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

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

$$\tilde{x}_N(t) = \sum_{n=-N}^{N} \frac{\omega}{2\pi} \mathrm{e}^{\mathrm{i}n\omega t} \int x(t)\mathrm{e}^{-\mathrm{i}n\omega t}\mathrm{d}t \tag{1.4-1}$$

以及离散傅里叶变换

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

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

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

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

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

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。

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

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

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

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$\underset{Y}{\operatorname{argmax}}\,P(Y|X)$，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

$$P(Y,X)=P(X|Y)P(Y) \tag{1.4-7}$$

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

$$\underset{y}{\operatorname{argmax}}\, P(X|Y) = \underset{y}{\operatorname{argmax}} \sum_{Z} P(X, Z|Y) = \underset{y}{\operatorname{argmax}} \sum_{Z} P(X|Z)P(Z|Y) \tag{1.4-8}$$

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θ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

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

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可夫模型（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”

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

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

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

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Facobook的ABS模型，定义了条件概率

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

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

2.2 系统架构

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

2.7 技术路线

2.8 进度安排

2.9 模板

正文内容

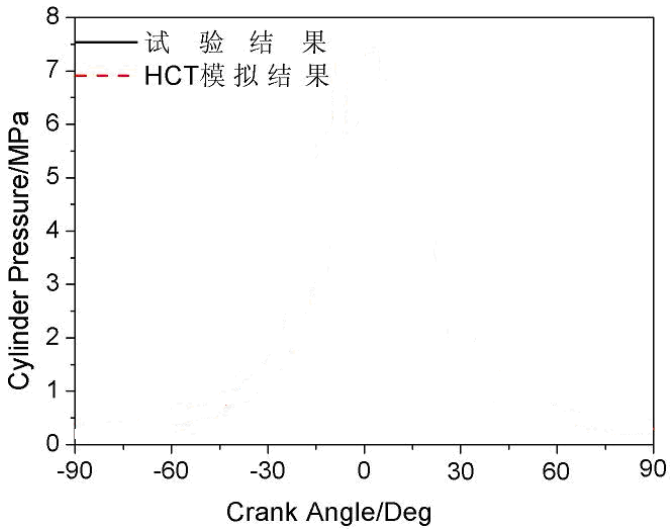

图 2.9-1 气缸压力随曲轴转角变化的曲线

表1 选取组分的热力学性质

组分	H_f (kcal/mol)	S_f (kcal/mol)	C_p (kcal/mol)
A1	100	100	100
A2			
A3			

续表 1

组分	H_f (kcal/mol)	S_f (kcal/mol)	C_p (kcal/mol)

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