

中科院声学所离线识别引擎

详细设计说明

版 本 历 史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 起止日期 | 备注 |
| V1.0 | 杜一广 |  | 2021-11-05 | 创建文件 |
|  |  |  |  |  |

目录

[1.概述 3](#_Toc32444)

[1.1编写目的 3](#_Toc842)

[1.2 读者对象 3](#_Toc32443)

[1.3术语定义 4](#_Toc31499)

[2.总体设计 4](#_Toc23650)

[2.1 总体需求 5](#_Toc3831)

[2.2 运行环境 5](#_Toc10012)

[2.2.1 软件环境 5](#_Toc29681)

[2.2.2 硬件环境 6](#_Toc7898)

[2.3离线识别服务总体结构设计 6](#_Toc13910)

[2.3.1离线识别服务总体时序流程 6](#_Toc493)

[3.能力接口模块设计 7](#_Toc6391)

[4.数据设计 10](#_Toc5789)

[4.1 HTTP接口请求 10](#_Toc1227)

[4.3 HTTP接口返回 13](#_Toc13966)

# 1.概述

1

1.1编写目的

本文档主要用于为实现系统功能而进行的系统数据设计说明，具体描述了系统包含的的软件模块的实现流程、功能、接口、数据结构等内容，供项目组开发人员和软件维护人员阅读。

## 1.2 读者对象

本文档的读者对象为离线识别系统的开发人员、测试人员、系统维护人员及接入识别系统的第三方业务人员，通过本文档能够从总体上了解识别系统的架构形式及数据流向。

本说明给出离线识别系统的设计说明，包括最终实现的系统必须满足的功能、性能、接口、附属测试工具程序及设计约束等。

目的在于：

* 为开发人员提供依据；
* 为代码修改、维护提供条件；

## 1.3术语定义

语音识别（Automatic Speech Recognition）。

# 2.总体设计

2

## 2.1 总体需求

功能需求：以语音为处理对象，通过全文转写、场景分割、叠音检测、静音检测、XML服务等技术，让机器把语音转换为相应的文本结果；

接口需求：接口参数简单明了，接口中的每个参数都要有实际意义，保证接口调用流程清晰。

系统验证工具：提供系统完整性，功能正确性的验证工具；

## 2.2 运行环境

### 2.2.1 软件环境

|  |  |  |
| --- | --- | --- |
| 分类 | 名称 | 版本 |
| 操作系统 | Centos | 7.0以上 |
| 数据库 | Redis | 3.2.0 |
| 能力接口 | Tomcat | 9.0.35 |

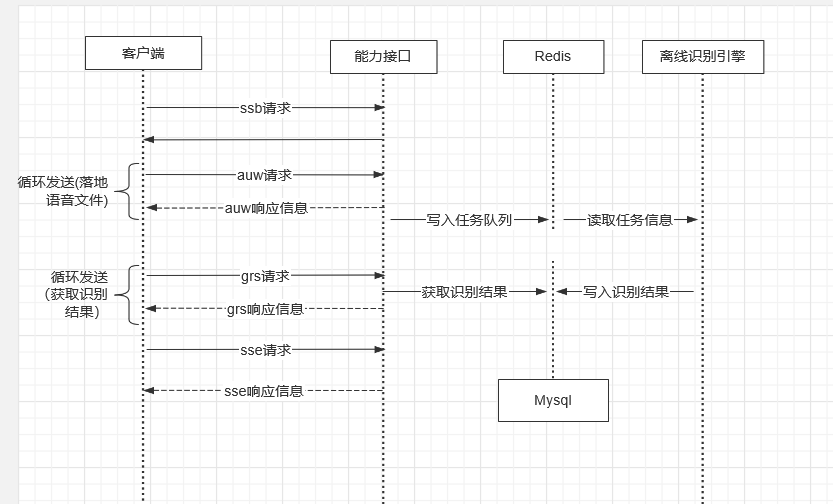
### 2.2.2 硬件环境

|  |  |  |
| --- | --- | --- |
| 服务器 | 最低配置 | 推荐配置 |
| redis服务器 | CPU：1  内存：256M | CPU：1  内存：1G |
| 识别服务器 | CPU：4  内存：10G  磁盘：20G | CPU：10  内存：15G  磁盘：30G |

