

uniapp 安卓腾讯语音识别UTS原生插件

目 录

使用说明	1
使用方法	1.1
更新日志	1.2
枚举类型	1.3
插件方法	2
申请插件所需权限	2.1
一句话识别	2.2
一句话识别授权	2.2.1
查询授权状态	2.2.2
一句话识别监听	2.2.3
取消一句话识别监听	2.2.4
录音文件监听	2.2.5
一句话语音识别	2.2.6
开启内置录音	2.2.7
停止内置录音	2.2.8
开启MediaRecorder录音	2.2.9
停止MediaRecord录音	2.2.10
录音文件识别极速版	2.3
极速文件识别授权	2.3.1
检测极速文件识别授权状态	2.3.2
极速文件识别监听	2.3.3
取消极速文件识别监听	2.3.4
录音文件识别	2.3.5
极速文件识别开始录音识别	2.3.6
停止极速文件录音识别	2.3.7
实时语音识别	2.4
实时语音识别初始化	2.4.1
检测实时语音识别初始化状态	2.4.2
开始实时录音	2.4.3
停止实时录音	2.4.4
取消实时录音	2.4.5
pcm转wav	2.4.6
实时语音识别释放资源	2.4.7

使用方法

介绍

安卓安卓腾讯语音识别UTS原生插件是集成了腾讯语音识别安卓sdk而开发的uniapp(x)插件，插件包含了一句话识别，录音文件急速识别和实时语音识别，欢迎使用

联系作者

插件定制和咨询插件的相关问题可关注微信公众号联系作者，作者将全力协助您使用插件



插件地址

<https://ext.dcloud.net.cn/plugin?id=22974>

演示程序

使用方法

加群下载演示程序



uniapp leven 系列...

群号: 106179987



用法

在需要使用插件的页面加载以下代码

```
import * as module from "@uni_modules/leven-uts-qCloudAsr"
```

示例程序

示例程序页面较多，请通过插件市场插件页面导入示例项目

更新日志

首次发布

枚举类型

声道枚举

状态码	描述
CHANNEL_IN_MONO	-
CHANNEL_IN_BACK	-
CHANNEL_IN_BACK_PROCESSED	-
CHANNEL_IN_DEFAULT	-
CHANNEL_IN_FRONT	-
CHANNEL_IN_FRONT_PROCESSED	-
CHANNEL_IN_LEFT	-
CHANNEL_IN_LEFT_PROCESSED	-
CHANNEL_IN_PRESSURE	-
CHANNEL_IN_RIGHT	-
CHANNEL_IN_RIGHT_PROCESSED	-
CHANNEL_IN_STEREO	-
CHANNEL_IN_VOICE_DNLINK	-
CHANNEL_IN_VOICE_UPLINK	-
CHANNEL_IN_X_AXIS	-
CHANNEL_IN_Y_AXIS	-
CHANNEL_IN_Z_AXIS	-

音频编码类型

状态码	描述
ENCODING_PCM_16BIT	-
ENCODING_AC3	-
ENCODING_AC4	-
ENCODING_E_AC3	-

状态码	描述
ENCODING_MP3	-
ENCODING_AAC_ELD	-
ENCODING_AAC_HE_V1	-
ENCODING_AAC_HE_V2	-
ENCODING_AAC_LC	-
ENCODING_AAC_XHE	-
ENCODING_DEFAULT	-
ENCODING_DOLBY_MAT	-
ENCODING_DOLBY_TRUEHD	-
ENCODING_DRA	-
ENCODING_DSD	-
ENCODING_DTS	-
ENCODING_DTS_HD	-
ENCODING_DTS_HD_MA	-
ENCODING_DTS_UHD_P1	-
ENCODING_DTS_UHD	-
ENCODING_DTS_UHD_P2	-
ENCODING_E_AC3_JOC	-
ENCODING_IEC61937	-
ENCODING_INVALID	-
ENCODING_MPEGH_BL_L3	-
ENCODING_MPEGH_BL_L4	-
ENCODING_MPEGH_LC_L3	-
ENCODING_MPEGH_LC_L4	-
ENCODING_OPUS	-
ENCODING_PCM_8BIT	-
ENCODING_PCM_24BIT_PACKED	-
ENCODING_PCM_32BIT	-

