}也仅仅降低为为 9.10%，其在较低信噪比条件下表现良好，得益于其混合噪声进行掩蔽的训练方法。

表 4-5 SNR=10 时语种识别 EER 和C_{avg}指标

噪声类型	可见噪声				不可见噪声			
	factory1		babble		white		factory2	
指标	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}
X-Vector	45.67	41.00	36.16	32.33	40.44	39.71	35.62	33.50
Resnet	35.49	39.36	32.29	30.23	34.68	37.79	26.45	24.11
Conformer	**11.50**	**11.67**	**7.49**	**8.38**	**10.54**	**11.26**	**3.84**	**3.90**
WavLM+AM	0.86	1.11	0.54	0.85	**0.74**	0.96	0.49	0.61
WavLM+LM	1.08	1.04	0.99	0.94	1.60	1.62	0.22	0.22
WavLM+AM+LM	**0.74**	**0.62**	**0.39**	**0.38**	0.99	**0.93**	**0.10**	**0.08**
XLSR	1.38	1.40	2.27	2.31	1.40	1.40	0.49	0.46

表 4-6 SNR=15 时语种识别 EER 和C_{avg}指标

噪声类型	可见噪声				不可见噪声			
	factory1		babble		white		factory2	
指标	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}
X-Vector	39.85	33.67	28.23	25.75	38.28	36.03	28.05	26.26
Resnet	34.33	34.78	22.51	20.58	35.07	42.81	8.74	8.54
Conformer	**5.22**	**5.36**	**2.66**	**2.87**	**5.47**	**5.46**	**1.67**	**1.66**
WavLM+AM	0.64	0.51	**0.37**	0.63	0.67	0.68	0.34	0.42
WavLM+LM	0.25	0.32	0.39	0.35	0.79	0.68	0.10	0.07
WavLM+AM+LM	**0.15**	**0.12**	0.39	**0.13**	**0.15**	**0.12**	**0.00**	**0.00**
XLSR	0.39	0.39	0.89	0.91	0.59	0.50	0.44	0.42

可以发现，在信噪比较低的场景下，联合 AM 和 LM 的方法性能甚至低于只使用 AM 的方法，其原因在于低信噪比条件下语音识别的准确率大幅下降，其置信度的区分性降低，导致更多的语音交给 LM 进行判别，又因为低信噪比下转录

的文本错误较多，从而从总体上降低了语种识别的性能。但由于 AM 和 LM 的阈值可以进行调节，对于低信噪比的场景，我们可以降低其阈值以获得更优的性能。

4.3.3.3 联合语音增强的语种识别

上文中可以看到，噪声的引入降低了语种识别的精度，并随着信噪比的增加，噪声成分逐渐淹没语音成分，语种识别的性能逐渐下降，甚至可能导致语种模型的失效。语音增强旨在抑制语音中的噪声分量，因此从理论上说，引入语音增强会提升下游任务在带噪场景下的性能。然而，通过实验发现，简单的将语音增强的语音进行识别可能会降低语种识别的准确率。接下来将对这一现象进行分析。对于一条语音s，增强后的语音用$\hat{s}$表示。其关系如下式所示：

$$\hat{s} = s + \lambda n + d \tag{4-24}$$

其中n表示噪声，λ表示语音增强算法处理后剩余的噪声比例，d表示语音增强引入的失真，λ是一小于 1 的小数，由于噪声水平的降低不会反过来降低下游任务的性能，因此失真d的引入是造成语种识别错误率上升的直接原因。这种失真可能会抵消干净语音中的有效成分，导致其语种识别准确率低于直接将带噪语音进行语种识别得到的结果。本文采用一简单的方法降低失真d在$\hat{s}$中的比例，通过引入一加权因子α，将带噪语音在增强语音上进行混合从而对缺失的信息进行补偿。处理后的增强语音如下式所示：

$$\begin{aligned} \hat{s}' &= (1-\alpha)\cdot\hat{s} + \alpha\cdot(s+n) \\ &= s + (\lambda + \alpha - \alpha\lambda)n + (1-\alpha)d \end{aligned} \tag{4-25}$$

定理 4.1 带噪语音和增强语音混合后的噪声分量系数大于原增强语音中噪声分量系数。

证明：

首先混合噪声分量和原始噪声分量差值为：

$$(\lambda + \alpha - \alpha\lambda) - \lambda = (1-\lambda)\alpha \tag{4-26}$$

由于语音增强模型对噪声进行了抑制，因此$\lambda < 1$，所以$1-\lambda > 0$，又因为加权系数$\alpha \in \{x|0 < x < 1\}$，所以$(1-\lambda)\alpha > 0$。

综上，混合后噪声分量$(\lambda + \alpha - \alpha\lambda) > \lambda$。结论得证。 ■

引入加权因子可以降低失真分量，但同时在一定程度上提高了噪声分量的比重。在较高信噪比条件下，失真对性能的影响远高于噪声带来的影响，因此这种方法能够真正将语音增强应用在语种识别的任务中，提高在带噪场景下语种识别的性能。下面通过实验验证所提出的引入加权因子的方法，并进一步探索语音增强对语种识别在不同信噪比和噪声类别下的提升空间。

本文选择 factory1 噪声，基于本章提出的 WavLM 多语种识别模型，在几个不同的信噪比下测试了不同加权因子对语种识别准确率的影响，具体的，以 0.05 作为加权因子的递增间隔，测试了在对应值下的 EER 和C_{avg}指标，并绘制了其在 0 到 1 区间内的变化曲线。如图 4-9 所示：

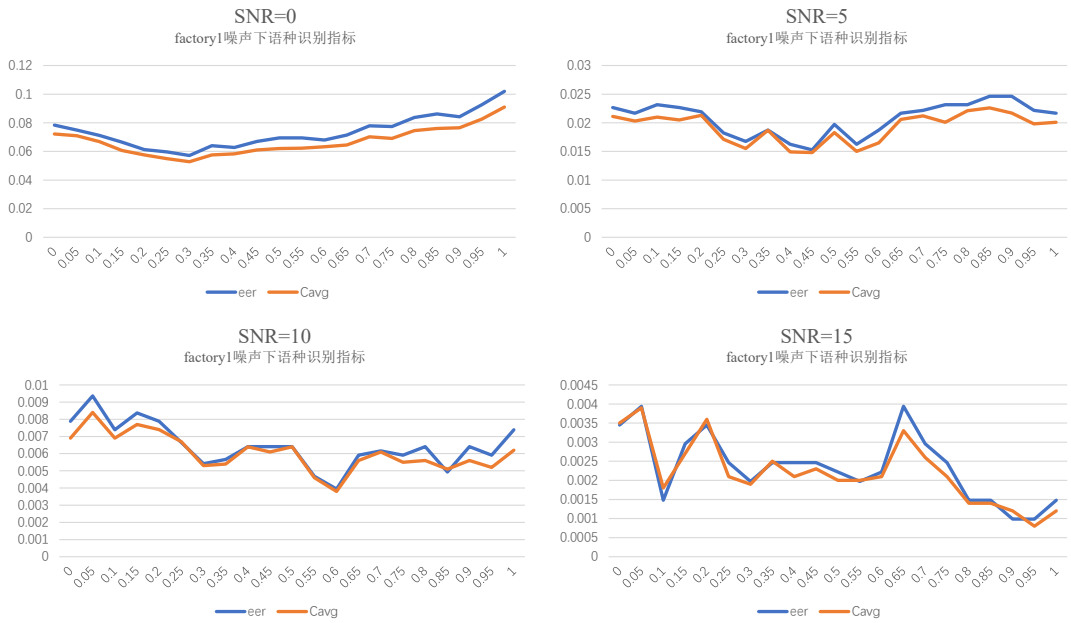


图 4-9 使用 factory1 噪声在不同信噪比环境下的语种识别 EER 和C_{avg}

图中横坐标为加权因子，纵坐标为指标数值。我们可以观察到，若直接使用增强语音进行测试，即加权因子为 0，其在 4 个信噪比条件下的 EER 分别为 7.83%、2.27%、0.79%和 0.34%，而直接使用带噪语音进行测试，其 EER 指标分别为 10.20%、2.17%、0.74%和 0.15%，在部分信噪比条件下，增强语音并没有带来正向的收益，反而降低了语种识别的性能。再对训练集中未出现的 white 噪声进行了测试，发现其加权因子对性能影响的规律和 factory1 噪声类似，如图 4-10 所示。

其中加权因子为 0.35、0.4、0.8 和 1.0 时取得最大，最优 EER 分别为 4.63%、1.48%、0.59%和 0.15%，最优C_{avg}分别为 4.38%、1.41%、0.6%和 0.12%。对于 factory1 噪声，当加权因子分别为 0.3、0.45、0.6 和 0.9 时取得最佳的性能，EER 分别为 5.71%、1.53%、0.39%和 0.098%，C_{avg}分别为 5.28%、1.48%、0.38%和 0.12%。均远高于直接使用带噪语音进行测试的结果，证明了使用加权因子对增强语音进行后处理能够有效的抑制增强带来的失真；结合语音增强对噪声的抑制，有效的提高下游语种识别任务的性能。本文选择 factory1 噪声中最好的加权因子作为所有噪声的加权因子，对 0 到 15dB 的信噪比范围内的语音进行了联合语音增强和语种识别的测试。

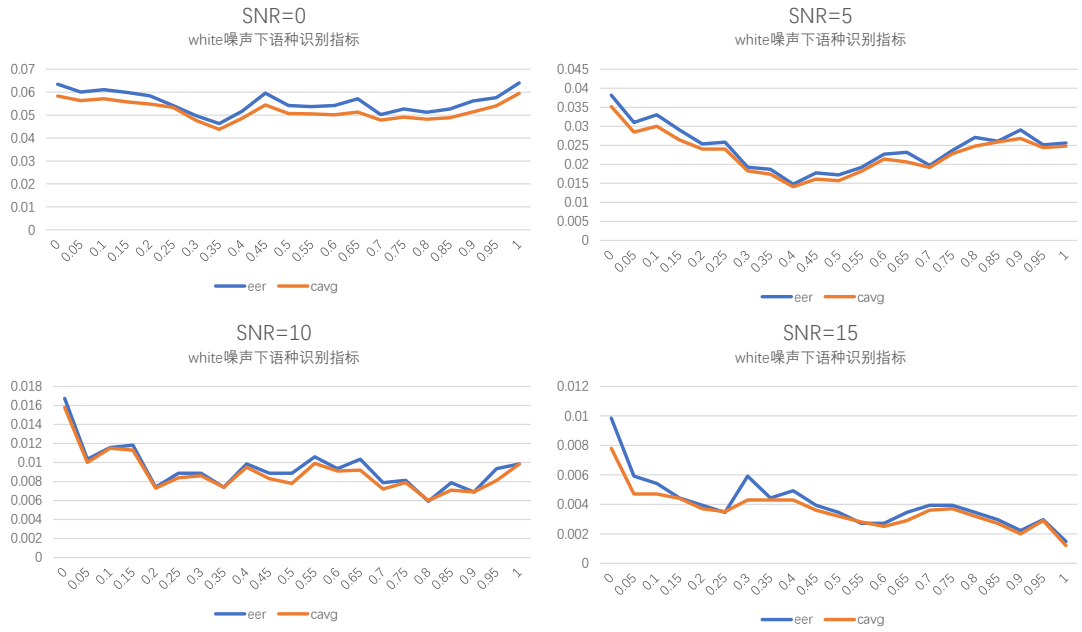


图 4-10 使用 white 噪声在不同信噪比环境下的语种识别 EER 和C_{avg}

从表 4-7 到表 4-10 我们可以看到，对语音增强后的语音使用本文提出的加权方法后，在各种信噪比条件下性能都有进一步的提升，即使在 0dB 的强噪声干扰下，本文提出的模型 WavLM+AM+LM 也有 4.95%的平均 EER 和 4.5%的平均C_{avg}，相较于其他基线模型，在不使用第三章提出的增强算法下，最优的 XLSR 模型平均 EER 为 21.53%，平均C_{avg}为 20.9%。而基于有监督的方法 Conformer+AM+LM 在所有信噪比条件下，对于 factory1 噪声，其平均 EER 为 15.59%，平均C_{avg}为 15.26%，远高于 ResNet 的 36.29%和 40.13%，证明了本文所提出的方法有效的提高了在面对复杂噪声环境时的语种识别准确率。

