南 开 大 学

本 科 生 毕 业 论 文 （设 计）

中文题目： **基于 ECAPA-TDNN 的语种识别系统设计与实现**

外文题目： Design and Implementation of Language Identification System Based on ECAPA-TDNN

学　　号： 1911460

姓　　名： 任永文

年　　级： 2019 级

学　　院： 计算机学院

系　　别： 计算机科学与技术

专　　业： 计算机科学与技术

完成日期： 2023 年 4 月

指导教师： 秦勇　教授

关于南开大学本科生毕业论文（设计）的声明

本人郑重声明：所呈交的学位论文，是本人在指导教师指导下，进行研究工作所取得的成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或没有公开发表的作品内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名：

年　　月　　日

本人声明：该学位论文是本人指导学生完成的研究成果，已经审阅过论文的全部内容，并能够保证题目、关键词、摘要部分中英文内容的一致性和准确性。

学位论文指导教师签名：

年　　月　　日

摘 要

本文针对语音语种识别技术的研究现状和发展趋势进行了探讨，介绍了传统的 GMM、HMM 等模型以及近年来兴起的深度学习模型和自监督预训练模型的应用情况。针对深度学习模型中的 ECAPA-TDNN 模型进行了详细介绍，分析了该模型在语音信号处理方面的优势和应用前景，包括对变化的语音信号具有更强的鲁棒性，以及对于小数据集的训练具有更好的表现等。在实验部分，本文将 ECAPA-TDNN 模型应用于语音语种识别任务中，设计了对比实验以验证模型的有效性，包括使用不同的超参数和损失函数等，从而分析得出了影响该模型性能表现的可能因素。最后，本文将该模型部署到了网页系统以方便用户使用。该研究为语音语种识别技术的进一步发展和实际应用提供了有益的探索和参考。

关键词： ECAPA-TDNN，语音语种识别，深度学习

Abstract

This article discusses the research status and development trends of speech language identification technology, including traditional models such as GMM and HMM, as well as recent advances in deep learning models and self-supervised pre-training models. The ECAPA-TDNN model in particular is described in detail, analyzing its advantages and potential applications in speech signal processing, including stronger robustness to changing speech signals and better performance on small datasets. In the experimental section, the ECAPA-TDNN model is applied to speech language identification tasks, and comparative experiments are designed to verify the effectiveness of the model, including the use of different hyperparameters and loss functions, in order to analyze potential factors that affect model performance. Finally, the model is deployed to a web system for ease of use by users. This study provides valuable exploration and reference for the further development and practical application of speech language identification technology.

Key Words: ECAPA-TDNN,speech language identification,deep learning

目 录

第一章　引言

第一节　背景与意义

当今社会，语音技术的应用已经渗透到我们日常生活的方方面面，如语音助手、语音识别、语音合成等。其中，语音语种识别作为语音技术的重要应用之一，具有广泛的应用场景和重要的理论研究价值。

语音语种识别的主要任务是通过对不同语音信号的特征提取和模式识别，自动地对输入语音信号的语种进行识别。该技术可以应用于语音翻译、语音导航、语音识别等领域，为用户提供更加智能、便捷、高效的服务。同时，语音语种识别的研究也能够深入探索语音信号的特征提取、语音模式分类、深度学习等领域，推动语音技术的发展和进步。

然而，语音语种识别任务面临着很多困难和挑战，如不同语种的发音方式、语音信号噪声和干扰等问题，这些因素都会影响识别的准确性和稳定性。因此，针对语音语种识别技术的研究和探索，具有重要的现实意义和理论价值。本文基于深度学习实现了一个语音语种识别系统，旨在提高语音语种识别的准确率和鲁棒性，为语音技术的应用和发展做出贡献。

第二节　国内外研究成果

本节将综述语音语种识别领域中的传统方法和深度学习方法，并对近年来的自监督语音预训练模型进行简要介绍。在语音语种识别领域，传统的基于高斯混合模型的方法已经被证明具有一定的局限性。McLaughlin 等 1999 年提出的 GMM-UBM（Gaussian mixed model-universal background model）[**mclaughlin_study_1999**] 方法使用 GMM 对语音信号进行建模后使用 UBM 进行建模和训练，通过最大似然估计训练模型参数，进而实现语音信号的分类，但需要大量数据来估计协方差矩阵。Fine 等 2001 年提出的 GMM-SVM（Gaussian

mixed model-support vector machine）[**fine_hybrid_2001**] 方法则使用 SVM 进行分类，在训练数据较少的情况下也能够取得较好的效果。Garcia-Romero 等 2011 年提出 i-vector（identificaiton-vector）[**garcia-romero_analysis_nodate**]，该方法则是将 GMM 超向量映射到一个低维向量即为 i-vector 特征，该方法含有较为显著的语音特征，能够较好地表达语音信号的说话人信息和环境信息，从而实现更准确的语音语种识别。