枚举类型

状态码	描述
ENCODING_PCM_FLOAT	-

申请插件所需权限

方法名

requestPermissions

用法

- 用法如下：

```
module.requestPermissions(res => {
  console.log(JSON.stringify(res))
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "grantedList": [
      "android.permission.READ_EXTERNAL_STORAGE",
      "android.permission.WRITE_EXTERNAL_STORAGE"
    ]
  },
  "message": "权限被拒绝",
  "code": -1
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

申请插件所需权限

参数名	参数类型	参数描述
data.grantedList	Array	申请的权限列表，被拒绝的时候会返回
code	Integer	返回类型，0.成功，其他：失败

一句话识别

一句话识别授权

方法名

oneSentenceRecognizeInit

用法

- 用法如下：

```
module.oneSentenceRecognizeInit({  
  appId: "",  
  secretId: "",  
  secretKey: "",  
  token: ""  
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
appId	String	是	无	腾讯申请的appId
secretId	String	是	无	腾讯申请的secretId
secretKey	String	是	无	腾讯申请的secretKey
token	String	否	无	临时授权token

回调

- 示例

```
{  
  "data": {},  
  "message": "",  
  "code": 0;  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

查询授权状态

方法名

oneSentenceRecognizeInitStatus

用法

- 用法如下：

```
module.oneSentenceRecognizeInitStatus(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "status": true
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	Boolean	授权状态，true：已授权，false：未授权
code	Integer	返回类型，0.成功，其他：失败

一句话识别监听

方法名

oneSentenceRecognizeListener

用法

- 用法如下：

```
module.oneSentenceRecognizeListener(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {  
    "type": "startRecord",  
    "result": "",  
    "volumn": 10  
  },  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

参数名	参数类型	参数描述
data.type	String	识别类型，startRecord：开始录音，stopRecord：停止录音，recognizeResult：识别结果，updateVolume：音量改变
data.result	String	识别结果数据
data.volumn	Integer	当前音量值
code	Integer	返回类型，0.成功，其他：失败

取消一句话识别监听

方法名

cancelOneSentenceRecognizeListener

用法

- 用法如下：

```
module.cancelOneSentenceRecognizeListener(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

录音文件监听

方法名

oneSentenceRecognizerAudioPathListener

用法

- 用法如下：

```
module.oneSentenceRecognizerAudioPathListener(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {  
    "audioPath": ""  
  },  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.audioPath	String	录音文件路径
code	Integer	返回类型，0.成功，其他：失败

一句话语音识别

方法名

oneSentenceRecognize

识别结果在监听中回调

用法

- 用法如下：

```
module.oneSentenceRecognize({
  //录音文件地址
  assetFile: "test1.mp3",
  // 0：默认状态 不过滤脏话 1：过滤脏话
  filterDirty: 0,
  //0：默认状态 不过滤语气词 1：过滤部分语气词 2:严格过滤
  filterModal: 0,
  //1：默认状态 根据场景智能转换为阿拉伯数字；0：全部转为中文数字。
  convertNumMode: 0,
  //引擎模型类型，传null或不传默认使用“16k_zh”
  engSerViceType: "16k_zh",
  //0：默认状态 不过滤句末的句号 1：滤句末的句号
  filterPunc: 1,
  // 热词id。用于调用对应的热词表，如果在调用语音识别服务时，不进行单独的热词id设置，自动生成
  hotwordId: "1335468b9e7c11ea9ae9446a2eb5fd98",
  //识别音频的音频格式，支持wav、pcm、ogg-opus、speex、silk、mp3、m4a、aac。
  voiceFormat: "mp3",
  // 开启热词增强功能
  reinforceHotword: 1
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
url	String	否	无	网络语音文件地址
path	String	否	无	本地语音文件路径