表 4-7 SNR=0 时语音增强联合语种识别 EER 和C_{avg}指标

噪声类型	可见噪声				不可见噪声			
	factory1		babble		white		factory2	
指标	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}
X-Vector	49.51	45.02	43.13	40.16	41.28	40.70	46.77	41.29
Resnet	38.79	41.28	36.11	39.47	35.62	39.91	33.94	36.09
Conformer	**29.31**	**29.01**	**27.98**	**27.77**	**20.89**	**21.11**	**16.11**	**16.12**
WavLM+AM	**4.68**	**5.34**	**5.71**	**6.34**	**4.24**	**4.87**	**1.03**	**1.14**
WavLM+LM	11.77	11.44	12.73	12.27	10.69	10.38	1.65	1.59
WavLM+AM+LM	6.40	5.76	7.93	7.40	4.63	4.38	0.84	0.81
XLSR	10.22	9.98	14.83	14.66	5.17	4.98	2.22	2.29

表 4-8 SNR=5 时语音增强联合语种识别 EER 和C_{avg}指标

噪声类型	可见噪声				不可见噪声			
	factory1		babble		white		factory2	
指标	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}
X-Vector	47.83	42.67	37.78	33.67	43.50	42.09	38.82	35.45
Resnet	35.71	42.38	33.30	31.23	38.13	47.05	27.49	24.88
Conformer	**17.39**	**16.64**	**13.55**	**13.68**	**15.57**	**15.91**	**7.27**	**7.35**
WavLM+AM	1.85	1.77	1.58	1.85	1.92	2.22	0.64	0.80
WavLM+LM	3.79	3.53	2.76	2.54	4.06	3.91	0.49	0.45
WavLM+AM+LM	**1.63**	**1.49**	**1.45**	**1.44**	**1.48**	**1.41**	**0.20**	**0.20**
XLSR	2.46	2.39	5.17	5.23	3.15	3.08	1.01	1.04

表 4-9 SNR=10 时语音增强联合语种识别 EER 和C_{avg}指标

噪声类型	可见噪声				不可见噪声			
	factory1		babble		white		factory2	
指标	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}
X-Vector	44.51	37.69	32.46	28.97	43.74	39.35	32.32	29.72
Resnet	35.47	39.35	28.28	25.44	38.79	46.56	20.99	19.17
Conformer	**10.30**	**9.99**	**6.01**	**6.11**	**12.07**	**11.80**	**4.04**	**4.17**
WavLM+AM	1.08	1.14	0.86	0.96	**0.91**	1.11	0.64	0.68
WavLM+LM	1.18	1.09	0.84	0.78	1.87	1.79	**0.15**	0.15
WavLM+AM+LM	**0.39**	**0.38**	**0.25**	**0.23**	0.94	**0.91**	0.17	**0.14**
XLSR	0.99	0.98	1.97	2.05	3.10	2.82	0.79	0.72

表 4-10 SNR=15 时语音增强联合语种识别 EER 和C_{avg}指标

噪声类型	可见噪声				不可见噪声			
	factory1		babble		white		factory2	
指标	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}	EER(%)	C_{avg}
X-Vector	39.11	33.27	27.66	24.86	40.30	35.87	28.82	26.53
Resnet	35.20	37.51	22.54	20.67	37.83	45.95	12.86	12.08
Conformer	**5.37**	**5.41**	**2.46**	**2.99**	**8.03**	**7.91**	**2.17**	**2.11**
WavLM+AM	0.49	0.52	0.57	0.59	0.74	0.85	0.42	0.56
WavLM+LM	0.34	0.34	0.34	0.33	0.94	0.86	0.17	0.14
WavLM+AM+LM	**0.10**	**0.12**	**0.10**	**0.12**	**0.22**	**0.20**	**0.00**	**0.01**
XLSR	0.39	0.37	0.94	1.00	0.64	0.57	0.47	0.48

4.3.3.4 语种识别与字错误率相关性

语音识别模型的评价指标是字错误率（Character Error Rate）或词错误率（Word Error Rate），对于中文，由于相邻字符之间没有空格分割，主要使用字错误率作为评估指标，而对于英语等包含空格分隔符的语种，其评判指标主要是词错误率，对于同一对文本，其 WER 往往高于 CER，并且在训练的初始阶段，WER 收敛速度较慢，为了更好的观察到训练过程中的性能变化，对于所有的语种，本文统一使用字错误率作为评价指标。在前面的实验中，基于 WavLM 的多语种语音识别模型在语种识别任务上具有最优的性能，值得注意的是其语音识别的字错误率也优于基于 Conformer 的有监督语音识别模型，因此字错误率和语种识别准确率可能存在某种联系。本文对字错误率和语种识别的准确率做了进一步探索。

对于本文提出的两个多语种语音识别模型，在训练过程中对验证集上的字错误率和语种识别的准确率进行测试，同时绘制其散点图，如图 4-11 和图 4-12 所示：

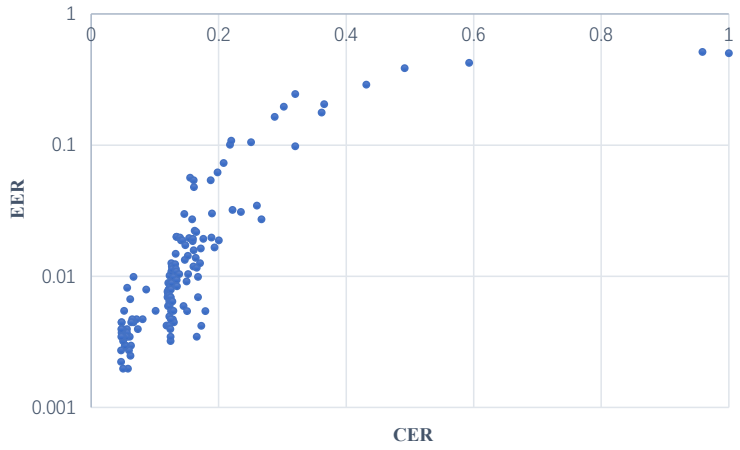


图 4-11 字错误率和语种 EER 散点图

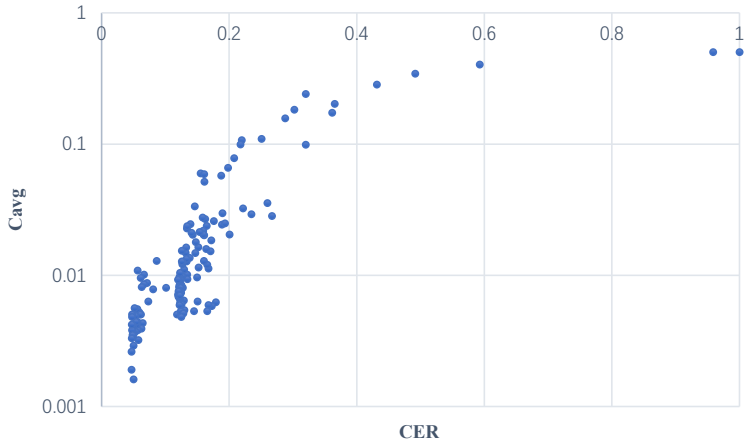


图 4-12 字错误率和语种C_{avg}散点图

我们可以观察到，虽然统计的是两个不同的模型在训练期间的结果，但 EER 和 CER 指标之间都表现出较强的相关性，随着语音识别字错误率的降低，语种识别的 EER 和C_{avg}指标也随之降低。语音识别的解码依赖其对字符的置信度，字错误率的降低表明语音识别模型学习到了该语种更多的先验知识，从而有利于语种的判别。我们也可以看到，对于具有更低的字错误率的模型，其 EER 更低，表明若进一步提升语音识别模型的性能，语种识别的准确率也将会随之提升。

4.4 本章小结

本章对语种识别模型进行了研究，基于 WavLM 预训练模型和 Conformer 模块，构建了一个多语种语音识别模型，并在多语种语音识别的基础上，通过对其后验输出进行建模，提出了一种基于词法和语法信息的语种识别方案。这种方案相较于传统端到端方案具有更好的泛化性能，在低资源语种的识别上优势明显。通过级联第三章提出的语音增强算法，进一步提升了在噪声环境下的语种识别效果。

第五章 面向民航中英文双语陆空通话的语种识别原型系统

本章主要基于前文提出的语音增强算法和语种识别算法，设计并实现面向民航机场嘈杂环境的陆空通话语种识别原型系统。该系统借鉴微服务架构思想，将算法模块封装成一个个微服务单元进行调用。系统主要包括面向复杂环境的语音增强和语种识别两大功能，用户可以使用浏览器进行访问并通过直接录音或音频文件的方式进行交互。同时针对基于深度学习的算法运行效率低下的问题，本文设计了一种基于消息批处理的任务调度框架，有效的提高了系统中接口的吞吐量。接下来从软件工程的角度，详细描述了系统的设计和测试过程。

5.1 需求分析

本章所设计的系统用于民航领域中塔台管制员和飞行员之间的陆空通话场景。近年来，为了更好的辅助塔台管制员进行客机的飞行管制，对可能的危险操作进行有效的预警，各大机场开始引入针对民航的中英文语音识别系统将语音转化为便于计算机处理的文本。但由于机场噪声干扰，语音识别等任务的准确率会大幅下降。并且机场会同时面临国内和国外的民航客机，飞行员说话的语种是未知的，因此要准确的对语音内容进行识别还需要语种识别系统的辅助。面对以上行业痛点问题，对需求进行了归纳：

