

2022 年英特尔杯大学生电子设计竞赛嵌入式系统专题邀请赛

2022 Intel Cup Undergraduate Electronic Design Contest

- Embedded System Design Invitational Contest

# 初选项目设计方案书



Intel Cup Embedded System Design Contest

项目题目：\_\_\_\_\_

学生姓名：\_\_\_\_\_

指导教师：\_\_\_\_\_

参赛学校：\_\_\_\_\_

# 项目题目

## 摘要

摘要正文五号宋体，首行缩进二个字，单倍行距。

**关键词：**五号宋体，逗号分开，最后一个关键字后面无标点符号。

# 目 录

第一部分 项目背景.....	2
1.1 新闻行业背景与现状.....	2
1.2 嵌入式人工智能与边缘计算.....	2
1.3 项目意义与前景.....	2
1.4 语音识别技术.....	2
1.4.1 特征提取.....	2
1.4.2 模型建立.....	3
1.5 文本摘要技术.....	4
1.5.1 抽取式摘要.....	5
1.5.2 生成式摘要.....	5
1.6 网络通信技术.....	5
1.6.1 网络协议.....	5
1.6.2 实时系统.....	6
第二部分 项目设计方案.....	7
2.1 研究开发内容.....	7
2.1.1 项目具体目标人群与场景.....	7
2.1.2 项目开发功能模块.....	7
2.1.3 子系统开发内容.....	7
2.2 系统架构.....	8
2.2.1 硬件平台.....	8
2.2.2 整体架构.....	9
2.3 技术关键.....	9
2.4 主要特色.....	9
2.5 预期目标（技术指标等）.....	9
2.6 项目实施方案.....	9
2.6.1 整体方案.....	9
2.6.2 语音识别模块.....	9
2.6.3 文本摘要模块.....	11
2.6.4 网络通信与边缘计算模块.....	12
2.7 技术路线.....	12
2.8 进度安排.....	12
2.9 模板.....	12
第三部分 团队组成.....	15
参考文献.....	16

# 第一部分 项目背景

## 1.1 新闻行业背景与现状

## 1.2 嵌入式人工智能与边缘计算

## 1.3 项目意义与前景

## 1.4 语音识别技术

语音是人机交互中的一个重要环节。语音识别技术自从计算机诞生以来，就有学者不断地进行研究。由于近几年神经网络与深度学习成为前沿热点，将各种网络应用于语音识别的应用也有许多的成果。国内在这方面研究较多的公司例如科大讯飞，技术主要运用于输入法、录音笔等领域。国外的例如微软，在2017年在基准测试Switchboard task中达到了人类相同的水平。此外，由于语音识别目前发展地已经较为成熟，各种输入法，微信语音、腾讯会议等实时交流软件中，也具有一定的语音转录功能。

但是由于语音识别需要占用一定的资源，现阶段的应用一般不会在本地离线部署识别算法与框架，大多数应用场景更适合采用云计算的方式进行。同时对于一个完备的人工智能语音识别系统，其成本也较高，不太符合大多数人的需求。因此，我们团队将从嵌入式与边缘计算的角度考虑，如何在有限的资源内选择合适的模型，能够最大限度地兼顾准确率、复杂度与实时性。目前有许多不同较为成熟的模型与框架，各有特点。但是它们的基本思想是一致的，就是首先预处理（去除静音、噪音并切分），随后提取声学特征信息，并用概率模型将其分解为声学模型与语言模型，具体如下。

### 1.4.1 特征提取

声学信息是一个时域的、连续的信息。作为机械波，其信息可以利用在空间某一个点处的振动描述。计算机系统在进行采样时，需要将其离散化，通过一定的采样率得到振动强度关于时间的变化。通过傅里叶级数

$$\tilde{x}_N(t) = \sum_{n=-N}^N \frac{\omega}{2\pi} e^{in\omega t} \int x(t) e^{-in\omega t} dt \quad (1.4-1)$$

以及离散傅里叶变换

$$\tilde{X}_k = \sum_{n=0}^{N-1} \tilde{x}_n \left[ \cos\left(\frac{2\pi}{N} kn\right) - i \sin\left(\frac{2\pi}{N} kn\right) \right] \quad (1.4-2)$$