参数名	参数类型	是否必填	默认值	参数描述
assetFile	String	否	无	assets目录下语音文件
filterDirty	Integer	否	0	0：默认状态 不过滤脏话 1：过滤脏话
filterModal	Integer	否	无	0：默认状态 不过滤语气词 1：过滤部分语气词 2:严格过滤
convertNumMode	Integer	否	1	1：默认状态 根据场景智能转换为阿拉伯数字； 0：全部转为中文数字。
engServiceType	String	否	16k_zh	引擎模型类型， 传null默认使用“16k_zh”
filterPunc	Integer	否	0	0：默认状态 不过滤句末的句号 1：滤句末的句号
hotwordId	String	否	无	热词id。用于调用对应的热词表，如果在调用语音识别服务时，不进行单独的热词id设置，自动生效默认热词；如果进行了单独的热词id设置，那么将生效单独设置的热词id。

参数名	参数类型	是否必填	默认值	参数描述
voiceFormat	String	否	无	识别音频的音频格式，支持wav、pcm、ogg-opus、speex、silk、mp3、m4a、aac。
reinforceHotword	Integer	否	0	开启热词增强功能，1：开启，0：不开启

url、path和assetFile三个必须传其中一个，不能全部为空

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

开启内置录音

方法名

recognizeWithRecorder

需要录音权限 android.permission.RECORD_AUDIO

用法

- 用法如下：

```
module.recognizeWithRecorder({
    //0 : 默认状态 不过滤脏话 1: 过滤脏话
    filterDirty: 0,
    //0 : 默认状态 不过滤语气词 1: 过滤部分语气词 2:严格过滤
    filterModal: 0,
    //0 : 默认状态 不过滤句末的句号 1: 滤句末的句号
    filterPunc: 1,
    //1: 默认状态 根据场景智能转换为阿拉伯数字; 0: 全部转为中文数字。
    convertNumMode: 1
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filterDirty	Integer	否	0	0 : 默认状态 不过滤脏话 1: 过滤脏话
filterModal	Integer	否	无	0 : 默认状态 不过滤语气词 1: 过滤部分语气词 2:严格过滤
convertNumMode	Integer	否	1	1 : 默认状态 根据场景智能转换为阿拉伯数字; 0: 全部转为中文数字。

参数名	参数类型	是否必填	默认值	参数描述
engSerViceType	String	否	16k_zh	引擎模型类型， 传null默认使用 “16k_zh”
filterPunc	Integer	否	0	0：默认状态 不过滤句末的句号 1：滤句末的句号
hotwordId	String	否	无	热词id。用于调用对应的热词表，如果在调用语音识别服务时，不进行单独的热词id设置，自动生效默认热词；如果进行了单独的热词id设置，那么将生效单独设置的热词id。
reinforceHotword	Integer	否	0	开启热词增强功能，1：开启， 0：不开启

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象

开启内置录音

参数名	参数类型	参数描述
code	Integer	返回类型，0.成功，其他：失败

停止内置录音

方法名

stopRecognizeWithRecorder

停止后识别结果在监听中回调

用法

- 用法如下：

```
module.stopRecognizeWithRecorder(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {},  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

开启MediaRecorder录音

方法名

startMediaRecorder

需要录音权限 android.permission.RECORD_AUDIO

用法

- 用法如下：

```
module.startMediaRecorder({
  //保存文件名称，不传或为空则按时间戳保存
  // filename: "recorder.m4a"
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filename	String	否	无	保存文件名称，不传或为空则按时间戳保存

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示

开启MediaRecorder录音

参数名	参数类型	参数描述
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

停止MediaRecord录音

方法名

stopMediaRecorder

停止后识别结果在监听中回调

用法

- 用法如下：

```
module.stopMediaRecorder({
  //0 ：默认状态 不过滤脏话 1：过滤脏话
  filterDirty: 0,
  //0 ：默认状态 不过滤语气词 1：过滤部分语气词 2:严格过滤
  filterModal: 0,
  //0 ：默认状态 不过滤句末的句号 1：滤句末的句号
  filterPunc: 1,
  //1：默认状态 根据场景智能转换为阿拉伯数字；0：全部转为中文数字。
  convertNumMode: 0,
  //识别音频的音频格式，支持wav、pcm、ogg-opus、speex、silk、mp3、m4a、aac。
  voiceFormat: "m4a",
  //引擎模型类型，传null或不传默认使用“16k_zh”
  engSerViceType: "16k_zh",
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filterDirty	Integer	否	0	0 ：默认状态 不过滤脏话 1：过滤脏话
filterModal	Integer	否	无	0 ：默认状态 不过滤语气词 1：过滤部分语气词 2:严格过滤