## 2.3离线识别服务总体结构设计

### 2.3.1离线识别服务总体时序流程

离线识别服务总体时序流程如下所示：

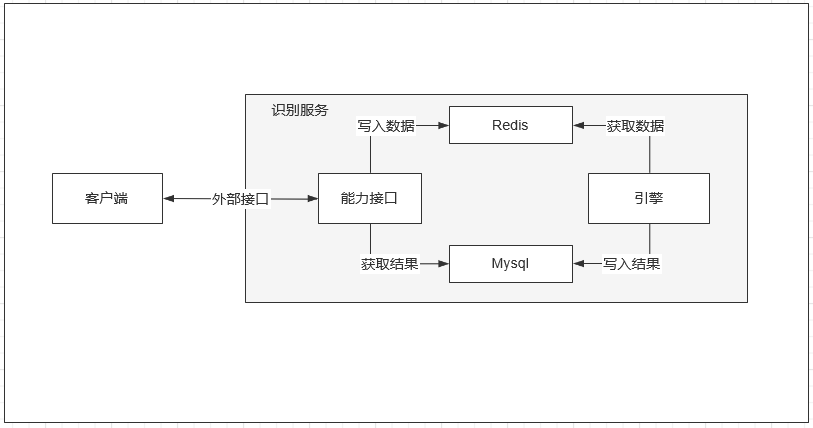


# 3.能力接口模块设计

3

离线识别能力接口通过外部接口与客户端交互，通过Redis与Mysql与引擎进行交互。

模块间关系如下：



**HTTP接口相关函数介绍：**

**public** Object **offline\_stream**(String cmd, String type, String sid, **int** audioStatus, String data, String postfix,String channels,String stereo\_on,String user,String token, **int** urgent)

接口功能：该函数主要通过HTTP协议与客户端进行网络通信的数据交互，接收客户端发送的请求数据，并将识别后的数据返回给客户端。

**内部接口相关介绍：**

Redis接口：将会话信息按照下列格式组装成json字符串，放入redis的List队列，由有空闲线程的引擎取走。

|  |  |  |
| --- | --- | --- |
| 接口参数 | 数据格式 | 参数说明 |
| sid | 字符串 | 请求的唯一标识号 |
| audioName | 字符串 | 保存的语音文件名 |
| audioUrl | 字符串 | 保存语音文件的路径 |
| audioServerIP | 字符串 | 语音文件所在服务器IP |
| channels | 字符串 | 声道数 |
| stereo\_on | 字符串 | 双声道类型 |
| postfix | 字符串 | 语音文件后缀名 |
| isLID | 字符串 | 是否过语种识别引擎 |

Mysql接口：能力接口查询数据库任务会话状态，会话状态为识别完成后，读取对应的识别接口xml文件，通过grs接口返回至客户端，xml格式数据说明如下：



# 4.数据设计

4

## 4.1 HTTP接口请求

|  |  |  |
| --- | --- | --- |
| 接口参数 | 数据格式 | 参数说明 |
| jsonrpc | 字符串 | JsonRpc版本号 固定值字符串“2.0” |
| method | 字符串 | 命令字，固定为“offline\_stream” |
| id | 整数 | JsonRpc规定参数，暂未使用 |
| params | JSON对象 | 方法参数 |
| type | 字符串 | 离线语音识别使用引擎类型 |
| cmd | 字符串 | 命令类型 |
| postfix | 字符串 | 返回结果格式或音频后缀名 |
| user | 字符串 | 鉴权用户名 |
| token | 字符串 | 鉴权密钥 |
| sid | 字符串 | 请求的唯一标识号 |
| data | 字符串 | 音频数据 |
| audioStatus | 整数 | 音频状态 |
| channels | 字符串 | 声道数 |
| stereo\_on | 字符串 | 双声道类型 |
| urgent | 整数 | 任务优先级 |