自监督语音预训练模型是近年来比较热门的一种语音识别技术，它利用大量未标注的语音数据进行预训练来提高语音识别模型的性能。Aäron van den Oord 等人于 2018 年提出的 CPC（Contrastive Predictive Coding）是一种基于对比学习的自监督预训练模型，它依赖噪声对比估计训练模型，训练编码器提取输入到上下文网络的特征，并在输出端进行正例和负例的鉴别性学习来优化网络参数。Chung 等人于 2019 年提出的 APC（Autoregressive Predictive Coding）是一种生成性模型，在训练过程中预测重建未来语音片段的频谱。这种方法类似于 CPC，但是使用了自回归模型来生成样本，而不是使用对比模型来区分相邻的样本，从而可以处理可变长度的序列。张正友等人于 2020 年提出的 NPC（Noise Predictive Coding）在输入被隐蔽掉的帧前后的一些帧后进行隐蔽重建，而不是进行预测。使用卷积块来限制信息的前向传递过程，从而保证重建使用的信息来自周围的前后文。Baevski 等人于 2019 年提出的 wav2vec 在第一阶段使用无监督方法对一个嵌入式卷积神经网络进行预训练，第二阶段使用有标签的数据微调训练一个 BERT 模型，通过两阶段训练过程提高了预训练模型的性能。Baevski 等人于 2020 年提出的 wav2vec2.0 则利用了一个更大的数据集和更深的神经网络，音频分段输入 CNN 层提取特征后一方面输入到 VQ 层，另一方面随机 mask 掉一些帧输入到 Transformer 层提取特征。该方法可以提供更具有代表性的特征表示，训练速度更快，效果更好。

在语音语种识别任务中，深度学习方法已经成为了主流。其中，使用 DNN（Delay Neural Network）提取声学特征的方法已经被广泛研究和应用。Bao 等 2013 年提出的 BN（BottleNeck）-DNN 在 DNN 基础上增加了瓶颈层，对声学特征进行多层的非线性映射后降维压缩，最后提取到鲁棒性更强的 i-vector 特征，该方法对长时音频效果更好，但对短时音频效果较差。Peddinti 等 2015 年提出

的 TDNN（Time Delay Neural Network）通过多个全连接层将不定长语音信号映射为固定维数的向量，即为 x-vector 特征。提出的 Extended-TDNN 使用了更加灵活的结构，包括不同大小和方向的卷积核、不同的池化策略和不同的激活函数来扩展感受野，并加入了 Dense 层增加网络深度，使用残差结构提高网络的稳定性，提取到更加鲁棒的 x-vector 特征。Desplanques 等 2020 年提出的 ECAPA（Emphasize Channel Attention）-TDNN 采用了基于注意力机制的通道选择方法扩展感受野，使用残差块提高模型的性能，使其在多种语音识别任务中表现出色。

综上所述，基于深度学习的语音语种识别方法已经成为当前的研究热点和前沿，ECAPA-TDNN 作为其中的一种方法，其具有多层上下文信息和金字塔结构的特点，在语音信号处理中具有良好的应用前景。目前，语音信号处理领域已有一些研究基于 ECAPA-TDNN 获得了一些成果。例如，胡曼等人提出了一种基于 ECAPA-TDNN 的多语种情感识别方法，取得了较好的效果。另外，许多研究者也在探究如何结合其他技术和方法，进一步提升基于 ECAPA-TDNN 的语音信号处理的效果和应用价值。本文将基于 ECAPA-TDNN 的语音语种识别进行研究，对 ECAPA-TDNN 进行深入分析，并对其在语音语种识别中的应用进行实验验证。最后，希望本文的研究成果将有助于深入理解 ECAPA-TDNN 方法及其在语音语种识别中的应用，并为相关领域的研究和应用提供一定的参考和借鉴价值。