将其转换为正弦信号叠加，形成频谱。

对声音的特征提取，主要包含声波的一系列总体特征，包含音色、能量分布、频率分布，以及和韵律相关的关于一段时间的局部特征。目前主要声学特征提取方法包括线性预测系数（LPC）、倒谱系数（CEP）、梅尔频率倒谱系数（MFCC）等。线性预测系数将离散信号值估计为关于前面若干样本的线性函数，并使用线性函数中的一组系数来描述这个信号。离散信号值 $x(n)$ 的估计具体可以表示为

$$\hat{x}(n) = \sum_{i=1}^p a_i x(n-i) \quad (1.4-3)$$

其中的线性预测系数（LPC） $a_i$ 可以通过最小二乘法（最小化方均误差）计算，即通过自相关系数 $R(i) = E\{x(n)x(n-i)\}$ （ $E$ 为关于变量 $n$ 的期望值）计算方程的解

$$\sum_{i=1}^p a_i R(j-i) = R(j) \Leftrightarrow \mathbf{R}\mathbf{A} = \mathbf{r}, \mathbf{R}_{ij} = \mathbf{R}(i-j), \mathbf{r}_j = R(j) \quad (1.4-4)$$

即可。具体的方法可以通过迭代法等求得。

倒谱（Cepstrum）是对于功率谱的对数进行傅里叶逆变换，可以通过对于输入信号序列首先进行离散傅里叶变换得到频谱，随后平方得到功率谱，取对数后进行逆变换得到的即为倒谱。由于功率谱直接进行逆变换可以得到自相关序列，因此倒谱可以理解为其对数压缩。倒谱系数则为对于倒谱的估计，其中线性倒谱系数（LPCC）使用前若干个自相关系数的线性预测。从线性预测系数（LPC）到线性倒谱系数可以由以下公式求得。

$$cc(n) = \begin{cases} 0, & n < 0 \\ 1, & n = 0 \\ a_n + \sum_{k=1}^{n-1} \frac{k}{n} cc(k)c_{n-k}, & 0 < n \leq p \\ \sum_{k=n-p}^{n-1} \frac{k}{n} cc(k)a_{n-k}, & n > p \end{cases} \quad (1.4-5)$$

梅尔频率倒谱系数（MFCC）将首先声音进行非线性（对数）的预处理，通过梅尔刻度

$$m = 2595 \log \left( 1 + \frac{f}{700} \right) = 1125 \ln \left( 1 + \frac{f}{700} \right) \quad (1.4-6)$$

将频率 $f$ 非线性地重新分布。将功率的傅里叶变换的结果映射到梅尔刻度上，使用窗口函数重新采样，得到新的结果，再和倒谱系数一样取对数进行傅里叶逆变换或者离散余弦变换，即可从中得到倒谱系数。

## 1.4.2 模型建立

语音特征提取完毕后，可以通过声学模型的变换得到音素序列，随后根据语言模型建立从音素到文本的对应转换，因此语音识别的核心在于声学模型与语言模型的建立。输入的语音特征序列为 $X$ ，输出的文本序列为 $Y$ ，那么要求的就是 $\arg\max_Y P(Y|X)$ ，即使条件概率最大的文本序列 $Y$ 。而由于序列 $X$ 已知，条件概率可以用联合概率代替，即

$$P(Y, X) = P(X|Y)P(Y) \quad (1.4-7)$$

对于两个部分可以分别建模，求出文本 $Y$ 产生语音序列 $X$ 的概率（声学模型），以及产生文本 $Y$ 的概率（语言模型）。由于对于每一个字的音素是已知的，因此可建立相应的声学模型。同时，声学模型所求的概率 $P(X|Y)$ 最大的 $Y$ 可以进一步根据音素划分为

$$\operatorname{argmax}_y P(X|Y) = \operatorname{argmax}_y \sum_Z P(X, Z|Y) = \operatorname{argmax}_y \sum_Z P(X|Z)P(Z|Y) \quad (1.4-8)$$

其中 $Z$ 为文本序列 $Y$ 对应的音素序列。

声学模型主要有高斯混合模型(GMM)、深度神经网络(DNN)、循环神经网络(RNN)、卷积神经网络(CNN)等模型结合隐马尔科夫模型(HMM)将经特征处理过的音频信号变换为音素。

混合模型是用来描述总体特征中子特征的一种概率模型，并且不需要知道子特征的观测信息。一个混合模型具有 $K$ 个混合的部分，每一个部分具有参数 $\theta_{1..K}$ 与混合系数 $\varphi_{1..K}$ ，且通过对于 $N$ 个样本的观察值 $x_{1..N}$ ，可以给出部分的观察值 $z_{1..N}$ ，同时要给出在参数 $\theta$ 下每一个部分的概率分布 $F(x|\theta)$ ， $z_i$ 关于 $\varphi_{1..K}$ 的分类。而对于高斯混合模型，其每一部分的概率密度分布满足正态分布。整体的概率分布则可以写为

$$P(x|\mu, \sigma^2) = \sum_{k=1}^K \varphi_k N(\mu_k, \sigma_k^2) \quad (1.4-9)$$

