基于单片机智能语音识别系统

丁艳玲　黄颖　杨晓飞　贡国庆
南京机电职业技术学院　江苏南京　211306

摘　要：在智能泵站系统中，人机交互功能不可或缺，而语音识别是人机交互功能中至关重要的一环。本设计针对特定泵站控制词汇的语音识别模型，模型满足实际泵站工作的离线单机 CPU 环境运行要求，设计基于深度学习的 CNN+BiLSTM+CTC 语音识别声学模型。采用迁移学习的训练策略，在公开中文语音数据集上训练得到通用中文语音识别声学模型，通过在小数据量的特定词汇数据集上对模型进行微调，提升模型的泛化能力。

关键词：深度学习；人机交互；语音识别

1 语音识别

语音识别技术涉及计算机科学、人类语言学、声学等领域，是实现人机交互的重要一环，现在国内外各大科技公司如百度、讯飞、谷歌以及微软等将多种语言的语音识别率提升到 95% 以上，语音识别技术经历了多次蓬勃发展时期。

目前的语音识别技术已经比较成熟，但是并不适用于实际泵站控制环境，一是泵站控制的安全性要求，要求系统运行环境必须是离线单机 CPU 环境，二是针对泵站控制指令采集的特定词汇语音数据集数据量较少， 对模型的泛化能力要求比较高，三是要求设计的语音识别模型具有一定的可扩展性。

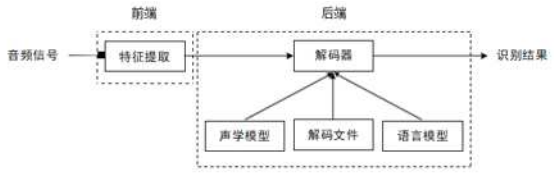

图 1　系统架构

语音识别就是把语音转化为文字。给定一段语音序列 Y，我们想找到与其对应的概率最大的文字序列 W，目前的语音识别系统架构如图 1。

完整的语音识别系统包括两个模块。语音音频预处理以及特征处理模块，负责对语音数据进行特征提取。声音实际上是一种波，但是波形在时域上几乎没有 描述能力，因此需要对声波做特征变换，得到能够表示这段语音数据内容的特征向量，通常的做法是将语音分帧后再特征提取，将语音数据表示为很多帧的多维特征向量，这个过程就是语音的特征提取。

针对实际泵站运行环境， 本文对泵站提供的有限控制命令构建专门的特定指令数据集。录制要求的数据格式如表 1 所示，和公开数据集一致，方便后续模型泛化训练。录制环境符合实际的泵站总控室环境，在安静室内近场在前端浏览器录制。

表 1　泵站控制命令

类型	拼音	汉字
闸门控制	1–9haozhamen kaiqi/guanbi 1–9haojizu qidong/tingzhi 1–9haokaiguan kaiqi/guanbi duanluqi hezha/fenzha jishugongshui qidong/tingzhi	1–9 号闸门 1–9 号机组 1–9 号开关 开启 / 关闭 启动 / 停止 开启 / 关闭 断路器 合闸 / 分闸 技术供水 启动 / 停止
其他控制	dakai/guanbizonghecang zhaomingdeng/fengji dakai/guanbi wushuicang zhaomingdeng/fengji dakai/guanbiranqicang zhaomingdeng/fengji dakai/guanbi dianlicang zhaomingdeng/fengji dakai/guanbi 1–3hao jinggai	打开 / 关闭 打开 / 关闭 打开 / 关闭 打开 / 关闭　综合舱 污水舱 燃气舱 电力舱　照明灯 / 风机 照明灯 / 风机 照明灯 / 风机 照明灯 / 风机 打开 / 关闭 1–3 号井盖

由上表可知，一共 80 条泵站控制指令。对于每一条控制指令，人工录制普通话语音数据 40 条。一共录制 3200 条指令语音数据，按照 7:1 的比例对数据集切分，其中 2800 条用于训练指定词汇语音识别模型，400 条用于测试模型识别效果。

2 系统总体设计

本文实现的智能泵站平台下语音识别系统的整体架构如图 2 所示。

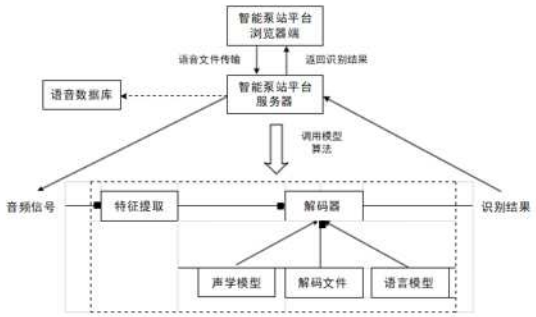

图 2 语音识别系统架构

图中的音频信号由智能泵站平台浏览器端录取，通过 HTTPS 通信协议传递到平台服务器端，调用语音识别模型算法，对音频信号做特征提取得到 Fbank 语 音特征后通过解码器解码。其中解码器由声学模型，解码文件以及语言模型构成， 声学模型将语音特征解码为对应的拼音序列，语言模型将拼音序列解码为汉字序列，算法模型需要调用相关的解码文件。泵站智能平台服务器端获得识别结果后判断是否将音频放入语音数据库，并将识别结果返回到浏览器端。

泵站平台要求实现 B/S(Brower/Server，浏览器和服务器) 架构的语音识别系统，系统必须能在离线环境或者是局域网内实现对特定泵站控制词汇的语音识别。

智能泵站语音识别系统包含智能泵站平台浏览器端语音采集与服务器端通信交互，语音特征处理，语音识别算法等模块。整个系统比较复杂，不同模块之间的交互也比较复杂，比如智能泵站平台服务器端是在 Java 环境下搭建的，浏览器端语音采集需要用到 html 以及 JavaScript 语言，而语音识别模型的实现语言是 Python3。本文用到的开发工具及其版本号如表 2 所示。