第二章 ECAPA-TDNN 模型

ECAPA-TDNN 是提出的一种基于 TDNN 的网络模型，TECAPA-TDNN 中大量使用了 TDNN 结构。TDNN 结构包含 Conv1dReluBn 三个操作，其中 Conv1d 是 1 维空洞卷积，在卷积过程通过设置不同的空洞率可以提取到音频的不同特征，Relu 是激活函数，Bn 是标准化层。ECAPA-TDNN 的网络流程为将 80 维的音频 Fbank 或者 MFCC 特征输入模型后首先通过一个 TDNN 层，再通过 3 个 SE-Res2Block，做一个全连接后再通过一个注意力卷积池化层，全连接映射到 192 维即为最终的到的特征向量。整个模型使用了较多的残差结构来考虑全局信息，也解决了梯度消失问题。除此之外，模型还有很多其他改进，下面主要介绍其中有代表性的三点。

第一节 SE-Res2Block 模块

模型中用到了 3 层 SE-Res2Block，如图所示，Conv1dReluBn 是传统的 TDNN 块，在此基础上通过融合 SE 结构和 Res2 结构得到新的模块，能够提取输入张量多尺度的特征，提高网络的性能和鲁棒性。

SE 模块是 SENet（Squeeze and Excitation Network）提出的一个即插即用的模块，通过学习输入特征的通道间关系，将有用的信息放大，减少无用信息的干扰。该模型主要由两部分组成：压缩和激励。压缩部分使用自适应平均池化操作将原始大小为 H*W*C 的张量压缩成 1*1*C，减少了参数量，增加了感受野。激励部分将 1*1*C 的张量通过两个全连接层进行自注意操作再通过 sigmoid 函数映射到 [0,1] 之间，得到每个通道的重要性即通道权重，将通道权重作用到原始的 H*W*C 的输入特征上，实现了对通道的加权。SE 模块引入了通道注意力机制，在通道维度上进行特征提取，提升了模型的精度。

Res2Block 模块是 Res2Net 中提出的一种在 ResNet 基础上改进得到的残差结构。如图所示，传统的 ResNet 张量经过 1*1 的卷积后直接做 3*3 的卷积得到

输出，而 Res2Net 将该张量经过 1*1 的卷积后分成多部份，第一部分直接输出，剩下每个部分都依次加上上一部分的输出后再做一个 3*3 的卷积得到输出。通过这样的改进能够增加感受野，使得输出包含多尺度的特征。

SE-Res2Block 模块将 SE 模块和 Res2Block 模块进行了结合，通过对输入特征的通道关系进行加权，并且在残差连接中添加了通道加权，实现了更加精细的特征提取和重建。在 ECAPA-TDNN 中，SE-Res2Block 模块被广泛应用于 TDNN 网络的每个卷积层中，有效地提高了网络的性能和鲁棒性。

第二节　多层特征聚合和求和

在 ECAPA-TDNN 模型中使用了三个 SE-Res2Block，它们使用不同 dilation 的空洞卷积，可以提取到音频的不同特征。在每个 SE-Res2Block 的输入中，不仅包含上一个块的输出，还加上了之前所有块的输出。通过这种结构，模型能够关注到音频的多层信息，实现多层特征的融合。此外，每个 SE-Res2Block 的输出被拼接在一起，作为全连接层的输入。这种残差结构的设计能够提高模型的稳定性和泛化能力，以及防止梯度消失。这些设计使得该模型在语音语种识别任务中表现出色，达到了较高的识别准确率。

第三节　通道和上下文依赖的统计池化层

这一改进主要针对的是图中的 ASP 层，ASP 层是 Attention Statics Pooling[**okabe_attentive_2018**] 的缩写，是一种用于语音信号处理的注意力机制。在该层中，输入的音频信号 x 被分成多个子带，并计算每个子带的均值和标准差，将其拼接在 x 后面作为输入，实现上下文依赖，使注意力机制能够关注到全局属性。取平均后通过一系列计算，包括 linear-tanh-linear 得到每个通道的注意力，再通过 softmax 得到每个通道的权重，这些权重代表了每个通道的重要性。最后，将 x 带权重的均值和标准差拼接在一起，得到最终的输出。这种通道注意力机制使得网络能够关注到相同音频中不相似的部分，从而提高了模型的准确性和鲁棒性。