对于参数的估计，可以使用EM算法进行最大似然估计的近似计算。

深度神经网络(DNN)具有多层结构，存在多个隐藏层，并且层与层之间是全连接的，每一个感知机将若干输入的线性组合输入到激活函数中并输出到下一层。通过样本的输入前向传播，可以计算得到损失函数误差，随后进行反向传播更新每一层的参数，最终达到阈值以下则训练完成，得到神经网络各层的参数。循环神经网络(RNN)则在神经网络的基础上增加了对于时间序列的考量，在隐藏层中加入延迟的函数，可以使时间序列中的前置状态影响当前状态。RNN对于具有短期时间相关性的问题例如自然语言处理等具有一定的优势，毕竟对于较长的时间序列输入来讲，一次性地输入神经网络中也是不现实的。类似的思想也可以体现在隐马尔科夫模型(HMM)上，都是对于输入数据局部性的考量与假设。卷积神经网络(CNN)包含卷积层，将全连接的模型进一步细化，通过层之间的卷积运算得到输出，而参数则包含在卷积层里。同样由于其对于范围有所缩小，CNN较为适合图像处理以及自然语言处理的模型训练。

而对于语言模型，使用“链式法则”

$$P(Y) = P(Y_n|Y_{1..n-1})P(Y_{1..n-1}) = \cdots = \prod_{i=1}^n P(Y_i|Y_{1..i-1}) \quad (1.4-10)$$

较简单基于统计的方法一般采用n-gram模型，即每一项条件概率仅依赖于前 $n-1$ 个字。因此，可以通过对于已有文本的统计，获得这些条件概率，并通过结合声学模型，共同确定 $Y$ 值。这种方法的局限在于当字符长度 $n$ 较大时，结合上下文的信息将被弱化。解决这个问题的办法主要可以使用神经网络直接对于上述公式中概率的建模。目前也有许多预训练的模型可以应用。

## 1.5 文本摘要技术

文本摘要技术属于自然语言处理的范畴，也是有一定的历史了。同样也随着近些年硬件资源的丰富与深度学习的不断发展，可以在准确度、智能程度上有显著提升。由于文本自动摘要是自然语言处理一个较为重要的应用之一，目前国内各个互联网企业对于人工智能的投入多多少少会涉及到这方面，例如百度的智能云，提供了文本自动摘要的接口，对于新闻语音播报、新闻聚合、消息推送等存在字数限制的场景有比较实际的用途。文本摘要主要的两种类型为抽取式摘要与生成式摘要。抽取式摘要是从原文中选取关键词句，可以基于传统的统计与聚类，也可以基于目前流行的神经网络。生成式摘要则可以避免抽取式摘要中存在的

一系列连贯性与灵活性的问题，允许新的词组出现。

### 1.5.1 抽取式摘要

传统的方法例如最简单的Lead-3算法，根据文章本身撰写的特点，直接抽取最开始的三个句子，也可以达到不错的效果。稍微复杂些的算法有TextRank，将文档中句子作为结点，语义关系作为边，第*i*结点的权重为

$$W_i = (1 - d) + d \sum_{V_j \in \text{in}(V_i)} \frac{w_{ji} W_j}{\sum_{V_k \in \text{out}(V_j)} w_{jk}} \quad (1.5-1)$$

其中*d*为阻尼系数，一般为0.85， $w_{ij}$ 为相关性，作为边权。这个公式是一个递推式，需要迭代求值。第*i*个结点的权重可以理解为对于所有从*j*到*i*的入边权重占所有从*j*出去边权重总和的比例求和。如果一个句子与其他句子相关性较小，那么最终迭代收敛后，由于 $w_{ij}$ 十分小，结点的权重也会下降。最终根据结点权值排序，可以得到较为重要的句子作为摘要。同时如果将句子换为词组，经过同样的算法，最终也可以得到几个关键词。另一种方法是用聚类算法。简单来讲就是将句子编码形成向量，再使用K均值聚类算法将句子向量无监督学习分为若干类别，最后选择与质心最近的聚类作为关键句摘要。这个算法关键在于向量化要处理得合理，使相似的句子可以距离较近。相比于TextRank算法，聚类算法的模型较为简单。

由于序列标注问题可以很好地适应神经网络分类模型，文本摘要也可以转换为序列标注问题。同样以句子为单位，标签为摘要或不摘要两类。模型建立一般可以使用RNN框架，例如SummarRuNNer，Seq2Seq方法等等。如果更加细致地划分，句子的分类也可以使用成为摘要句的概率作为输出，选取概率最大的若干句作为摘要。

### 1.5.2 生成式摘要

生成式摘要相当于根据一个长文本，重新生成一个短文本，与机器翻译的任务相类似，都是文本序列转换问题。因此其目前在Seq2Seq方法上的应用有一些成果。使用Seq2Seq时，可以加入注意力机制（Attention）优化。例如Facebook的ABS模型，定义了条件概率

$$p(Y_{i+1}|X, Y_C; \theta) \propto \exp(\mathbf{V} \tanh(\mathbf{U} \tilde{Y}_C) + \mathbf{W} \text{enc}(X, Y_C)), \tilde{Y}_C = [\mathbf{E}Y_{i-C+1}, \dots, \mathbf{E}Y_i] \quad (1.5-2)$$