参数名	参数类型	是否必填	默认值	参数描述
convertNumMode	Integer	否	1	1：默认状态 根据场景智能转换为阿拉伯数字； 0：全部转为中文数字。
engServiceType	String	否	16k_zh	引擎模型类型， 传null默认使用 “16k_zh”
filterPunc	Integer	否	0	0：默认状态 不过滤句末的句号 1：滤句末的句号
hotwordId	String	否	无	热词id。用于调用对应的热词表，如果在调用语音识别服务时，不进行单独的热词id设置，自动生效默认热词；如果进行了单独的热词id设置，那么将生效单独设置的热词id。
voiceFormat	String	否	无	识别音频的音频格式，支持wav、pcm、ogg-opus、speex、silk、mp3、m4a、aac。
reinforceHotword	Integer	否	0	开启热词增强功能，1：开启， 0：不开启

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

录音文件识别极速版

极速文件识别授权

方法名

fileFlashRecognizeInit

用法

- 用法如下：

```
module.fileFlashRecognizeInit({
  appId: "",
  secretId: "",
  secretKey: "",
  token: ""
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
appId	String	是	无	腾讯申请的appId
secretId	String	是	无	腾讯申请的secretId
secretKey	String	是	无	腾讯申请的secretKey
token	String	否	无	临时授权token

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

检测极速文件识别授权状态

方法名

fileFlashRecognizeInitStatus

用法

- 用法如下：

```
module.fileFlashRecognizeInitStatus(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {
    "status": true
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	Boolean	授权状态，true：已授权，false：未授权
code	Integer	返回类型，0.成功，其他：失败

极速文件识别监听

方法名

fileFlashRecognizeListener

用法

- 用法如下：

```
module.fileFlashRecognizeListener(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {  
    "type": "recognizeResult",  
    "result": ""  
  },  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.type	String	识别类型，listenerSuccess：监听成功，recognizeResult：识别结果

参数名	参数类型	参数描述
data.result	String	识别结果数据
code	Integer	返回类型，0.成功，其他：失败

取消极速文件识别监听

方法名

cancelFileFlashRecognizeListener

用法

- 用法如下：

```
module.cancelFileFlashRecognizeListener(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

录音文件识别

方法名

fileFlashRecognize

识别结果在监听中回调

用法

- 用法如下：

```
module.fileFlashRecognize({
  //录音文件地址
  assetFile: "test1.mp3",
  //识别音频的音频格式，支持wav、pcm、ogg-opus、speex、silk、mp3、m4a、aac。
  voiceFormat: "mp3",
  /*以下参数不设置则使用默认值*/
  //引擎模型类型，传null或不传默认使用“16k_zh”
  // engineModelType: "16k_zh",
  // 0：默认状态 不过滤脏话 1：过滤脏话
  // filterDirty: 0,
  //0：默认状态 不过滤语气词 1：过滤部分语气词 2：严格过滤
  // filterModal: 0,
  //1：默认状态 根据场景智能转换为阿拉伯数字；0：全部转为中文数字。
  // convertNumMode: 0,
  //0：默认状态 不过滤句末的句号 1：滤句末的句号
  // filterPunc: 1,
  //是否开启说话人分离（目前支持中文普通话引擎），默认为0，0：不开启，1：开启。
  // speakerDiarization: 0
  //是否只识别首个声道，默认为1。0：识别所有声道；1：识别首个声道。
  // firstChannelOnly: 1
  //是否显示词级别时间戳，默认为0。0：不显示；1：显示，不包含标点时间戳，2：显示，包含标点时
  // wordInfo: 0
  /**网络超时时间。
   * 注意：如果设置过短的时间，网络超时断开将无法获取到识别结果；
   * 如果网络断开前音频文件已经上传完成，将会消耗该音频时长的识别额度
   * **/
  //单位:毫秒，默认30秒
  // connectTimeout: 30 * 1000
  //单位:毫秒，默认10分钟
  // setReadTimeout: 600 * 1000
  // 开启热词增强功能
  // reinforceHotword: 1
```

```
// sentenceMaxLength: 10
}, res => {})
```