表 2 语音识别软件开发环境

编程语言	Python 3.6.7
语音处理	wave 0.0.2
语音特征提取	python speech features 0.6
频域变换	Scipy 1.1.0
数组及矩阵处理	Numpy 1.15.4
可视化工具	matplotlib 3.0.1
深度学习工具	Keras 2.1.3
深度学习工具	Tensorflow 1.10.0
运算平台	CUDA Toolkit 10.0
深度神经网络库	cuDNN 10.0
WEB 应用框架	Flask 1.1.1

整个智能泵站平台系统是 B/S 架构的， 因此需要在浏览器上采集用户输入 的语音控制指令数据，再传输回平台服务器进行识别。语音采集功能包括浏览器端的语音采集界面设计以及调用用户本机麦克风进行录音，基于 JavaScript 和 HTML 进行开发。

2.1 语音识别模型的调用和推断

获取浏览器传回的语音控制命令文件后，在平台服务器端调用模型对语音控制命令进行识别平台服务器是在 Java 环境下开发的，而语音识别模型是在 Python 环境下实现的因此需要跨编程语言调用模型。通常 Java 可以通过 Windows 的控制台语句来调用语音识别模型文件， 然后获取控制台标准输出流来得到识别结果，但是这种方法每次调用模型都要重新加载模型，影响了识别效率，本文在 Python 环境下开发了语音识别的 WEB 服务供平台服务器访问，在开启 WEB 服务时先将模型参数加载在内存中，每次平台服务器调用语音识别服务时无需再次加载模型，直接进行语音指令的推断，大大提升了整个系统的效率。

平台服务器端与浏览器端的通信交互在系统中主要是为了实现语音数据的传输，由于采集语音需要通过 WebRTC 的 getUserMedia 方法获取设备音频输入，具有较高的安全风险，必须使用 HTTPS 通信协议。

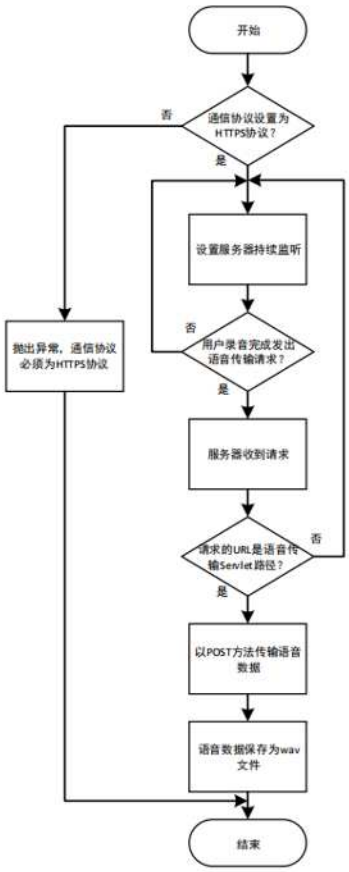

图 3　平台服务器端、浏览器端的通信交互

语音识别模型的 WEB 服务基于 Python 环境的 Flask 框架下开发，设置为 本地的 IP 端口，保证语音识别服务不会被恶意访问，同时实现 HTTP 协议下的 GET 方法来调用模型进行推断。平台服务器端通过 HttpURLConnection 访问语音识别服务，读取到语识别服务返回的识别结果。

2.2 系统测试

系统实际是在泵站平台局域网下运行， 因此在离线单机环境下进行测试，系统运行方案如图 4 所示。测试时，首先在 Python 环境下启动语音识别服务， 加载所需库以及模型参数，然后启动泵站平台服务器，打开浏览器点击录音按钮录制控制指令，查看返回的识别结果。

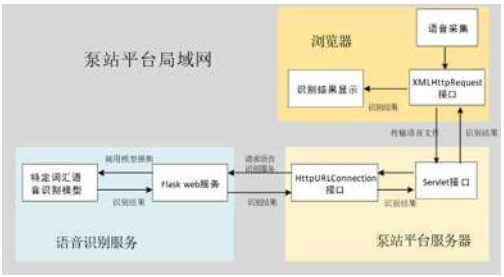

图 4　系统运行方案

3 结束语

本文由于特定词汇数据集数据量较小， 采用迁移学习策略训练 CNN+BiLSTM+CTC 模型， 设计特定词汇的 HMM 语言模型，能够较好的识别特定泵站控制指令，但是模型整体还是比较复杂。后续可以在实际使用后自动采集更多的特定词汇语音数据，设计一个完全由特定词汇数据驱动的轻型的端到端语音识别模型，或者针对本文提出的算法做一定的压缩优化，在保证识别准确率的前提下提升模型的调用推断效率。

参考文献：

[1] 张旭 . 中国泵站工程的现状与发展 [J]. 图书情报导刊 , 2002, 12(4):210–211.

[2] 荆嘉敏 , 刘加 , 刘润生 . 基于 HMM 的语音识别技术在嵌入式系统中的应用 [J]. 电子技术应用 ,2003, 029(10):12–14.

[3] 王海坤 , 潘嘉 , 刘聪 . 语音识别技术的研究进展与展望 [J]. 电信科学 ,2018, 34(2):1–11.

[4] 屈振华 , 李慧云 , 张海涛等 . WebRTC 技术初探 [J]. 电信科学 ,2012,28(10):106–110.

作者简介：丁艳玲 , 女 1978–，汉，吉林榆树人，硕士，副教授，研究方向智能控制。

杨晓飞 , 男 1984–, 汉，江苏泰州人，硕士 , 讲师，研究方向教学。