其中 $\mathbf{E}$ ， $\mathbf{U}$ ， $\mathbf{V}$ ， $\mathbf{W}$ 都是参数矩阵。其中的enc编码模型就可以使用注意力机制优化，对于不同的输入增加权重，使生成的结果更加准确。

## 1.6 网络通信技术

现今各项技术都可以和互联网紧密结合，包括AI技术。当前AI应用很大一部分是分布式与云计算，用户将数据上传至服务器，计算完成后结果返回用户。边缘计算中，也需要涉及终端运算的结果与云的通信，将处理过的数据传送至云端的服务器，这样就涉及到了计算机网络技术。

### 1.6.1 网络协议

计算机之间通过网络通信需要遵守同一个协议，将数据分块，加上文件头打包后便可以通过物理链路传递到目标地址。互联网最基本的协议是TCP/IP协议，TCP/IP协议从顶层到底层分为应用层、传输层、网络层与链路层。

简要地讲，应用层是主机之间应用对于数据流格式的规范，例如HTTP、FTP等等。传输

层则通过定义端口识别数据包的目标应用以及准确性，主要有TCP协议与UDP协议。网络层需要确定的是双方主机的IP网络地址，对方比较IP地址后通过规则解析获得MAC地址。链路层则是通过双方的MAC地址进行数据包的物理层传播。

### 1.6.2 实时系统

实时通信对于系统的实时性具有较高的要求。为保证功能，实时系统必须在规定的时间内完成一定量的任务。涉及到网络传输的实时系统对于网络的稳定性以及带宽有一定的要求。同时生成与传输的数据不能大于网络的带宽，否则就要进行压缩或延时。边缘计算就是从数据量方面对于实时系统在网络传输时具有一定的优化。

实时系统的一个重要特征就是抢占式调度。在语音实时转录等场景，需要同时进行音频采样与分析计算，是一个并行的任务。一般的抢占式调度可以采用现代操作系统中的时间片轮转算法，而根据具体的任务也可以有更多不同根据优先级排序的算法。对于特定任务的嵌入式系统，除了通过任务与进程的调度，还可以从硬件上实现并行。例如比如上述语音实时识别，可以采用多核CPU进行共享内存，或者通过音频采集器件上的缓存结合外设DMA（直接存储器访问）方式等。



## 第二部分 项目设计方案

### 2.1 研究开发内容

#### 2.1.1 项目具体目标人群与场景

随着互联网技术的快速发展,现代通信和传播技术,大大提高了信息传播的速度和广度。在新闻传播行业上,新闻发布的即时性在很多场景中尤为重要;在一场时间相对较长的新闻发布会后,各方力量都在全力争当“最先发布者”,在事件发生后第一时间将信息传送至受众面前,供受众决策。

一方面,对于十分重要的自然与社会事件,例如自然灾害的应急新闻发布会,发布会与报道发出的时间差对新闻发布速度以及凝练与准确性提出了更高要求,与这一信息有切身关联的人们就可以及时采取规避措施,防范灾害、降低伤亡;另一方面,对于与社会大多数成员无关的信息,只需要把最简洁的信息呈现出来,防止新闻泛滥、减少冗余信息;而与信息相关联的少数群体可以继续深挖,从而获得全方位的认识。

借助最新的媒介技术,在新闻传播学框架下,进行与新闻事件同步的、力求达到“微时差”或“无时差”的新闻报道,在追求即时性的同时提取出新闻的关键信息,使新闻简洁凝练、准确无误,对于新闻报道者是十分重要的。

#### 2.1.2 项目开发功能模块

经过对问题的具体分析与实现方案的深入学习,我们采用自顶向下的研发方案将该项目分为三个主要部分:语音识别子系统、文本摘要子系统、网络通信与边缘计算子系统。通过不断挖掘其中的衔接方式,最终实现上述的功能分离。项目后期,我们将会采用自底向上的方式分别完善上述三个子系统,最终对其进行组装为功能完善、效果优异的完整系统,实现我们所需要的预期项目方案、达到预期项目研发目标。

#### 2.1.3 子系统开发内容

##### ① 语音识别子系统

在语音识别子系统中,对采集的声学信息进行预处理,对声学特征信息进行提取,随后,根据声学模型得到音素序列,根据语言模型完成文本转换。

该子系统的功能为:在新闻发布会等场景中完成声音的采集,并准确、快速转换为文本篇章,获得提取新闻文本摘要必需的全部文本信息。

##### ② 文本摘要子系统

在文本摘要子系统中,将对生成式摘要和抽取式摘要的提取效果进行研究,探索开发生成式、抽取式摘要相互结合的提取方案,提高生成结果的准确性。

该子系统的功能为:从完整识别出的新闻文本中选择关键词句、转换文本序列,提取出新闻关键信息文本,即新闻摘要;使新闻报道在保证准确性的前提下,达到简洁凝练的目标要求。