（1）算法服务需求：本文设计的原型系统需要包含语音增强和语种识别两个核心功能，用于复杂多变的机场环境下的中英文双语语种识别。另外还需要用于去除静音帧的声音活性检测（Voice Activation Detection, VAD），减少系统无意义的计算消耗。为了对本文提出的方法进行验证，语音增强算法基于第三章所提出的模型，语种识别算法基于第四章提出的 WavLM+AM+LM 模型。对于静音检测算法，本文使用开源的 webrtcvad 库[①]并对其封装。

（2）访问方式需求：作为下游任务的前端，系统以基于 Restful 风格的 API 接口对外提供服务。为了便于演示，需要一个访问和交互较为便捷的网页作为系统界面，用户只需要通过一个浏览器，即可实现对系统的访问，从而可以通过便携式设备如手机平板等在任意地点使用本系统，使其具备了在各种声学场景下进行演示的条件。

（3）UI 界面需求：根据算法的实际应用场景同时兼顾算法的可视化分析，本

[①] https://github.com/wiseman/py-webrtcvad

文的系统需要具备以下功能模块：音频输入、算法处理、结果展示和音频播放，用户与这些独立的功能直接进行 UI 交互。在功能下有更细粒度的子功能划分，具体功能如表 5-1 所示。

（4）性能需求：考虑到民航机场场景下存在多个通信信道，可能同时有多个塔台管制员和飞行员之间的通话，因此本文的系统需要具备一定的并发能力，并且在并发的条件下保持系统的稳定。考虑到正常情况下，塔台管制员由小于 10 人的团队组成，因此设计的目标并发量为 10，且系统平均延迟低于 700ms。

表 5-1 系统功能划分

功能模块	一级子功能	二级子功能	说明
语音增强	音频输入	语音上传	本模块用于语音增强算法的效果演示，用户可以通过文件或直接录音的方式输入到系统，算法包括两个部分，VAD 和 SE。
		在线录制	
		语音增强算法	
	结果展示	原始音频展示	
		增强语音展示	
	音频播放	播放控制	
		进度控制	
语种识别	音频输入	语音上传	本模块对语音语种进行判别，支持两种语言的语种。同时结合语音增强算法可用于噪声条件下的语种识别。对外提供 Restful API。
		在线录制	
	算法处理	静音检测算法	
		语音增强算法	
		语种识别算法	
	结果展示	原始音频展示	
		增强语音展示	
	音频播放	播放控制	
		进度控制	

5.2 系统总体设计

本系统采用微服务架构设计，在算法层，不同的算法模块封装成一个个微服务，可独立进行开发部署，与其他算法模块没有依赖关系，应用层通过消息中间件进行远程过程调用（Remote Procedure Call, RPC）。具体的，算法层包含基于全卷积神经网络的语音增强算法、基于 webrtcvad 的静音检测、基于 WavLM 的语种识别算法。然后是中间件层，各个算法微服务模块通过消息队列中间件进行服务注册和发现。其次是应用层，提供 API 的应用层微服务通过消息队列的注册和发现机制连接算法微服务。从而完成远程消息调用过程，实现算法的访问。应用层提供每个算

法访问的 API 接口，具体的，提供语音识别算法、语音增强算法、静音检测算法接口。系统的最外层通过 Nginx 完成负载的均衡。来自客户端的 HTTP 请求首先经过请求分发网关 Nginx 服务器，再将请求转发给应用微服务。基于 SpringBoot 的应用服务收到请求后对其进行解析，将获取到音频二进制数据依次通过 Controller 层、Service 层发送给算法微服务进行处理，算法微服务依次经过算法服务层、模型层和消息处理层将处理得到的数据返回给消息队列，算法层处理完成后，应用微服务获取到算法结果，进行封装整合，从而完成应用服务层的处理，最后将结果返回给 API 接口。展示层是系统的前端界面，其用于接受用户的输入，并将算法返回结果进行实时绘制，其绘制过程基于浏览器的渲染引擎和 JavaScript 引擎的协作，通过对 CSS 样式文件、HTML 文件和 JavaScript 文件等进行解析得到。所有的服务均通过 docker 进行部署，可分布式的运行在不同的服务器上，从而很方便的在算法负载较大时进行水平和垂直的扩容。其总体框架如图 5-1 所示。

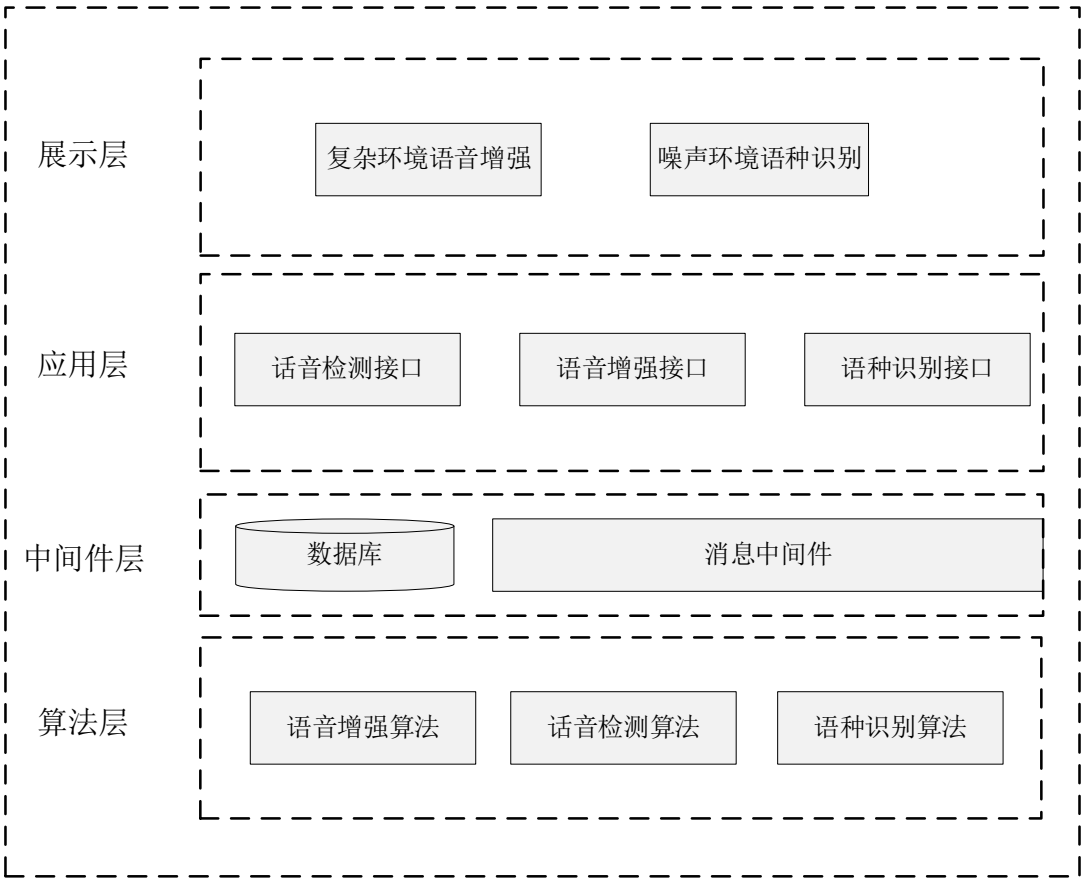


图 5-1 总体框架图

5.3 系统详细设计

系统根据核心算法共分为两大子系统：语音增强系统、语种识别系统。针对复

杂环境下的应用场景，联合声音活性检测、语音增强和语种识别设计了一个新的模块。接下来将基于顺序图对这些模块的处理流程进行详细分析。

5.3.1 语音增强子系统

语音增强子系统包括音频输入、算法处理、结果展示和音频播放等功能，其中除了核心的算法处理外，其余模块均由前端直接进行处理。对于语音增强算法调用的流程，分为如下几个步骤，首先用户在前端界面通过录音或者选择本地文件的方式录入待处理的语音，接下来通过 http 接口上传到应用层，在应用层中，控制器接收传入的音频参数，经过预处理后交给消息中间件，进入阻塞状态。语音增强算法绑定特定的消息队列，当监听到应用层有消息传入时进行处理，算法处理完成后通过一临时队列传给应用层，这时跳出阻塞状态，对返回的消息进行封装处理，将增强语音传回前端界面，前端收到接口返回信息后对其进行渲染，显示出增强前后的波形图。顺序图如图 5-2 所示：

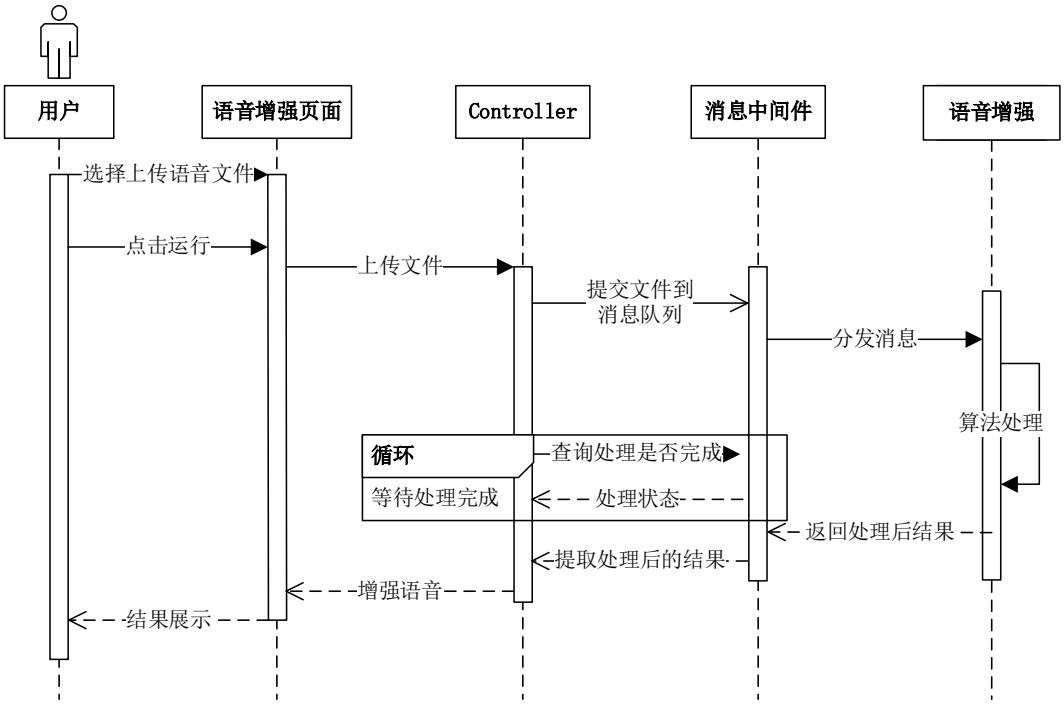


图 5-2 语音增强子系统顺序图

5.3.2 语种识别子系统

语种识别模块包括对基于联合语音识别先验信息的语种识别算法的封装，以及提供算法的使用接口和前端交互界面。该模块对算法进行微服务整合，语种识别算法独立的运行在 docker 容器中，通过向消息中间件进行队列绑定完成服务的注