例如：

|  |
| --- |
| Session begin请求  {  "jsonrpc": "2.0",  "method": "offline\_stream",  "id": 1,  "params": {  "type": "01",  "cmd": "ssb",  "postfix": "sent.txt",  "user": "xxx",  "token": "xxx"  }  } |
| Audio write请求  {  "jsonrpc": "2.0",  "method": "offline\_stream",  "id": 1,  "params": {  "cmd": "auw",  "sid": "4d5660000000",  "type": "1",  "data": "base64编码的音频数据",  "audioStatus": 1,  "postfix": "mp3",  "channels": "0",  "stereo\_on": "4",  "urgent": 1  }  } |
| Get result请求  {  "jsonrpc": "2.0",  "method": "offline\_stream",  "id": 1,  "params": {  "cmd": "grs",  "sid": "4d5660000000",  "type": "1"  }  } |
| Session end请求  {  "jsonrpc": "2.0",  "method": "offline\_stream",  "id": 1,  "params": {  "cmd": "sse",  "sid": "4d5660000000"  }  } |

## 4.3 HTTP接口返回

|  |  |  |
| --- | --- | --- |
| **SSB请求返回** | | |
| 返回参数 | 数据格式 | 参数说明 |
| jsonrpc | 字符串 | JsonRpc版本号 固定值字符串”2.0” |
| id | 整数 | JsonRpc规定参数，暂未使用 |
| result | JSON对象 | 正确返回参数 |
| ret | 整数 | 接口返回值，0表示正常返回 |
| sid | 字符串 | 请求的唯一标识号 |
| 返回示例:  {  "jsonrpc": "2.0",  "id": 1,  "result": {  "ret": 0,  "sid": "4d5660000000"  }  } | | |

|  |  |  |
| --- | --- | --- |
| **AUW请求返回** | | |
| 返回参数 | 数据格式 | 参数说明 |
| jsonrpc | 字符串 | JsonRpc版本号 固定值字符串”2.0” |
| id | 整数 | JsonRpc规定参数，暂未使用 |
| result | JSON对象 | 正确返回参数 |
| msg | 字符串 | 返回消息内容 |
| 返回示例:  {  "jsonrpc": "2.0",  "id": 1,  "result": {  "msg": "data is accepted!"  }  } | | |

|  |  |  |
| --- | --- | --- |
| **GRS请求返回** | | |
| 返回参数 | 数据格式 | 参数说明 |
| jsonrpc | 字符串 | JsonRpc版本号 固定值字符串”2.0” |
| id | 整数 | JsonRpc规定参数，暂未使用 |
| result | JSON对象 | 正确返回参数 |
| ret | 整数 | 接口返回值，0表示正常返回 |
| sid | 字符串 | 请求的唯一标识号 |
| recStatus | 整数 | 离线语音识别结果状态:  2：正在获取识别结果  5：整个会话识别结果获取完成 |
| result | 字符串 | 识别结果 |
| 返回示例:  {  "jsonrpc": "2.0",  "id": 1,  "result": {  "ret": 0,  "sid": "4d5660000000",  "recStatus": 5,  "result": "xml文件内容"  }  } | | |

|  |  |  |
| --- | --- | --- |
| **SSE请求返回** | | |
| 返回参数 | 数据格式 | 参数说明 |
| jsonrpc | 字符串 | JsonRpc版本号 固定值字符串”2.0” |
| id | 整数 | JsonRpc规定参数，暂未使用 |
| result | JSON对象 | 正确返回参数 |
| ret | 整数 | 接口返回值，0表示正常返回 |
| sid | 字符串 | 请求的唯一标识号 |
| 返回示例:  {  "jsonrpc": "2.0",  "id": 1,  "result": {  "ret": 0,  "sid": "4d5660000000"  }  } | | |

|  |  |  |
| --- | --- | --- |
| **错误请求返回** | | |
| 返回参数 | 数据格式 | 参数说明 |
| jsonrpc | 字符串 | JsonRpc版本号 固定值字符串”2.0” |
| id | 整数 | JsonRpc规定参数，暂未使用 |
| error | JSON对象 | 错误返回参数 |
| code | 整数 | jsonRpc错误码 |
| message | 字符串 | jsonRpc 错误消息 |
| data | JSON对象 | 错误返回参数 |
| errno | 整数 | 错误码，见接口文档 |
| errmsg | 字符串 | 错误说明 |
| 返回示例:  {  "jsonrpc": "2.0",  "id": 1,  "error": {  "code": -34000,  "message": "server error",  "data": {  "errno": -32010,  "errmsg": "params cmd must be required!"  }  }  } | | |