##### ③ 网络通信与边缘计算子系统

在该子系统中,将通过边缘计算架构,在新闻报道的边缘侧发起,在本地边缘计算层完成语音识别与文本摘要计算,而无需交由云端;最后,将终端计算的结果与云通信,将处理后的数据传至云端。

该子系统的功能为：在边缘计算层完成必需的语音识别与文本摘要计算，大幅提高计算效率、降低计算时间，快速生成新闻摘要，从而保证新闻报道的即时性。

## 2.2 系统架构

### 2.2.1 硬件平台

在本项目中，计划使用比赛提供的一块基于第11代英特尔酷睿处理器的边缘计算主机（GNS-V40），以及Habana人工智能开发资源；同时也将在研发过程中，加入必要的辅助设备，完善项目功能。


① 基于第11代英特尔酷睿处理器的边缘计算主机（GNS-V40）。

边缘计算主机工作与12V直流电压下，CPU型号为Tiger Lake，同时具有16G的DDR4内存。内置SSD硬盘128G。外设扩展包括蓝牙、Wifi、USB 3.0/2.0接口等。主机配备有2个Gbps级的LAN接口、6个COM接口以及2个HDMI接口。其上还配置了8个GPIO，以扩充灵活性。

② Habana人工智能开发资源

大赛平台提供的配有Gaudi加速器的Amazon EC2高性能云主机、深度学习镜像与容器可以加速模型的训练。

Amazon EC2（Amazon弹性计算云）是Amazon云服务平台，提供的弹性云主机可以通过远程登录联网与控制。大赛提供的Habana实验室的深度学习镜像与弹性云主机实例如下。镜像一种是图片中的基础版，另一种是预先配置了TensorFlow和PyTorch框架的DLAMI。基础版上装有Habana Synapse软件与Docker引擎，系统安装后可以用从Docker上拉取Habana的不同框架Docker镜像，方便模型的迁移与部署。预先配置版则可以直接通过Python调用硬件资源，例如在TensorFlow上，可以通过`tf.load_library`和`tf.load_op_library`以插件的形式使用。



Habana® Deep Learning Base AMI (Ubuntu 20.04)


★★★★★ (0) | 1.3.0 先前版本 | 提交人 Habana Labs

Linux/Unix, Ubuntu 20.04 | 64 位 (x86) Amazon 系统映像(AMI) | 更新时间: 22/2/14

Habana Deep Learning Base AMI is built for accelerated deep learning on AWS EC2 with Habana® Gaudi® AI Processor.

更多信息

选择



Habana® Deep Learning Base AMI (Ubuntu 18.04)

选择

系列	类型	vCPU	内存 (GiB)	实例存储 (GB)	可用的优化 EBS	网络性能	IPv6 支持
dl1	dl1.24xlarge	96	768	4 x 1000 (SSD)	是	4x 100 Gigabit	是

图 2.2-1 Amazon EC2 平台 Habana 镜像

由于最终需要进行模型的迁移与部署，使用Docker将有利于这一点的实施，训练的模型将连同容器一起部署到边缘计算主机的Docker引擎下，以简化部署环境的流程。

③ Diligent Nexys 4 DDR开发板。

Diligent Nexys 4 DDR开发板是基于Xilinx Artix-7 FPGA的编程平台，上面具有丰富的内存与外设资源。其中包含15850个逻辑单元（每个包含4个6输入运算单元和8个触发器）、4860K的块RAM和128M的DDR2。内部时钟大于450MHz。外设资源包括SD卡连接器、4组Pmod通用外设接口，音频支持脉冲宽度调制输出与脉冲持续时间调制输入，两组7段数码管和一系列按钮开关LED等。其工作在5V直流电下，通过USB或直流电源供电。

④ 国内ECS云服务器

国内各大厂商的ECS云服务器与Amazon EC2类似，可以通过远程连接实例部署服务器与网络连接。虽然Amazon在中国也开展了相关业务，但是从稳定性、网络连接等各方面性能考虑，作为最终连接边缘计算与用户的装置，最好还是使用物理距离较近的连接。