册。在该模块中，用户通过应用服务器提供的 API 发送待识别的语音，算法端接收到语音文件后进行判别，将结果返回给消息队列中间件，应用服务器通过阻塞线程等待算法端结果，最后返回给 API 调用者。基于 HTML 的前端模块渲染待识别音频，和显示语种识别结果，用户可以播放音频和查看该音频的语谱图。顺序图如图 5-3 所示：

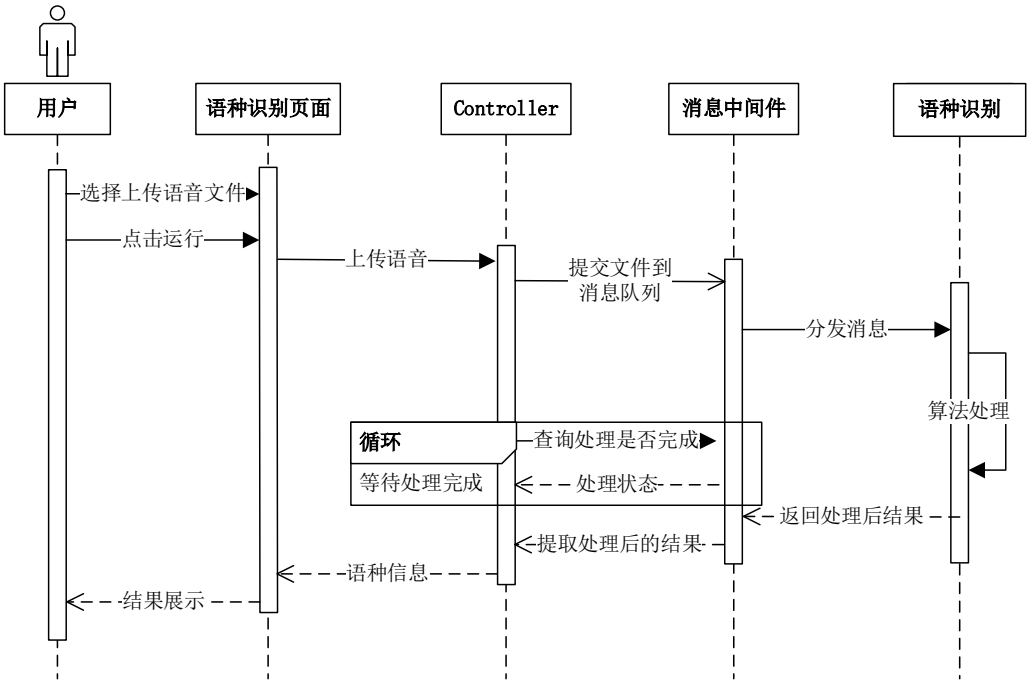


图 5-3 语种识别子系统顺序图

5.3.3 面向复杂环境的语种识别子系统

针对复杂噪声环境，本子系统采用将算法进行级联的方式进行调度。算法包括静音检测算法、语音增强算法和语种识别算法。和 5.3.2 节中的顺序图类似，应用层通过消息中间件的方式和算法服务进行通信，从而完成 RPC 调用。对于多个算法相互依赖的情况，通过在应用层进行调度来完成。具体的，首先语音交由 VAD 静音检测算法进行预处理，去除无意义的静音帧；应用层在拿到 VAD 模块返回结果后，再以相同的方式调用语音增强算法模块，最后再是语种识别模块，所有算法处理完成后将最终的结果进行封装并返回给前端渲染。这种调用方式降低了算法模块之间的耦合度，同时基于消息队列的调用方式可以允许存在多个消费者，从而可以在算法处理压力较大时分配消息给其他服务器进行分布式的处理，提高了系统的处理能力。基于消息队列的架构还可以进行流量削峰，在压力负载过大时提高系统的稳定性。顺序图如图 5-4 所示：

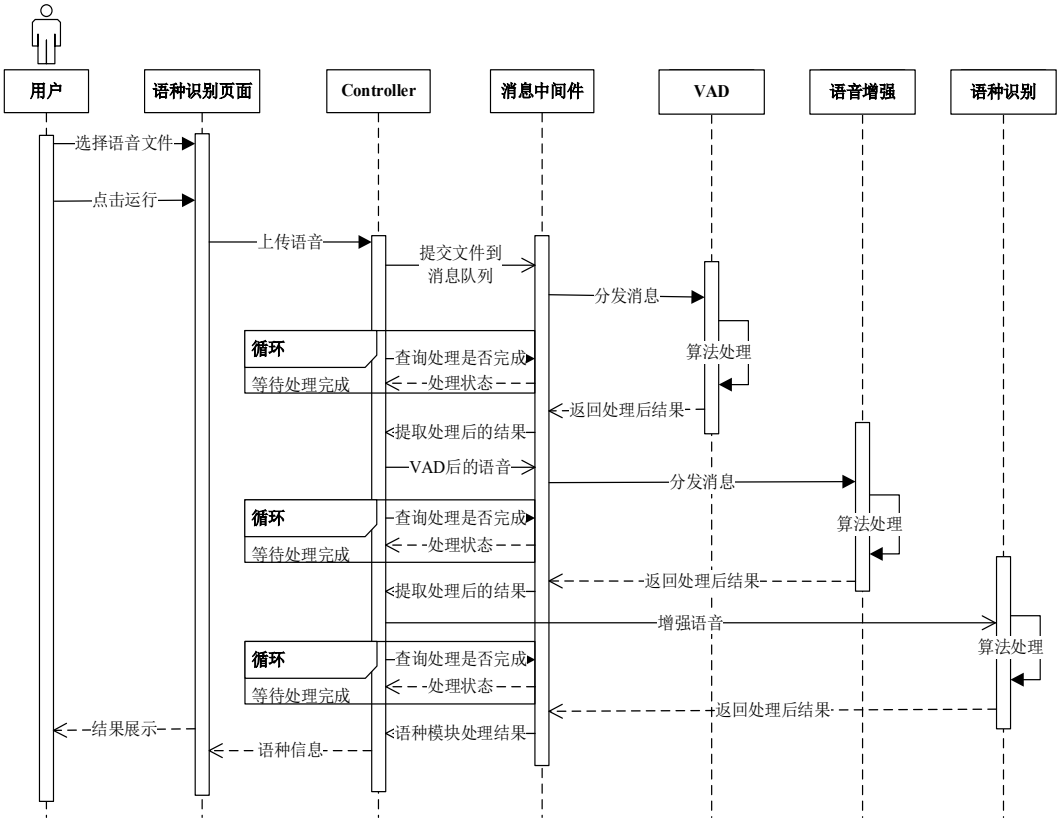


图 5-4 复杂环境下的语种识别子系统顺序图

5.4 系统性能优化

受益于显卡在并行计算上的巨大优势，基于深度学习的模型往往选择部署到具有 GPU 的服务器上，然而 GPU 显存带宽有限，导致难以同时部署多个服务以应对大量的并发请求。对于每个 http 请求，应用层都会单独开辟一个线程进行处理，由于应用层没有太多密集计算，因此线程消耗的系统资源有限。然而对于算法端，模型参数量较大，若每次请求到达后单独加载模型参数到内存中是较为耗时的，因此往往采用预加载模式。但受到显存大小限制，常驻内存的模型数量是受限的，算法端通常只能使用单线程的方式串行的接收消息队列传来的消息，这就导致了在高并发情景下，系统平均响应延迟的大幅增加。并且在这种串行方式下算法端难以充分利用 GPU 的并行能力，对服务器资源造成极大的浪费。

百度在 2016 年的 DeepSpeech2 中提到了一种批调度方法[96]，其通过将多个用户的请求数据流组装成一个批次进行处理，并使用一种动态调度策略，当前一个批次完成后立即处理下一个批次，这种方法被证明能有效的提高计算效率，同时降低了用户端到端延迟。本文受到其启发，设计了一种用于消息队列 RPC 调用的批处

理调度层，对来自消息中间件的消息进行打包，提高显卡的计算效率。和百度不同的是，考虑到同一接口中有多个不同算法协同处理，并没有使用相邻 batch 无缝衔接的计算模式，而是在相邻 batch 之间增加一个时间间隔，这个时间内其他算法模块可以更加高效的利用 GPU 资源，同时该模块可以收集到更多的消息，提高了批处理效率。使用该调度层前后算法的调用如图 5-5 所示：

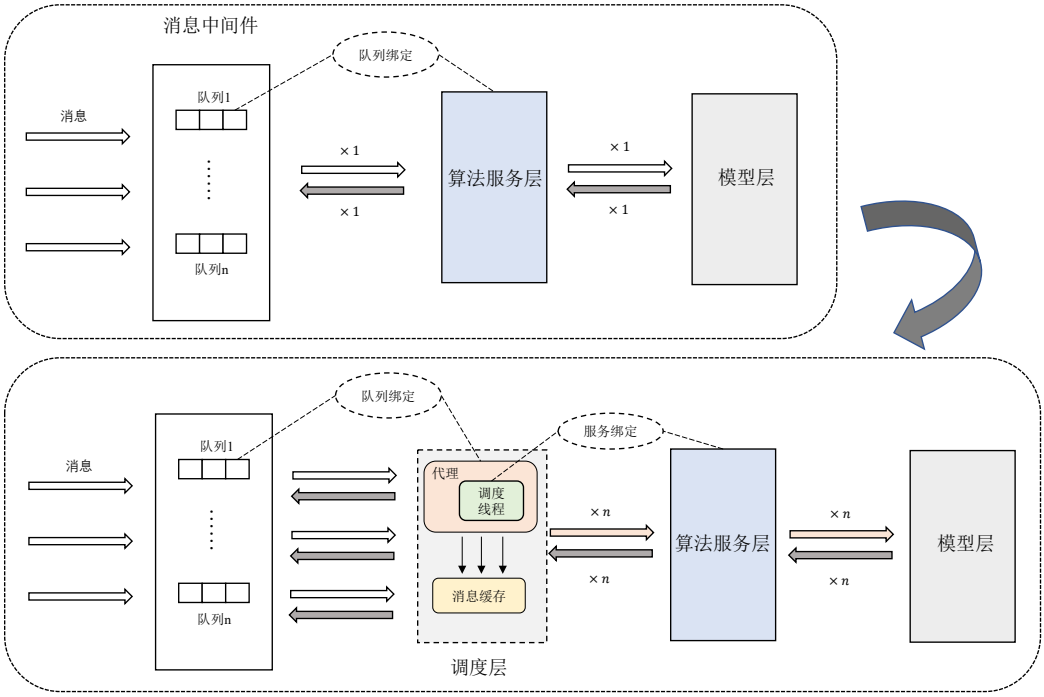


图 5-5 有无调度层算法调用流程对比

对于原有的串行方案，算法端由两部分组成：算法服务层和模型层。算法服务用于语音消息的预处理以及相关的业务逻辑，其和特定的消息队列进行绑定，从而接收生产者发来的消息，然而由于不能多线程并发的调用模型，算法服务同一时刻只能处理一个消息。模型层专注于算法的计算，算法服务层将消息处理成模型层需要的格式后由模型进行推理，然后返回结果给消息队列。可以看到，该架构下，算法服务一次只能处理一个消息，难以应对高并发的场景。