第三章　损失函数

损失函数用来衡量模型预测结果与真实结果之间的差异。通常情况下，我们会对预测向量 P 进行归一化，以表示每个分量表示预测结果为该类别的概率，而 Q 则是通过将真实类别进行 one-hot 编码后得到的向量。对 P 和 Q 进行差异性计算，得到的值即为损失 loss。在训练过程中，我们希望同一类别的数据尽量靠近，不同类别的数据尽量远离。

第一节　交叉熵和相对熵

交叉熵和相对熵是两个常用的损失函数。香农理论定义每传输单位信息意味着接收者的不确定性减少了一半，因此，假设有用信息量为 Y，不确定性为原来的 X 倍，则 X 与 Y 满足 $X=(\frac{1}{2})^Y$。若一件事发生的概率为 P，则 $P=X=(\frac{1}{2})^Y$，推导得到 $Y=-log_2(P)$，那么平均信息量为 $\sum_{i=1}^{n}P(x_i)Y(x_i)=-\sum_{i=1}^{n}P(x_i)logP(x_i)$，这个概念就是熵，它用来定义信息的不确定性和混乱程度，熵越小，两个分布之间的差异性就越小。

交叉熵损失函数用于测量两个概率分布之间的差异性。对于两个向量 P 和 Q 而言，交叉熵的公式是 $H(P,Q)-\sum_{i=1}^{n}P(x_i)logQ(x_i)$。相对熵（KL 散度）也是用于测量两个概率分布之间的差异性，与交叉熵不同的是，它表示的是从 P 角度来看，P 和 Q 的差异性，因此相对熵没有对称性。相对熵的公式是 $D_{KL}(P,Q)=\sum_{i=1}^{n}P(x_i)log(\frac{P(x_i)}{Q(x_i)})$。推导可得 $H(P,Q)=D_{KL}(P,Q)+H(P)$，也就是说 P 与 Q 的交叉熵等于 P 与 Q 的相对熵加 P 的信息熵，一般来说，交叉熵比相对熵多一个常数量。

第二节　Softmax 及其改进

对于分类任务，常常使用 softmax 函数将预测结果进行归一化，使其表示每个类别的概率。但是，softmax 并没有考虑到类内距离的问题。因此，研究者们

对 softmax 进行了改进，使其在优化类内距离的同时能够更好地分类。其中一种改进方法是 AAM-Softmax，与 Softmax 相比，首先对 w 和 x 都做了归一化，这样能使得 $w \cdot x$ 由 $||w|| \cdot ||x|| \cdot cos(\theta)$ 变为 $cos(\theta)$，使得决定数据在每一类的概率只取决于 θ，从而使得系统能聚焦到角度上。其次，对于属于本类的数据，将 $cos\theta$ 转化为 $cos(m_1\theta + m_2) - m_3 (m_1 > 1, m2 > 0, m3 > 0)$ 能够使值变小，为了让结果达到原来的值，系统会减小 θ，从而使得类内的距离变小，从而达到我们的目的。AAM-Softmax 只使用到了 m_2 参数，另外两种改进方法是 L-Softmax 和 SphereFace，它们分别引入了 m_1 和 m_3 参数。

第四章　实验及结果

为了验证 ECAPA-TDNN 在语音语种识别领域的表现，论文用 Pytorch 框架实现了 ECAPA-RDNN 模型，并基于该模型进行了实验训练得到一个适用于语音语种识别任务的网络模型。除此之外，设置了对比实验探讨影响模型精确度的因素，对实验结果进行分析后得到了一些有意义的经验结论。

第一节　数据准备

本实验采用的数据集是 Common Voice，该数据集包含了 45 种不同语种的短音频，数据格式为 wav 格式，采样频率为 16kHz，并划分好了训练集、验证集和测试集。

为了消除数据的噪声和差异，实验对数据进行了预处理，将音频统一为 2-3 秒的片段，并进行标准化处理。另外，为了增强数据集的多样性和鲁棒性，本文还采用了以下三种数据增强技术：

1. 非语音噪声混合：利用 MUSAN 数据集中的音乐、噪声和背景声音，将其与原始语音信号混合，增加噪声和其他非语音因素。
2. 模拟混响注入：使用 RIR Noises 中的模拟混响脉冲响应，将其与原始语音信号混合，以模拟不同的混响环境。
3. SpecAugmentation：在频域进行数据增强，通过对语音信号的声谱图进行遮盖和替换等操作，来增加数据集的多样性和鲁棒性。