第 8 页 共 16 页

### 2.2.2 整体架构

系统的整体架构图2.2-2，整体功能分为

- ① 声音采集和预处理模块，使用FPGA或ASIC并行实现对于音频信息的预处理，包括模拟信号定时采集、AD转换、FFT以及Mel滤波。
- ② 语音识别模块，包括声学模型和语言模型，声学模型使用RNN在python上实现，而语言模型使用Bert预训练模型。
- ③ 文本摘要模块，将生成的全部文本实时转换为摘要，使用LSTM和python。
- ④ 当文本摘要的结果较为稳定或者达到一定的条件时，可以选择向云端发送请求。
- ⑤ 云端服务器接收到请求，处理数据并反馈给终端用户。

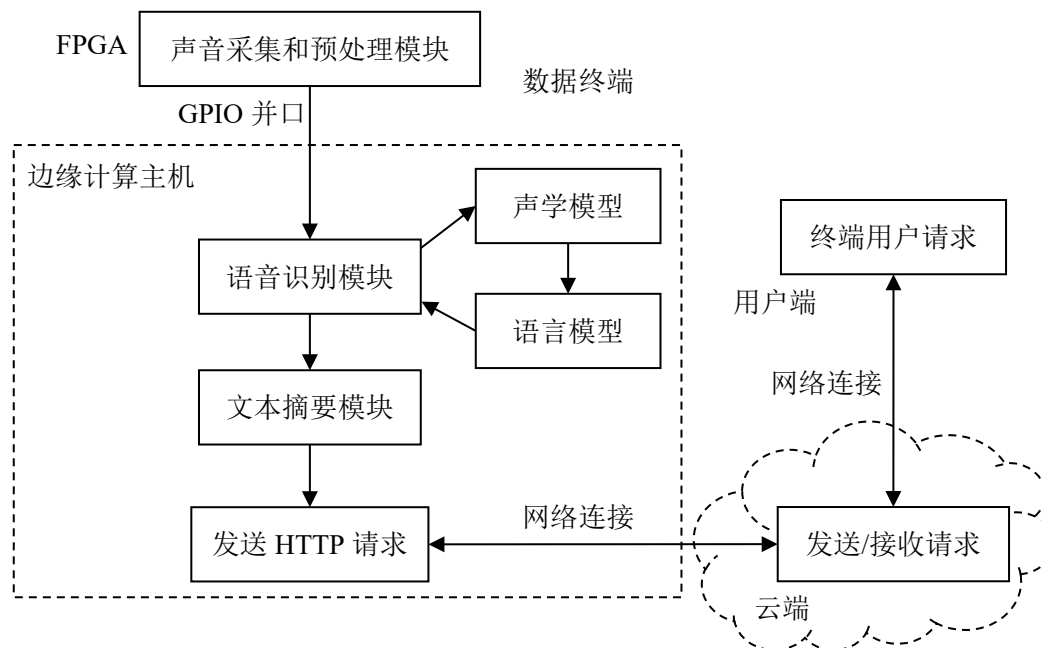


图 2.2-2 整体架构图

## 2.3 技术关键

## 2.4 主要特色

## 2.5 预期目标（技术指标等）

## 2.6 项目实施方案

### 2.6.1 整体方案

### 2.6.2 语音识别模块

- ① 声学信息的预处理

在对语音信号进行分析和处理之前，需要对其进行预加重、分帧、加窗等预处理操作。这些操作的目的是消除因为人类发生器官本身和由于采集语音信号的设备所带来的混叠、高次谐波失真、高频等因素，对语音信号质量的影响。尽可能保证后续语音处理得到的信号更均匀、平滑，为信号参数提供优质的参数，提高语音处理质量。

预加重(Pre-emphasis)是一种在发送端对输入信号高频份量进行补偿的信号处理方式。随着信号速率的增长，信号在传输过程当中受损很大；为了在接收终端能获得比较好的信号波形，就需要对受损的信号进行补偿，预加重技术的思想就是在传输线的始端加强信号的高频成分，以补偿高频份量在传输过程当中的过大衰减。而预加重对噪声并无影响，所以有效地提升了输出信噪比。对于预加重操作，可以采用一阶高通数字滤波器来实现。

分帧(Framing)是为了分析出语音信号每一帧特征参数组成的特征参数时间序列。由于语音信号的短时平稳性，对于语音信号的分析 and 处理需要建立在短时的基础上，进行短时分析。在分帧过程中，通过合理设置帧长和帧移，提取更为细致和丰富的语音信息，并使其更为平滑准确。

加窗是在分帧处理后的预处理操作，窗的目的是对抽样 $n$ 附近的语音波形加以强调而对波形的其余部分加以减弱。对语音信号的各个短段进行处理，实际上就是对各个短段进行某种变换或施以某种运算。用得最多的三种窗函数是矩形窗、汉明窗(Hamming)和汉宁窗(Hanning)。