对于优化后的方案，在消息中间件后增加了一个消息调度层，算法端保持不变。调度层由三部分组成，服务代理、调度线程和消息缓存。服务代理用于替代原有算法服务层对消息队列的绑定，消息队列直接和服务代理进行交互，无论算法是否处理完成，都可接收多个消息，并将消息放入到消息缓存中。调度线程是一单独的线程，和算法服务进行了绑定，其使用一种调度策略来决定交给算法服务层消息的时间点和数量，具体的，当消息缓存中的数量达到某一阈值或当前时间和调度开始时

间差值超过了一次调度周期，调度线程将消息缓存中的消息交由算法服务层进行处理，和原有的方案相比，这里可以并行的处理多条消息。其中批大小 N 和调度周期 T 是一个可调节的超参数，调度层的队列大小是一个比批大小稍大的常数，可在 batch 容量满时缓冲下一个批次的语音，提高调度效率。基于本文的设计的调度层，对系统的性能进行了测试。

本文选用 Locust 压力测试框架，其基于 Python 运行环境编写测试用例，测试了不同并发线程数下，使用批处理层的前后性能，包括每秒请求数、接口平均延迟和 90%接口延迟百分位值 P90。由于批处理技术每次打包多个音频数据到模型中，因此会占用更多显存，为了公平的比较，对于原始方案，本文为每个算法启用两个实例。批处理层中批大小 N 选择为 10，调度周期为 40ms。其中测试环境的软硬件条件如下表所示：

表 5-2 测试服务器配置

软硬件名称	型号
CPU 型号	E5-2678 × 2
内存容量	16GB × 4
显卡	Nvidia 3090 24GB
显示器	分辨率 2560 × 1440
网络带宽	100M
操作系统	Ubuntu 20.04
Pytorch	1.8.0

测试结果如图 5-6 到图 5-8 所示：

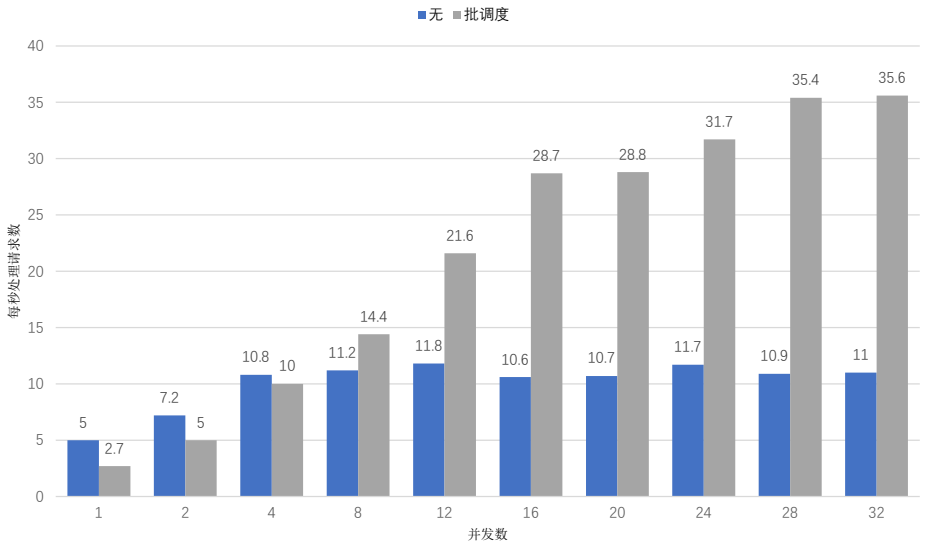


图 5-6 使用批调度前后每秒处理请求数对比

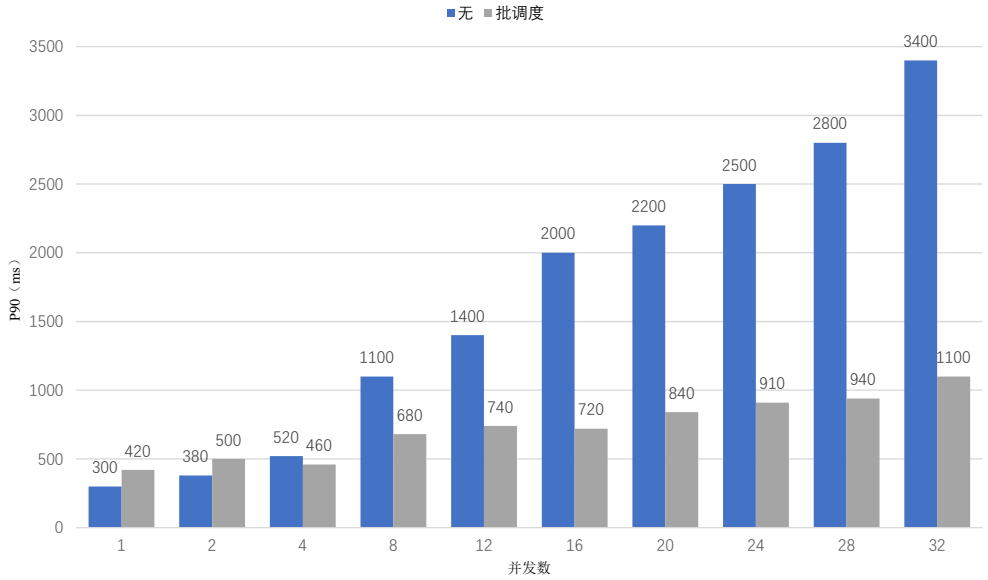


图 5-7 使用批调度前后 P90 指标对比

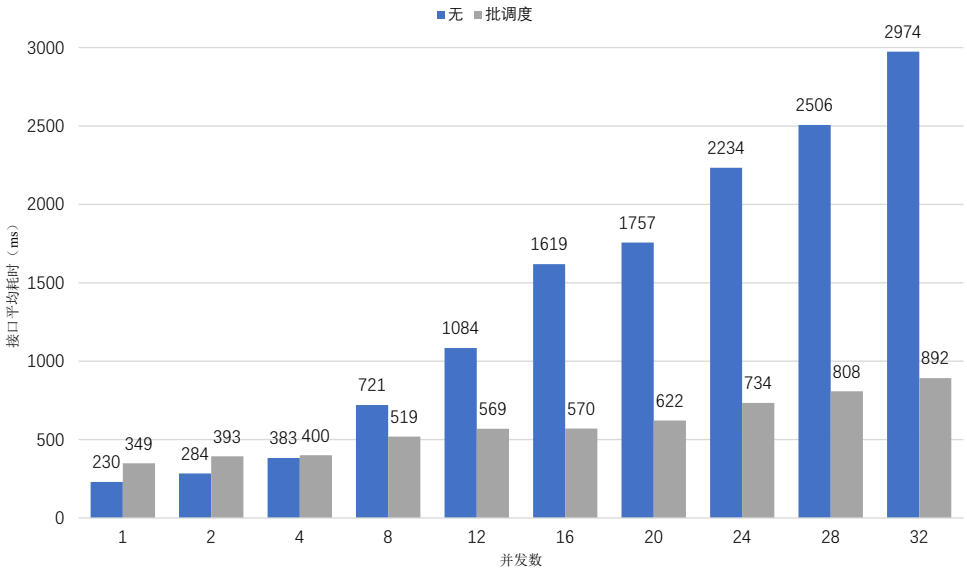


图 5-8 使用批调度前后接口平均延迟对比

可以看到，随着并发数的增加，系统逐渐到达性能极限，对于使用批调度的方案，在并发数为 30 左右时达到饱和，每秒请求数为 35 个，而对比原有的方案，其在并发数为 8 时就达到了极限，每秒请求数为 11 个，性能提升了两倍。对于接口平均耗时和前 90%接口耗时指标，随着并发数的增加逐渐升高，然而原有方案在大于 16 并发数时其接口延迟和 P90 分别达到了 1.6 秒和 2 秒，对于用户来说结果等待时间较长，实际体验较为卡顿，而优化后的方案仅为 570 毫秒和 720 毫秒，即使并发数达到 32，其接口延迟也大幅低于原有的方案，证明了调度层的有效性。

在较低的并发量下，由于调度层引入的等待周期，每个算法平均会增加 40ms 左右的延迟，因此在这种场景下，原有方案较优。然而在 200ms 基础上增加的 100ms 的接口响应延迟时可接受的，因为本文的方案在高并发场景下带来的性能提升收益远高于低并发场景下增加的延迟损失，并且可以使得系统更加稳定。

5.5 技术框架

本系统主要基于 B/S 的软件设计模式，应用服务器端采用 SpringBoot 框架开发，其通过 Controller+Service+DAO 的基本架构设计构建接口。前端采用 HTML+CSS+JavaScript 开发语言构建系统软件页面，同时使用 Bootstrap 和 Jquery 前端框架辅助 UI 组件的设计，前后端分离独立开发.算法端基于 Pytorch 和 Tensorflow 深度学习框架，消息中间件使用 RabbitMQ，请求分发网关使用 nginx，浏览器通过 HTTP 请求与服务器端建立连接。使用的技术框架如图 5-9 所示：

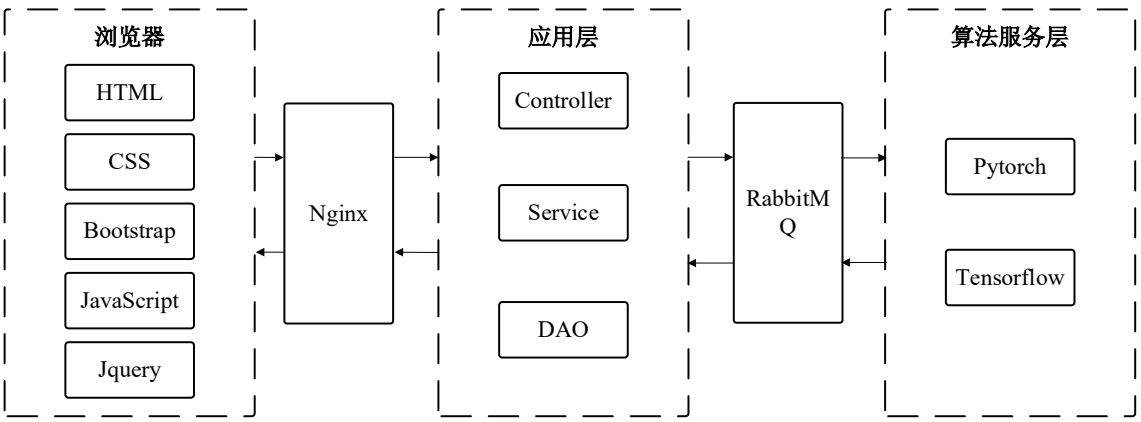


图 5-9 系统技术架构设计

5.6 系统测试

本部分详细介绍了系统功能界面的 UI 设计。为了保证系统能够正常使用，和对语音增强、语种识别和联合三个算法的复杂环境语种识别功能模块的进行了测试。

5.6.1 语音增强

语音增强算法界面如图 5-10 所示，用户点击语音增强模块，进入算法演示界面，用户选择含噪音频文件上传的“选择文件”按钮，选择需要处理的音频文件（格式为 wav），选中后，点击右侧的上传按钮。然后用户选择干净音频文件上传的“浏览”按钮，选择需要处理的音频文件（格式为 wav），选中后，点击右侧的上传按钮，