• 参数说明

参数名	参数类型	是否必填	默认值	参数描述
path	String	否	无	本地语音文件路径
assetFile	String	否	无	assets目录下语音文件
filterDirty	Integer	否	0	0：默认状态 不过滤脏话 1：过滤脏话
filterModal	Integer	否	无	0：默认状态 不过滤语气词 1：过滤部分语气词 2:严格过滤
convertNumMode	Integer	否	1	1：默认状态 根据场景智能转换为阿拉伯数字； 0：全部转为中文数字。
engineModelType	String	否	16k_zh	引擎模型类型，传null默认使用“16k_zh”
filterPunc	Integer	否	0	0：默认状态 不过滤句末的句号 1：滤句末的句号
speakerDiarization	Integer	否	0	是否开启说话人分离（目前支持中文普通话引擎），默认为0，0：不开启，1：开启。

参数名	参数类型	是否必填	默认值	参数描述
firstChannelOnly	Integer	否	0	是否只识别首个声道，默认为1。 0：识别所有声道；1：识别首个声道。
wordInfo	Integer	否	0	是否显示词级别时间戳，默认为0。 0：不显示；1：显示，不包含标点时间戳，2：显示，包含标点时间戳。
voiceFormat	String	否	无	识别音频的音频格式，支持wav、pcm、ogg-opus、speex、silk、mp3、m4a、aac。
reinforceHotword	Integer	否	0	开启热词增强功能，1：开启，0：不开启
connectTimeout	Integer	否	30000	连接网络超时时间，单位：毫秒
setReadTimeout	Integer	否	600000	读取网络超时时间，单位：毫秒
sentenceMaxLength	Integer	否	0	单标点最多字数，取值范围：[6，40]

path和assetFile三个必须传其中一个，不能全部为空

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

极速文件识别开始录音识别

方法名

startFileFlashRecognizeRecord

需要录音权限 android.permission.RECORD_AUDIO

用法

- 用法如下：

```
module.startFileFlashRecognizeRecord({
    //临时文件名
    filePrefix: "pcm_temp",
    //临时文件后缀
    fileSuffix: ".pcm"
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filePrefix	String	否	时间戳	临时文件名不包含后缀
fileSuffix	String	否	.pcm	临时文件后缀
sampleRate	Integer	否	16000	采样率
channel	String	否	CHANNEL_IN_MONO	声道，请参考【 声道枚举 】
audioFormat	String	否	ENCODING_PCM_16BIT	音频编码类型，请参考【 音频编码枚举 】

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

停止极速文件录音识别

方法名

stopFileFlashRecognizeRecognizeRecord

停止后识别结果在监听中回调

用法

- 用法如下：

```
module.stopFileFlashRecognizeRecognizeRecord({
  //0 : 默认状态 不过滤脏话 1: 过滤脏话
  filterDirty: 0,
  //0 : 默认状态 不过滤语气词 1: 过滤部分语气词 2:严格过滤
  filterModal: 0,
  //0 : 默认状态 不过滤句末的句号 1: 滤句末的句号
  filterPunc: 1,
  //1: 默认状态 根据场景智能转换为阿拉伯数字; 0: 全部转为中文数字。
  convertNumMode: 0,
  //识别音频的音频格式, 支持wav、pcm、ogg-opus、speex、silk、mp3、m4a、aac。
  voiceFormat: "pcm",
  //引擎模型类型, 传null或不传默认使用“16k_zh”
  engineModelType: "16k_zh",
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
filterDirty	Integer	否	0	0 : 默认状态 不过滤脏话 1: 过滤脏话
filterModal	Integer	否	无	0 : 默认状态 不过滤语气词 1: 过滤部分语气词 2:严格过滤