在此模块，将会首先对采集的语音信号进行必要的预处理，去除静音、噪音并切分，进行预加重、分帧和加窗操作，为后面的声学特征信息提取创造条件。

## ② 特征参数提取模块

一般原始语音信号较为复杂，直接将其作为输入送入到神经网络中，计算复杂度较高且性能较差，因此需要对语音信号进行特征提取。

特征提取是指尽量取出或削减语音信号中与识别无关的信息的影响，减少后续识别阶段需处理的数据量，生成表征语音信号中携带的说话人信息的特征参数。根据语音特征的不同用途，需要提取不同的特征参数，从而保证识别的准确率。

对于常见的语音特征参数，常用的有LPCC和MFCC。LPCC参数是根据声管模型建立的特征参数，主要反映声道响应。MFCC参数是基于人的听觉特性利用人听觉的临界带效应，在Mel标度频率域提取出来的倒谱特征参数。Mel倒谱系数是根据人类听觉系统的特性提出的，模拟人耳对不同频率语音的感知。人耳分辨声音频率的过程就像一种取对数的操作。例如：在Mel频域内，人对音调的感知能力为线性关系，如果两段语音的Mel频率差两倍，则人在感知上也差两倍。

在此模块，通过目前主要的声学特征提取方法，如LPCC、MFCC、CEP等，提取出声音的音色、能量分布、频率分布以及和韵律相关的关于一段时间的局部特征。

这两个模块的具体实现可以使用主机上的音频双向3.5mm接口收集，同时通过进程调度或嵌入主程序运算。但是为了充分发挥机器的性能，可将采样预处理工作独立于主机，并行化计算后，只将需要的样本传输进去，这样可以最大程度地减少数据带宽。这一部分使用FPGA完成，其优势在于芯片的定制化与并行、异步计算。FPGA与音频采集外设将声音信号加工为梅尔特征值后，使用自定义的通讯协议，通过GPIO与主机上的GPIO连接，实现音频信号的计算与传输。

## ③ 声学模型与语言模型建立

在语音识别系统中，使用隐马尔可夫模型(HMM)，用从左向右单向、带自环、带跨越的拓扑结构来对识别基元建模，一个音素就是一个三至五状态的HMM，一个词就是构成词的多个音素的HMM串行起来构成的HMM，而连续语音识别的整个模型就是词和静音组合起来的HMM。

上下文相关建模：协同发音，指的是一个音受前后相邻音的影响而发生变化，从发声机理上看就是人的发声器官在一个音转向另一个音时其特性只能渐变，从而使得后一个音的频谱与其他条件下的频谱产生差异。上下文相关建模方法在建模时考虑了这一影响，从而使模型能更准确地描述语音，只考虑前一音的影响的称为Bi-Phone，考虑前一音和后一音的影响的称为Tri-Phone。

英语的上下文相关建模通常以音素为基元，由于有些音素对其后音素的影响是相似的，因而可以通过音素解码状态的聚类进行模型参数的共享。聚类的结果称为senone。决策树用来实现高效的triphone对senone的对应，通过回答一系列前后音所属类别（元/辅音、清/浊音等等）的问题，最终确定其HMM状态应使用哪个senone。分类回归树CART模型用以进行词到音素的发音标注。

在此模块，充分利用intel的Habana人工智能开发资源进行训练，通过声学模型变换得到音素序列，并通过语言模型的建立，将音素转换为对应的文本。

### 2.6.3 文本摘要模块

在文本摘要模块，将探究抽取式方法和生成式方法两种主要思路。

#### ① 抽取式摘要

抽取式方法直接从原文中选择若干条重要的句子，并对它们进行排序和重组而形成摘要的方法。通常而言，抽取式方法可以分为两大类：无监督抽取式方法和有监督抽取式方法。

无监督抽取式方法不需要平行语料对来进行训练，略去了人工标记语料的繁琐。基于统计层面的，即最大化摘要句子对原始文档的表征能力。在这些方法中，最为著名的TextRank。

TextRank利用局部词汇之间关系（共现窗口）对后续关键词进行排序，直接从文本本身抽取；主要步骤如下：

- 把给定的文本T按照完整句子进行分割。
- 对于每个句子，进行分词和词性标注处理，并过滤掉停用词，只保留指定词性的单词，如名词、动词、形容词。
- 构建候选关键词图 $G = (V, E)$ ，其中V为节点集，由b)生成的候选关键词组成，然后采用共现关系（co-occurrence）构造任两点之间的边，两个节点之间存在边仅当它们对应的词汇在长度为K的窗口中共现，K表示窗口大小，即最多共现K个单词。
- 根据上面公式，迭代传播各节点的权重，直至收敛。
- 对节点权重进行倒序排序，从而得到最重要的T个单词，作为候选关键词。
- 由e)得到最重要的T个单词，在原始文本中进行标记，若形成相邻词组，则组合成多词关键词。例如，文本中有句子“Matlab code for plotting ambiguity function”，如果“Matlab”和“code”均属于候选关键词，则组合成“Matlab code”加入关键词序列。