即可启动该算法进行处理。用户也可以点击麦克风录音上传的 Record 按钮，即可启动麦克风，开始录音，点击 stop 停止录音，然后再点击下方的“上传”按钮就可以执行该算法。UI 界面最终显示原始波形图、原始语音频谱图、增强语音波形图和增强语音频谱图。

图 5-10 语音增强界面

可以看到增强后的语音频谱结构更加清晰，表明本算法有效的抑制了噪声。为了保证该系统的稳定运行，对本模块进行粒度更低的测试，测试用例表如下表所示：

表 5-3 语音增强模块测试用例表

序号	测试对象	测试方案	期望测试结果	实际测试结果
1	页面展示	浏览器进入页面	UI 显示正常	与期望一致，测试通过
2	文件上传	点击上传，选择 wav 文件	正常识别	与期望一致，测试通过
3	文件上传	点击上传，选择 png 文件	拒绝识别	与期望一致，测试通过
4	文件上传	点击上传，选择损坏的 wav 文件	系统提示异常	与期望一致，测试通过
5	算法	选择一带噪语音文件上传	噪音被去除	与期望一致，测试通过
6	算法	选择一干净语音文件上传	语音播放正常	与期望一致，测试通过
7	录音	选择录音上传实时语音	录制成功	与期望一致，测试通过

5.6.2 语种识别

语种识别算法界面如图 5-11 所示。其输入和语音增强模块类似，都是通过上传文件或录音的方式。本文的算法基于语音识别模型，因此除了语种识别结果外，还额外输出了文本结果。

图 5-11 语种识别界面

语音波形和频谱都正确的显示在界面上，语种结果输出正常，证明了本算法的有效性。为了保证该系统的稳定运行，对语种识别模块进行了粒度更低的测试，测试用例表如下表所示：

表 5-4 语种识别模块测试用例表

序号	测试对象	测试方案	期望测试结果	实际测试结果
1	页面展示	浏览器进入页面	UI 显示正常	与期望一致，测试通过
2	文件上传	点击上传，选择 wav 文件	正常识别	与期望一致，测试通过
3	文件上传	点击上传，选择 png 文件	拒绝识别	与期望一致，测试通过
4	文件上传	点击上传，选择损坏的 wav 文件	系统提示异常	与期望一致，测试通过
5	算法	选择一语音文件上传	识别结果正常	与期望一致，测试通过
6	算法	选择非该语种语音测试	系统无异常	与期望一致，测试通过

5.6.3 复杂环境的语种识别

面向复杂环境的语种识别界面如图 5-12 所示。本模块中，算法包含三个，静音检测、语音增强和语种识别，因此界面上包含语种识别和语音增强的综合结果，其使用步骤和上面提到的模块类似。

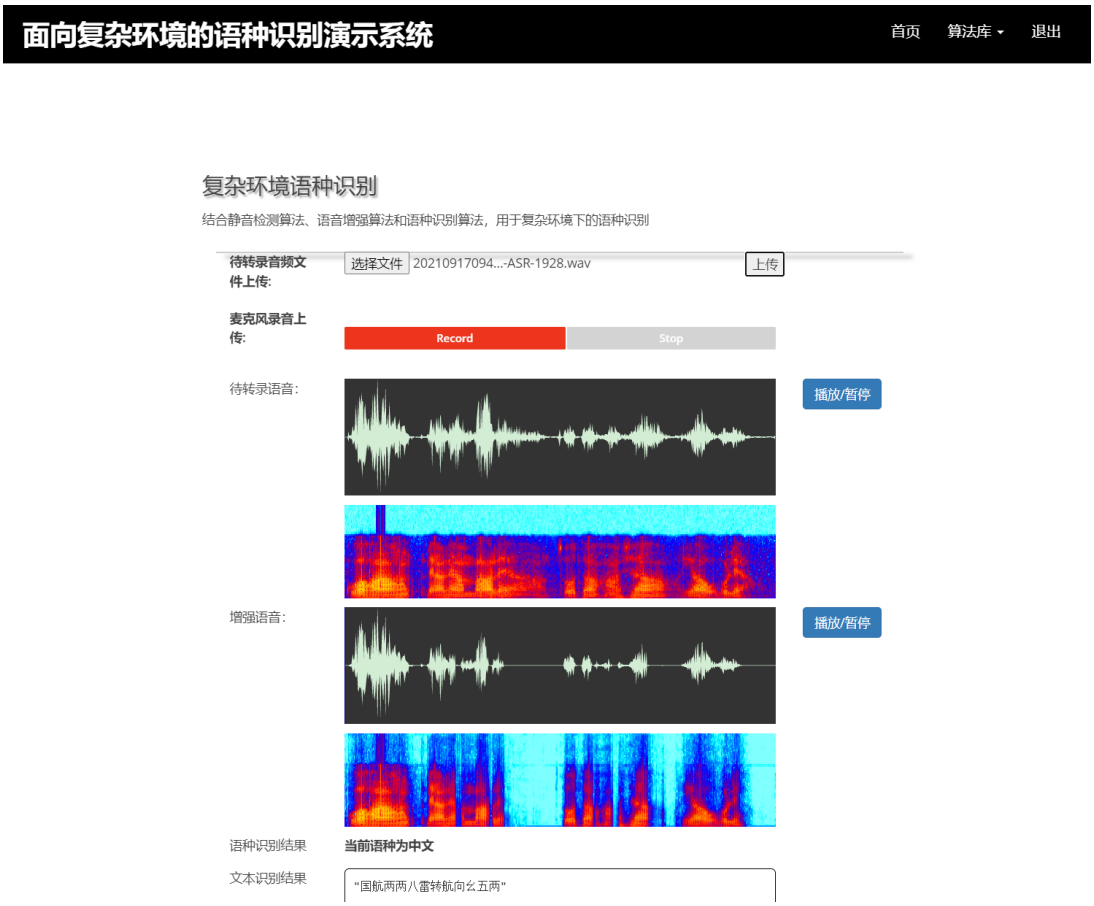


图 5-12 面向复杂环境的语种识别界面

再对该模块进行测试，测试用例表如表 5-5 所示：

表 5-5 语种识别模块测试用例表

序号	测试对象	测试方案	期望测试结果	实际测试结果
1	页面展示	浏览器进入页面	UI 显示正常	与期望一致，测试通过
2	文件上传	点击上传，选择 wav 文件	正常识别	与期望一致，测试通过
3	文件上传	点击上传，选择 png 文件	拒绝识别	与期望一致，测试通过
4	算法	选择含噪语音文件上传	识别结果正常	与期望一致，测试通过
5	算法	选择干净语音文件上传	识别结果正常	与期望一致，测试通过
6	算法	选择非该语种语音测试	系统无异常	与期望一致，测试通过

该模块中，语音首先通过 VAD 算法进行静音的去除，随后依次通过语音增强算法和语种识别算法，因此 UI 界面上包含了语音增强和语种识别的结果，从频谱图中可以看到，增强语音后的噪声分量明显减少，语音结构更加清晰，而这种噪声更少的的语音显然有助于语种识别算法的判别。

5.7 本章小结

本章设计了一种面向民航陆空通话的语种识别系统，基于微服务架构，通过消息中间件 RabbitMQ 实现各个组件的通信，保证了服务的稳定运行。该系统能够在一定噪声干扰下对语种进行准确的判别，同时为了进一步提升系统的性能，本文设计了一种批处理框架，在高达 32 个并发量的条件下系统也能够快速响应。

第六章 总结与展望

语种识别技术作为在多语种条件下人机交互的关键技术，关系到下游任务例如语音识别技术的实际应用。在真实的应用场景下，噪声种类多变、信噪比未知以及声学环境的复杂多样导致常规语种识别模型的性能下降。本文利用语言学特征对语种进行建模，并将语音增强算法和语种识别算法进行了结合，从整体上提升了在复杂环境下的语种识别性能。最后基于本文的算法模型，设计了一个面向民航的语种识别原型系统。本章将对本文的算法模型进行总结，以及对进一步的研究进行展望。

6.1 全文总结

本文主要研究了端到端语音增强算法和语种识别算法。以及他们的有机整合，提升了在复杂环境下的语种识别性能。对于语音增强算法，其任务是通过深度神经网络，训练带噪语音谱到干净语音谱的映射关系，从而提升带噪语音的感知质量和语音可懂度。通过对现有基于卷积神经网络的端到端语音增强算法进行深入研究和分析，发现其存在以下 2 个不足：首先，卷积神经网络由于其本身属性，只能使用固定大小的卷积核，导致模型感受野大小固定，然而不同噪声类型以及不同的信噪比条件下语谱图变化差异大，固定大小的感受野很难适应所有的场景，从而存在性能瓶颈。其次，模型中间层输出的特征图不受到损失函数约束，导致其难以有效的提取干净语音和带噪语音的特征。对于语种识别算法，通常作为一个分类任务，基于端到端直接建模，然而对于一些低资源语种，由于数据集的缺乏，会导致模型泛化性极差。当训练集和测试集条件不匹配时，性能会急剧下降。

本文围绕上述问题进行了深入探索，提出了对应的研究方案：

（1）在 RCNA 语音增强算法的基础上，对编码层进行了改进，将 SK 机制引入到语音增强，通过使模型动态的调整具有不同感受野大小的卷积分支的权重，从而使得模型具有自适应不同噪声环境的能力。实验结果证明，所提出的算法能够有效的提升模型的性能。

（2）将中继监督机制引入到语音增强算法中，对模型中间解码层输出的特征图进行约束，通过 SE 机制对特征图的注意力权重，选择具有最高权重的特征图作为预测的干净语音谱，选择权重最低的特征图作为预测的噪声谱，通过最小化他们和目标值之间的距离，并把这个距离作为附加的优化目标添加到原始损失函数中，对模型进行了多目标优化，通过和基准模型对比，证明了算法的有效性。