以上三种数据增强技术的组合可以显著提高模型的性能和鲁棒性，从而更好地适应不同的语音识别场景。

实验采用 FBank 作为语音信号的频谱特征。FBank 特征是语音信号识别中常用的特征提取方法之一。具体过程为对信号进行预加重，丰富高频信号后分帧加窗减少突变，通过傅里叶变换得到频谱信息后计算能量谱，过滤得到 FBank 频谱。最后，归一化得到的 FBank 特征向量就可以用于声学模型的训练和识别。

第二节 实验设置

实验使用 Pytorch 工具包实现了 ECAPA-TDNN 模型，并对语音信号进行了预处理、数据增强、归一化等步骤后提取了 80 维的 FBank 声学特征作为模型的输入。实验使用 Pytorch 工具包实现了 ECAPA-TDNN 模型，并将之作为基准模型来提取语音的 Embedding 特征，该模型的通道设置为与原论文相同的分别为 [1024,1024,1024,1024,3072] 维的网络层和 128 维的自注意力通道，最后映射为 192 维的特征向量，即为模型输出。分类器用线性映射实现，将向量由 192 维映射到任务所需的 45 维从而完成语音语种分类任务。

实验在一块配备了 NVIDIA GeForce RTX 3090 Ti 显卡的云服务器上进行。在训练过程中，采用了 dropout、正则化等方法以防止过拟合。采用 Adam 作为优化器，在训练过程中采用了学习率调整策略，同时为了防止训练不收敛添加了 early stop 机制，并将最大训练轮次设置为 50。实验将准确率和错误率作为性能指标，在每轮训练结束后，进行一次测试以评估模型性能。

实验针对学习率，批量大小和损失函数进行了对比试验，分析不同因素对模型性能的影响，并在该过程中进行微调以确定最佳的参数组合，提高模型的泛化能力和分类性能。除此之外，实验还针对是否使用预训练模型进行了对比试验，预训练模型使用的是在 Voxceleb2 数据集下训练好的说话人识别任务预训练模型。整个实验采用了交叉验证和实验重复来确保结果的稳定性和可靠性。

第三节 实验结果

实验针对学习率、批次大小、损失函数和是否使用预训练模型进行了对比试验，具体设置和所得结果如表4.1所示。

表 4.1 基于 ECAPA-TDNN 的多组对比实验

NO	loss_function	lr	batch	acc
1	softmax+crossentropy	1e-3	64	测试集 65%
2	softmax+crossentropy	1e-4	4	测试集 43%
3	aam+crossentropy	1e-3	64	收敛慢，训练集 31%+
4	aam+crossentropy	1e-4	64	收敛慢，训练集 32%+
5	aam+crossentropy+pretrain	1e-3	64	收敛快，训练集 79%，测试集 66%
6	aam+kldiv	1e-3	64	训练集升到 20% 后下降
7	aam+kldiv	1e-4	4	收敛慢，训练集 32%
8	aam+kldiv	1e-4	64	训练集 76%，测试集 46%
9	aam+kldiv+pretrain	1e-4	64	收敛快，训练集 89%，测试集 72%

实验结果表明，使用 AAM+kldiv+pretrain 的方法在分类准确率上表现最好，能达到 72% 以上，同时发现使用不同的损失函数会对模型的性能产生不同的影响。

通过 NO.1 可知，使用 Softmax 作为归一化函数可以使得模型很快就达到较高的准确率，而使用 AAM-Softmax 可以提高模型的边界间隔，让模型学习到更加鲁棒的特征表示，从而达到更高的准确率。但是在训练初期可能会导致模型性能下降，需要适当调整学习率以平衡模型的学习速度和稳定性。

通过 NO.2 和 NO.7 可知，batch_size 太小会导致模型难以收敛，batch_size 的取值和数据集有很大的关系，一般来说小 batch_size 会使收敛速度变慢，但最终学到的特征更准确。

通过 NO.6 和 NO.8 可知，对于 kdiv 来说 1e-4 是比较合适的学习率。通过 NO.4 和 NO.5 可知，对于 crossentropy 来说 1e-3 是比较合适的学习率。这与 kdiv 与 crossentropy 的内部实现有关，因为 pytorch 在实现时，crossentropy 比 kdiv 多除了一个 batch_size，因此对于同样的数据集和模型，所需的学习率更大。