参数名	参数类型	是否必填	默认值	参数描述
convertNumMode	Integer	否	1	1：默认状态 根据场景智能转换为阿拉伯数字； 0：全部转为中文数字。
engineModelType	String	否	16k_zh	引擎模型类型， 传null默认使用 “16k_zh”
filterPunc	Integer	否	0	0：默认状态 不过滤句末的句号 1：滤句末的句号
speakerDiarization	Integer	否	0	是否开启说话人分离（目前支持中文普通话引擎），默认为0， 0：不开启，1：开启。
firstChannelOnly	Integer	否	0	是否只识别首个声道，默认为1。 0：识别所有声道；1：识别首个声道。
wordInfo	Integer	否	0	是否显示词级别时间戳，默认为0。 0：不显示；1：显示，不包含标点时间戳，2：显示，包含标点时间戳。
voiceFormat	String	否	无	识别音频的音频格式，支持wav、pcm、ogg-opus、speex、silk、mp3、m4a、aac。

参数名	参数类型	是否必填	默认值	参数描述
reinforceHotword	Integer	否	0	开启热词增强功能，1：开启，0：不开启
connectTimeout	Integer	否	30000	连接网络超时时间，单位：毫秒
setReadTimeout	Integer	否	600000	读取网络超时时间，单位：毫秒
sentenceMaxLength	Integer	否	0	单标点最多字数，取值范围：[6，40]

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

实时语音识别

实时语音识别初始化

方法名

realtimeRecognizeInit

用法

- 用法如下：

```
module realtimeRecognizeInit({
  appId: "",
  secretId: "",
  secretKey: "",
  token: ""
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
appId	String	是	无	腾讯申请的appId
secretId	String	是	无	腾讯申请的secretId
secretKey	String	是	无	腾讯申请的secretKey
token	String	否	无	临时授权token
projectId	Integer	否	0	项目id

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

检测实时语音识别初始化状态

方法名

realtimeRecognizeInitStatus

用法

- 用法如下：

```
module.realtimeRecognizeInitStatus(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {  
    "status": true  
  },  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.status	Boolean	授权状态，true：已授权，false：未授权
code	Integer	返回类型，0.成功，其他：失败

开始实时录音

方法名

`startRealtimeRecognizeRecord`

需要录音权限 `android.permission.RECORD_AUDIO`

用法

- 用法如下：

```
module.startRealtimeRecognizeRecord({
    //语音文件保存文件名，为空不保存文件，不需要加扩展名，默认扩展名为.pcm
    saveFileName: this.fileName,
    //初始化识别请求
    audioRecognizeRequest: {
        //采样率，默认：16000
        sampleRate: 16000,
        //声道，默认：CHANNEL_IN_MONO
        channel: "CHANNEL_IN_MONO",
        //使用SDK内置录音器作为数据源
        // isAudioRecordDataSource:true,
        //是否播放声音，非内置录音器有效
        isPlaySound: this.playSound,
        //播放声音的文件，请将文件保存在assets目录下，非内置录音器有效
        assetFile: "test2.mp3",
        //是否消除回声，非内置录音器有效
        aecEnable: this.aecEnable,
        // 设置引擎(16k_zh--通用引擎，支持中文普通话+英文)
        engineModelType: "16k_zh",
        // 0：默认状态 不过滤脏话 1：过滤脏话
        filterDirty: 0,
        // 0：默认状态 不过滤语气词 1：过滤部分语气词 2：严格过滤
        filterModal: 0,
        // 0：默认状态 不过滤句末的句号 1：滤句末的句号
        filterPunc: 0,
        //1：默认状态 根据场景智能转换为阿拉伯数字；0：全部转为中文数字。
        convertNumMode: 1,
        //是否显示词级别时间戳，默认为0。0：不显示；1：显示，不包含标点时间戳，2：显示，包含标点
        wordInfo: 1,
        //自学习模型 id。如果设置了该参数，那么将生效对应的自学习模型，如果您不了解此参数，请不要
        // customizationId: "",
        // 开启热词增强功能
```

```
// reinforceHotword: 1,
//噪音参数阈值，默认为0，取值范围：[-1,1]，对于一些音频片段，取值越大，判定为噪音情况越
// noiseThreshold: 0,
// 强制断句功能，取值范围 5000-90000(单位:毫秒)，默认值0(不开启)。 在连续说话不间断情
// maxSpeakTime: 5000
},
//自定义识别配置
audioRecognizeConfiguration: {
    //分片默认40ms，可设置40-5000，必须为20的整倍数，如果不是，sdk内将自动调整为20的整倍数
    // sliceTime: 40,
    //是否使能静音检测
    silentDetectTimeOut: this.silentSwitch,
    //触发静音超时后是否停止识别，默认为true:停止，setSilentDetectTimeOut为true时参数有
    silentDetectTimeOutAutoStop: true,
    // 静音检测超时可设置>2000ms，setSilentDetectTimeOut为true有效，超过指定时间没有说
    audioFlowSilenceTimeOut: 5000,
    // 音量回调时间，需要大于等于sliceTime，实际时间为sliceTime的倍数，如果小于sliceTime
    minVolumeCallbackTime: 80,
    //是否压缩音频。默认压缩，压缩音频有助于优化弱网或网络不稳定时的识别速度及稳定性
    //SDK历史版本均默认压缩且未提供配置开关，如无特殊需求，建议使用默认值
    isCompress: this.isCompress
}
}, res => {})
```