（3）提出了一种利用词法和语法特征的语种识别算法。基于无监督预训练模型，为每个语种构建了一个对应的 Conformer 块，首先通过语音识别任务进行端到端训练，然后通过分析语音在每个语种上的置信概率，同时结合语言模型对文本进行打分。该方法能够充分利用语种语言学信息，实验结果表明，本文所提出方法比起传统的端到端方案具有更好的鲁棒性和泛化性。

（4）在算法的研究基础上，本文设计了一个用于民航复杂环境下的语种识别系统。系统基于微服务架构，通过消息中间件进行各个组件之间的通信，系统能够稳定的提供服务。开发的原型通过 Web 浏览器可以很方便的进行访问。为了提升系统在高并发条件下的性能，本文还提出了一种批处理框架，通过对消息的调度，有效的提高了系统的吞吐量。

6.2 研究展望

目前基于深度学习的语音增强已经遇到了瓶颈，在面对未知噪声、未知信噪比以及未知说话人时很难进一步大幅度提升模型的性能。同时真实环境更为复杂，现实中的噪声不仅仅局限在加性噪声上，并且还面临信号失真的问题，这会导致下游任务的语种识别性能下降。本文中提出算法虽然在一定程度上提升了带噪场景下语种识别的性能，然而在面对信噪比极低以及存在失真的语音时可能仍然会失效。

近年来，基于多声道的语音处理技术进入了研究者的视野，多声道语音算法可以利用语音的空间角度信息，不同通道之间可以进行信息的补偿弥补，从而更进一步提升算法的鲁棒性。然而下游任务例如语音识别和语种识别等往往基于单声道语音，因此在语音增强前端和下游任务后端之间存在一定的鸿沟，如何更加高效的利用多声道优势辅助下游任务值得进一步的关注。

致　谢

研究生生涯即将结束，在这三年里收获颇丰。首先要感谢的是我的导师蓝天教授，在生活和学习上他都给予了我极大的帮助，从论文的开题到正文撰写，提出了很多宝贵意见，他精益求精的治学态度对我的个人成长产生了很大的影响。

还要感谢的是我的师兄师姐们，是他们的孜孜不倦的指导，让我能够克服科研路上的困难，并一起完成了多个科研项目。也要感谢同门师妹师弟们，是你们认真的态度和踏实的作风让我们能够一起的面对并解决项目中的遇到的难题。

最后感谢我的家人们，是你们的默默支持支撑着我的学业。

参考文献

[1] Li H, Ma B, Lee K A. Spoken language recognition: from fundamentals to practice [J]. Proceedings of the IEEE, 2013, 101(5): 1136-1159.

[2] Tu Y-H, Du J, Lee C-H. Speech enhancement based on teacher–student deep learning using improved speech presence probability for noise-robust speech recognition [J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2019, 27(12): 2080-2091.

[3] Weninger F, Erdogan H, Watanabe S, et al. Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR [C]. Proc of the Latent Variable Analysis and Signal Separation: 12th International Conference, LVA/ICA. Springer, August 25-28, 2015, 2015: 91-99.

[4] Lai Y-H, Zheng W-Z. Multi-objective learning based speech enhancement method to increase speech quality and intelligibility for hearing aid device users [J]. Biomedical Signal Processing and Control, 2019, 48: 35-45.

[5] Affonso E T, Rodrigo D Nunes, Rosa R L, et al. Speech quality assessment in wireless voip communication using deep belief network [J]. IEEE Access, 2018, 6: 77022-77032.

[6] Araki S, Hayashi T, Delcroix M, et al. Exploring multi-channel features for denoising-autoencoder-based speech enhancement [C]. Proc of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2015: 116-120.

[7] Lu Y-J, Cornell S, Chang X, et al. Towards low-distortion multi-channel speech enhancement: The ESPNet-SE submission to the L3DAS22 challenge [C]. Proc of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2022: 9201-9205.

[8] Tawara N, Kobayashi T, Ogawa T. Multi-Channel Speech Enhancement Using Time-Domain Convolutional Denoising Autoencoder [C]. Proc of the INTERSPEECH. 2019: 86-90.

[9] Udrea R M, Vizireanu N D, Ciochina S. An improved spectral subtraction method for speech enhancement using a perceptual weighting filter [J]. Digital Signal Processing, 2008, 18(4): 581-587.

[10] Paliwal K, Wójcicki K, Schwerin B. Single-channel speech enhancement using spectral subtraction in the short-time modulation domain [J]. Speech communication, 2010, 52(5): 450-475.

[11] El-Fattah M A, Dessouky M I, Diab S, et al. Speech enhancement using an adaptive wiener filtering approach [J]. Progress In Electromagnetics Research M, 2008, 4: 167-184.

[12] Uemura Y, Takahashi Y, Saruwatari H, et al. Musical noise generation analysis for noise reduction methods based on spectral subtraction and MMSE STSA estimation [C]. Proc of the 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2009: 4433-4436.

[13] Fan H-T, Hung J-w, Lu X, et al. Speech enhancement using segmental nonnegative matrix factorization [C]. Proc of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014: 4483-4487.

[14] Mohammadiha N, Smaragdis P, Leijon A. Supervised and unsupervised speech enhancement using nonnegative matrix factorization [J]. IEEE Transactions on Audio, Speech, and Language Processing, 2013, 21(10): 2140-2151.

[15] 白志刚, 鲍长春. 在线更新噪声基矩阵的非负矩阵分解语音增强方法 [J]. Journal of Signal Processing, 2020, 36(6).

[16] Ephraim Y, Malah D. Speech enhancement using a minimum mean-square error log-spectral amplitude estimator [J]. IEEE transactions on acoustics, speech, and signal processing, 1985, 33(2): 443-445.

[17] Ephraim Y. Statistical-model-based speech enhancement systems [J]. the Journal of the Acoustical Society of America, 1992, 80(10): 1526-1555.

[18] Xu Y, Du J, Li-Rong Dai, et al. An experimental study on speech enhancement based on deep neural networks [J]. IEEE Signal processing letters, 2013, 21(1): 65-68.

[19] Hinton G, Deng L, Yu D, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups [J]. IEEE Signal processing magazine, 2012, 29(6): 82-97.

[20] Xu Y, Du J, Dai L-R, et al. A regression approach to speech enhancement based on deep neural networks [J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2014, 23(1): 7-19.

[21] Weninger F, Erdogan H, Watanabe S, et al. Speech enhancement with LSTM recurrent neural networks and its application to noise-robust ASR [C]. Proc of the Latent Variable Analysis and Signal Separation, Liberec, Czech Republic. Springer, August 25-28, 2015: 91-99.

[22] Medsker L R, Jain L. Recurrent neural networks [J]. Design and Applications, 2001, 5: 64-67.

[23] Hasannezhad M, Ouyang Z, Zhu W-P, et al. An integrated CNN-GRU framework for complex ratio mask estimation in speech enhancement [C]. Proc of the 2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). IEEE, 2020: 764-768.

[24] Park S R, Lee J. A fully convolutional neural network for speech enhancement [J]. arXiv:160907132, 2016.

[25] Tan K, Wang D. A convolutional recurrent neural network for real-time speech enhancement [C]. Proc of the Interspeech. 2018: 3229-3233.

[26] Paliwal K, Wójcicki K, Shannon B. The importance of phase in speech enhancement [J]. speech communication, 2011, 53(4): 465-494.

[27] Tan K, Wang D. Complex spectral mapping with a convolutional recurrent network for monaural speech enhancement [C]. Proc of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019: 6865-6869.

[28] Hu Y, Liu Y, Lv S, et al. DCCRN: Deep complex convolution recurrent network for phase-aware speech enhancement [J]. arXiv preprint arXiv:200800264, 2020.

[29] Reddy C K, Gopal V, Cutler R, et al. The interspeech 2020 deep noise suppression challenge: Datasets, subjective testing framework, and challenge results [J]. arXiv preprint arXiv:200513981, 2020.

[30] Defossez A, Synnaeve G, Adi Y. Real time speech enhancement in the waveform domain [J]. arXiv preprint arXiv:200612847, 2020.

[31] Luo Y, Chen Z, Yoshioka T. Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation [C]. Proc of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020: 46-50.

[32] Pandey A, Wang D. TCNN: Temporal convolutional neural network for real-time speech enhancement in the time domain [C]. Proc of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019: 6875-6879.

[33] Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models [J]. Advances in Neural Information Processing Systems, 2020, 33: 6840-6851.

[34] Lugmayr A, Danelljan M, Romero A, et al. Repaint: Inpainting using denoising diffusion probabilistic models [C]. Proc of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 11461-11471.

[35] Huang R, Lam M W, Wang J, et al. Fastdiff: A fast conditional diffusion model for high-quality speech synthesis [J]. arXiv preprint arXiv:220409934, 2022.

[36] Popov V, Vovk I, Gogoryan V, et al. Grad-tts: A diffusion probabilistic model for text-to-speech [C]. Proc of the International Conference on Machine Learning. PMLR, 2021: 8599-8608.

[37] Lu Y-J, Wang Z-Q, Watanabe S, et al. Conditional diffusion probabilistic model for speech enhancement [C]. Proc of the IEEE International Conference on Acoustics, Speech and Signal

Processing (ICASSP). IEEE, 2022: 7402-7406.

[38] Lounnas K, Satori H, Hamidi M, et al. CLIASR: a combined automatic speech recognition and language identification system [C]. Proc of the international conference on innovative research in applied science, engineering and Technology (IRASET). IEEE, 2020: 1-5.

[39] Ramus F, Mehler J. Language identification with suprasegmental cues: A study based on speech resynthesis [J]. The Journal of the Acoustical Society of America, 1999, 105(1): 512-521.

[40] Schultz T, Rogina I, Waibel A. LVCSR-based language identification [C]. Proc of the IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings. IEEE, 1996: 781-784.

[41] Lim B P, Li H, Chen Y. Language identification through large vocabulary continuous speech recognition [C]. Proc of the International Symposium on Chinese Spoken Language Processing. IEEE, 2004: 49-52.

[42] Hieronymus J L, Kadambe S. Robust spoken language identification using large vocabulary speech recognition [C]. Proc of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing. IEEE, 1997: 1111-1114.

[43] Xiong W, Droppo J, Huang X, et al. Achieving human parity in conversational speech recognition [J]. arXiv preprint arXiv:161005256, 2016.

[44] Okamoto T, Hiroe A, Kawai H. Reducing latency for language identification based on large-vocabulary continuous speech recognition [J]. Acoustical Science and Technology, 2017, 38(1): 38-41.

[45] Kukk K, Alumäe T. Improving Language Identification of Accented Speech [J]. arXiv preprint arXiv:220316972, 2022.

[46] Lin C-Y, Wang H-C. Language identification using pitch contour information in the ergodic Markov model [C]. Proc of the IEEE International Conference on Acoustics Speech and Signal Processing Proceedings. IEEE, 2006: I-I.