通过 NO.5 和 NO.9 可知，使用预训练模型可以显著提高模型的性能，即使预训练模型是在不同的任务（例如说话人识别）上训练的。这是因为说话人识别与语种识别都属于分类任务，因此该预训练模型也可以较好地适应新的任务

和数据，加快模型收敛速度。

总体来说，ECAPA-TDNN 模型能在语音语种识别任务上得到较好的结果，后续可以针对该任务以及所使用数据集的特点对模型本身进行改进创新。

第五章 系统部署

本章基于前两章实现的 ECAPA-TDNN 模型实现了一个语音语种识别系统，该系统支持用户上传音频或者在线录制音频后进行识别。本章将详细介绍系统的设计思路，功能的实现细节，并且对最终的结果进行展示，对系统功能进行测试。

第一节 系统设计

Flask 是一个轻量级的 Web 应用程序框架，上手容易且部署简单，因此本系统采用 Flask 框架作为 Web 服务的核心框架。用户可以通过 Web 界面上传或在线录制音频文件，传给后台经过 ECAPA-TDNN 语音语种识网络封装成的模型得到输出并返回的结果后，在 Web 界面上展示出来。系统的主要组成部分包括：

1. Flask 框架：作为 Web 服务的核心框架，提供路由、视图、请求处理等功能。具体实现为前端点击“预测”后将音频文件传输给后端并向后端发出请求，后端收到文件后将文件格式转化为 wav，提取 2-3 秒的音频输入模型预测得到结果向量返回给前端。
2. ECAPA-TDNN 语音语种识别网络：用于实现语音语种识别的核心算法，需要提前训练好，并提取该模型的 JIT 版本，方便在 Flask 后端调用。
3. 前端界面：前端使用 html 和 javascript 实现，提供上传文件，在线录制，结果展示，音频播放等功能。

第二节 功能实现

系统的功能包括音频获取、后端处理、结果展示等模块。下面将分别介绍这些模块的实现细节。

用户可以通过 Web 界面上传音频文件，也可以在线录音获得音频文件。音频上传使用了 html 的 file 组件实现，音频录制则使用 javascript 中的 MediaRecorder

类实现。文件将会在网页上展示，用户可以播放以验证音频，可以通过给文件创建一个 URL，用 audio 组件访问该 URL 实现该功能。除此之外，点击“预测”按钮后，前端会将文件发送给 flask 后端并请求返回结果。

在 Flask 应用中使用 ECAPA-TDNN 模型，需要使用 PyTorch 的 jit 模块将模型转换为 JIT 格式，并使用 torch.jit.load() 函数加载模型。在进行语音语种识别之前，我们需要对上传的音频文件进行预处理，包括采样率转换、时长裁剪、格式转换等操作。处理后的音频文件输入模型，输出特征向量通过路由传给前端。

前端捕捉到特征向量后选取其中最大的一个分量作为预测得到的结果类别，并按顺序将可能性最高的 5 种语言的结果进行展示，方便用户进行参考和比较，这部分功能通过 javascript 脚本实现。

第三节　系统测试

为了验证系统的性能和可靠性，我们进行了一系列的系统测试，并收集了一些样例音频进行验证。测试结果表明系统能够稳定地运行，并且具有良好的识别准确率和响应速度。

致 谢

在我即将完成本科学业，开始迈向人生新的阶段之际，我要对许多人表达我的谢意。

首先，我要感谢我的导师，他不仅仅是我在学术上的引路人，更是我人生中的良师益友。在我整个本科学习期间，导师给予了我无微不至的关心和指导，帮助我解决了许多学习和生活上的难题。导师的言传身教将成为我人生中的宝贵财富，我将铭记于心。

其次，我要感谢我的家人。感谢他们对我毫无保留的支持和关爱，没有他们的鼓励和支持，我不可能走到今天的台阶。家人的爱是我前行的动力，是我生命中最重要的支撑。

还要感谢我的同学和朋友。在我求知路上的你们，给了我很多帮助和鼓励，陪伴我度过了一个又一个难忘的日子。我们相互扶持，共同成长，这将是我一生中最美好的回忆之一。

最后，我要感谢所有为我提供帮助和支持的人们。感谢您们的鼓励、支持、启示和关怀。没有你们的帮助，我不可能取得今天的成果。谢谢您们！

再次感谢所有支持和帮助我的人，你们的支持是我不断前行的动力和信仰，我将怀着感激的心情，迈向新的人生旅程。