我们将探索使用TextRank抽取方法，通过把新闻文本分割成若干组成单元（句子），构建节点连接图，用句子之间的相似度作为边的权重，通过循环迭代计算句子的TextRank值，最后抽取排名高的句子组合成文本摘要。

#### ② 生成式摘要

生成式神经网络模型的基本结构主要由编码器（encoder）和解码器（decoder）组成，编码和解码都由神经网络实现。

编码器负责将输入的原文本编码成一个向量C（context），而解码器负责从这个向量C提取重要信息、加工剪辑，生成文本摘要。

这套架构即Sequence-to-Sequence（简称Seq2Seq），广泛应用于存在输入序列和输出序列的场景，比如机器翻译（一种语言序列到另一种语言序列）、image captioning（图片像素

序列到语言序列)、对话机器人(如问题到回答)等。

Seq2Seq架构中的编码器(Encoder)和解码器(Decoder)通常由递归神经网络(RNN)或卷积神经网络(CNN)或者LSTM实现。

我们将研究使用RNN/CNN/LSTM实现Seq2Seq的Encoder & Decoder,观察新闻篇章中生成式摘要与抽取式摘要的差异,并将二者结合起来再次研究模型的文本摘要表现,最终选定文本摘要的最佳方案。

## 2.6.4 网络通信与边缘计算模块

对于网络通信与边缘计算模块,实现在本地边缘计算层完成语音识别与文本摘要计算,将终端计算的结果与云通信,最终将处理后的数据传至云端。

在这一模块,将使边缘设备执行更多的智能算法任务,进行自然语言理解,实时提取语音信息,在云数据中心,算法执行框架更多地执行模型训练的任务,它们的输入是大规模的批量数据集,关注的是训练时的迭代速度、收敛率和框架的可扩展性等;而边缘设备更多地执行预测任务,输入的是实时的小规模数据,由于边缘设备计算资源和存储资源的相对受限性,它们更关注算法执行框架预测时的速度、内存占用量和能效。

我们将构建一个新闻摘要智能生成的计算平台,在边缘计算场景下使边缘设备产生、处理海量数据,有效地进行数据管理、分析与共享,提高计算速度,从而更好地保证新闻的即时性与报道效率。

## 2.7 技术路线

## 2.8 进度安排

4.1-4.5 完成项目设计方案书的初步撰写。

4.10-4.15 对项目的设计方案进行多次修正和改进。

4.16-4.20 实现语音的预处理模块。

4.21-4.25 实现语音的声学特征信息提取。

4.26-4.30 建立声学模型与语言模型,搭建神经网络,完成语音识别子系统的初步设计。

5.1-5.10 探究通过TextRank实现抽取式摘要技术。

5.11-5.15 探究通过HNN实现生成式摘要技术,对比文本摘要两种路线的效果差异与性能指标,选取合适的文本摘要方案。

5.16-5.20 将系统与intel开发平台充分融合,并利用Habana计算资源训练、优化模型。

5.21-5.30 通过边缘计算技术提升数据运算速度与效率。

6.1前 基本设计完成初赛作品。

6.1-6.10 根据作品内容和性能,准备结构清晰,内容完整的PPT。

6.11-6.20 多次模拟现场答辩,争取在分赛区决赛时发挥出最佳答辩水平。

6.21-6.30 再次熟悉设备和所设计的系统,达到最佳的现场演示效果。

7.1-7.31 争取进入全国总决赛。

## 2.9 模板

正文内容

表1 选取组分的热力学性质

组分	H <sub>f</sub> (kcal/mol)	S <sub>f</sub> (kcal/mol)	C <sub>p</sub> (kcal/mol)
A1	100	100	100
A2			
A3			





## 第三部分 团队组成

本部分主要介绍团队情况，包括团队成员组成，团队成员的特长，以及在项目实施中承担的任务分工等。

防疫安全背景下团队组织工作的保障和措施。

## 参考文献

- [1] 缪辉. 基于关键字提取及标点加注技术的口述病历识别系统设计与实现[D].武汉理工大学,2016.
- [2] 郑诗敏. 云环境下流数据关键字的实时查询处理技术研究[D].南京航空航天大学,2016.
- [3] 邓育彬. 基于深度学习的新闻文本情感原因抽取算法研究[D].湖南师范大学,2021.DOI:10.27137/d.cnki.ghusu.2021.002795.
- [4] 张晓丽. 面向新闻领域的关键词提取方法研究及系统实现[D].山西大学,2021.DOI:10.27284/d.cnki.gsxiu.2021.001254.
- [5] 秦宗杰.刍议新媒介技术背景下的即时性新闻[J].媒体融合新观察,2019(05):64-66.
- [6] <https://docs.habana.ai/en/latest>
- [7]