[47] Ng R W, Leung C-C, Lee T, et al. Prosodic attribute model for spoken language identification [C]. Proc of the IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2010: 5022-5025.

[48] Ng R W, Lee T, Leung C-C, et al. Spoken language recognition with prosodic features [J]. IEEE Transactions on Audio, Speech, and Language Processing, 2013, 21(9): 1841-1853.

[49] Tong R, Ma B, Zhu D, et al. Integrating acoustic, prosodic and phonotactic features for spoken language identification [C]. Proc of the 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings. IEEE, 2006: I-I.

[50] Ng R W, Leung C-C, Hautamäki V, et al. Towards long-range prosodic attribute modeling for language recognition [C]. Proc of the Eleventh Annual Conference of the International Speech Communication Association. 2010,

[51] McCree A, Richardson F, Singer E. Beyond frame independence: parametric modelling of time duration in speaker and language recognition [C]. Proc of the InterSpeech. 2008: 767-770.

[52] Torres-Carrasquillo P A, Reynolds D A, Deller J R. Language identification using Gaussian mixture model tokenization [C]. Proc of the IEEE international conference on acoustics, speech, and signal processing. IEEE, 2002: I-757-I-760.

[53] Moselhy A M, Abdelnaiem A A. LPC and MFCC performance evaluation with artificial neural network for spoken language identification [J]. International Journal of Signal Processing, Image Processing and Pattern Recognition, 2013, 6(3): 55.

[54] Liu W-W, Cai M, Yuan H, et al. Phonotactic language recognition based on DNN-HMM acoustic model [C]. Proc of the The 9th International Symposium on Chinese Spoken Language Processing. IEEE, 2014: 153-157.

[55] Chen K-Y, Tsai C-P, Liu D-R, et al. Completely Unsupervised Phoneme Recognition by a Generative Adversarial Network Harmonized with Iteratively Refined Hidden Markov Models [C]. Proc of the Interspeech. 2019: 1856-1860.

[56] Zissman M A, Singer E. Automatic language identification of telephone speech messages using phoneme recognition and n-gram modeling [C]. Proc of the IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 1994: I/305-I/308 vol. 301.

[57] Gauvain J-L, Lamel L. Identification of non-linguistic speech features [C]. Proc of the Human Language Technology, Plainsboro, New Jersey. March 21-24, 1993,

[58] Noda J J, Travieso-González C M, Sánchez-Rodríguez D, et al. Acoustic classification of singing insects based on MFCC/LFCC fusion [J]. Applied Sciences, 2019, 9(19): 4097.

[59] Qi Z, Ma Y, Gu M. A study on low-resource language identification [C]. Proc of the 2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). IEEE, 2019: 1897-1902.

[60] Reynolds D A, Quatieri T F, Dunn R B. Speaker verification using adapted Gaussian mixture models [J]. Digital signal processing, 2000, 10(1-3): 19-41.

[61] Shen W, Reynolds D. Improved GMM-based language recognition using constrained MLLR transforms [C]. Proc of the 2008 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2008: 4149-4152.

[62] Zhu D, Ma B, Li H. Soft margin estimation of gaussian mixture model parameters for spoken

language recognition [C]. Proc of the 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2010: 4990-4993.

[63] Burget L, Matejka P, Cernocky J. Discriminative training techniques for acoustic language identification [C]. Proc of the 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings. IEEE, 2006: I-I.

[64] Dehak N, Torres-Carrasquillo P A, Reynolds D, et al. Language recognition via i-vectors and dimensionality reduction [C]. Proc of the Twelfth annual conference of the international speech communication association. 2011,

[65] Lopez-Moreno I, Gonzalez-Dominguez J, Plchot O, et al. Automatic language identification using deep neural networks [C]. Proc of the 2014 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2014: 5337-5341.

[66] Snyder D, Garcia-Romero D, McCree A, et al. Spoken language recognition using x-vectors [C]. Proc of the Odyssey. 2018: 105-111.

[67] Garcia-Romero D, McCree A. Stacked Long-Term TDNN for Spoken Language Recognition [C]. Proc of the Interspeech. 2016: 3226-3230.

[68] Bartz C, Herold T, Yang H, et al. Language identification using deep convolutional recurrent neural networks [C]. Proc of the International conference on neural information processing. Springer, 2017: 880-889.

[69] Vuddagiri R K, Vydana H K, Vuppala A K. Curriculum learning based approach for noise robust language identification using DNN with attention [J]. Expert Systems with Applications, 2018, 110: 290-297.

[70] 邵玉斌, 刘晶, 龙华, et al. 面向真实噪声环境的语种识别 [J]. 北京邮电大学学报, 2021: 1-6.

[71] Hou W, Dong Y, Zhuang B, et al. Large-Scale End-to-End Multilingual Speech Recognition and Language Identification with Multi-Task Learning [C]. Proc of the INTERSPEECH. 2020: 1037-1041.

[72] Watanabe S, Hori T, Hershey J R. Language independent end-to-end architecture for joint language identification and speech recognition [C]. Proc of the 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU). IEEE, 2017: 265-271.

[73] Baevski A, Zhou H, Mohamed A, et al. wav2vec 2.0: A framework for self-supervised learning of speech representations [J]. arXiv preprint arXiv:200611477, 2020.

[74] Chung Y-A, Hsu W-N, Tang H, et al. An unsupervised autoregressive model for speech representation learning [J]. arXiv preprint arXiv:190403240, 2019.

[75] Liu A H, Chung Y-A, Glass J. Non-autoregressive predictive coding for learning speech representations from local dependencies [J]. arXiv preprint arXiv:201100406, 2020.

[76] Tjandra A, Choudhury D G, Zhang F, et al. Improved language identification through cross-lingual self-supervised learning [J]. arXiv preprint arXiv:210704082, 2021.

[77] 章森, 曹瑞兴, 邓海刚. 一种稳定, 精准, 实时的语音信号基频的检测与提取算法 [J]. Journal of Image and Signal Processing, 2020, 9: 246.

[78] LeCun Y, Boser B, Denker J S, et al. Backpropagation applied to handwritten zip code recognition [J]. Neural computation, 1989, 1(4): 541-551.

[79] Lee C-Y, Xie S, Gallagher P, et al. Deeply-supervised nets [C]. Proc of the Artificial intelligence and statistics. PMLR, 2015: 562-570.

[80] Li X, Wang W, Hu X, et al. Selective kernel networks [C]. Proc of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 510-519.

[81] Yu F, Koltun V. Multi-scale context aggregation by dilated convolutions [J]. arXiv preprint arXiv:151107122, 2015.

[82] Varga A, JMSteeneken H. Assessment for automatic speech recognition: II. NOISEX-92: A database and an experiment to study the effect of additive noise on speech recognition systems [J]. Speech communication, 1993, 12(3): 247-251.

[83] Kingma D P, Ba J. Adam: A method for stochastic optimization [J]. arXiv preprint arXiv:14126980, 2014.

[84] Lan T, Lyu Y, Hui G, et al. Redundant convolutional network with attention mechanism for monaural speech enhancement [C]. Proc of the ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020: 6654-6658.

[85] Li A, Zheng C, Cheng L, et al. A time-domain monaural speech enhancement with feedback learning [C]. Proc of the 2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC). IEEE, 2020: 769-774.

[86] Iwamoto K, Ochiai T, Delcroix M, et al. How bad are artifacts?: Analyzing the impact of speech enhancement errors on asr [J]. arXiv preprint arXiv:220106685, 2022.

[87] Chen S, Wang C, Chen Z, et al. Wavlm: Large-scale self-supervised pre-training for full stack speech processing [J]. IEEE Journal of Selected Topics in Signal Processing, 2022, 16(6): 1505-1518.

[88] Yang S-w, Chi P-H, Chuang Y-S, et al. Superb: Speech processing universal performance benchmark [J]. arXiv preprint arXiv:210501051, 2021.

[89] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [J]. Advances in neural

information processing systems, 2017, 30.

[90] Hsu W-N, Bolte B, Tsai Y-H H, et al. Hubert: Self-supervised speech representation learning by masked prediction of hidden units [J]. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2021, 29: 3451-3460.

[91] Gulati A, Qin J, Chiu C-C, et al. Conformer: Convolution-augmented transformer for speech recognition [J]. arXiv preprint arXiv:200508100, 2020.

[92] Park D S, Chak W, Zhang Y, et al. Specaugment: A simple data augmentation method for automatic speech recognition [J]. arXiv preprint arXiv:190408779, 2019.

[93] Ginsburg B, Castonguay P, Hrinchuk O, et al. Stochastic gradient methods with layer-wise adaptive moments for training of deep networks [J]. arXiv preprint arXiv:190511286, 2019.

[94] Graves A, Fernández S, Gomez F, et al. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks [C]. Proc of the Proceedings of the 23rd international conference on Machine learning. 2006: 369-376.

[95] Alumäe T, Kukk K. Pretraining approaches for spoken language recognition: TalTech submission to the OLR 2021 Challenge [J]. arXiv preprint arXiv:220507083, 2022.

[96] Amodei D, Ananthanarayanan S, Anubha R, et al. Deep speech 2: End-to-end speech recognition in english and mandarin [C]. Proc of the International conference on machine learning. PMLR, 2016: 173-182.

攻读硕士学位期间取得的成果

学术论文：

[1] **Chen C** (1/6). Selective Kernel Network with Intermediate Supervision Loss for Monaural Speech Enhancement [C]. 2021 IEEE 21th International Conference on Communication Technology (ICCT), Tianjin, China, 2021:1330-1334.

[2] **Chen C** (4/6). Improving Monaural Speech Enhancement with Dynamic Scene Perception Module [C]. 2022 IEEE International Conference on Multimedia and Expo (ICME), Taipei, Taiwan, 2022:1-6.

[3] **陈聪**（5/6）. 基于 RefineNet 的端到端语音增强方法[J]. 自动化学报, 2022:554-563.

专利：

[1] 一种基于自适应注意力机制和渐进式学习的单声道语音增强方法[P].中国，发明专利，CN113160839B，2022-10-14.

[2] 一种基于信息蒸馏与聚合的低信噪比语音增强方法[P].中国，发明专利，CN113936679A，2022-01-14.

[3] 一种基于时频跨域特征选择的语音分离方法[P].中国，发明专利，CN113113041B，2022-10-11.

[4] 基于多尺度信息感知卷积神经网络的单通道语音增强方法[P].中国，发明专利，CN113936680A，2022-01-14.

科研项目：

[1] 英汉双语无线电陆空通话语音识别引擎系统. 20201202-1511*.

[2] 面向航天地面系统的智能人机交互技术. SXX19629X060.

获奖情况：

[1] 2021-2022 年度校“学业二等奖学金”

[2] 2022 年度校“学术青苗奖学金”

[3] 2020 年度“研究生新生一等奖学金”

竞赛：

[1] 2022 iFLYTEK A.I.开发者大赛-多语种语音识别（非受限系统）赛道第二名