• 参数说明

参数名	参数类型	是否必填	默认值	参数描述
saveFileName	String	否	时间戳	语音文件保存文件名，为空不保存文件，不需要加扩展名，默认扩展名为.pcm
audioRecognizeRequest	Object	否	无	初始化识别请求
sampleRate	Integer	否	16000	采样率
channel	String	否	CHANNEL_IN_MONO	声道，请参考【声道枚举】
isAudioRecordDataSource	Boolean	否	false	使用SDK内置录音器作为数据源
isPlaySound	Boolean	否	false	是否播放声音，非内置录音器有效

参数名	参数类型	是否必填	默认值	参数描述
assetFile	String	否	无	播放声音的文件，请将文件保存在assets目录下，非内置录音器有效
aecEnable	Boolean	否	false	是否消除回声，非内置录音器有效
filterDirty	Integer	否	0	0：默认状态 不过滤脏话 1：过滤脏话
filterModal	Integer	否	无	0：默认状态 不过滤语气词 1：过滤部分语气词 2:严格过滤
convertNum Mode	Integer	否	1	1：默认状态 根据场景智能转换为阿拉伯数字； 0：全部转为中文数字。
engineModel Type	String	否	16k_zh	引擎模型类型，传null默认使用“16k_zh”
filterPunc	Integer	否	0	0：默认状态 不过滤句末的句号 1：滤句末的句号
wordInfo	Integer	否	0	是否显示词级别时间戳，默认为0。0：不显示； 1：显示，不包含标点时间戳，2：显示，包含标点时间戳。

参数名	参数类型	是否必填	默认值	参数描述
hotwordId	String	否	无	热词id。用于调用对应的热词表，如果在调用语音识别服务时，不进行单独的热词id设置，自动生效默认热词；如果进行了单独的热词id设置，那么将生效单独设置的热词id。
reinforceHotword	Integer	否	0	开启热词增强功能，1：开启，0：不开启
customizationId	String	否	无	自学习模型 id。如果设置了该参数，那么将生效对应的自学习模型,如果您不了解此参数，请不要设置
noiseThreshold	Integer	否	0	噪音参数阈值，默认为0，取值范围：[-1,1]，对于一些音频片段，取值越大，判定为噪音情况越大。取值越小，判定为人声情况越大。

参数名	参数类型	是否必填	默认值	参数描述
maxSpeakTime	Integer	否	0	强制断句功能，取值范围 5000-90000(单位:毫秒)，默认值 0(不开启)。在连续说话不间断情况下，该参数将实现强制断句（此时结果变成稳态，slice_type=2）。如：游戏解说场景，解说员持续不间断解说，无法断句的情况下，将此参数设置为10000，则将在每10秒收到 slice_type=2 的回调。
audioRecognizeConfiguration	Object	否	无	自定义识别配置
sliceTime	Integer	否	40	分片默认40ms，可设置40-5000, 必须为20的整倍数，如果不是，sdk内将自动调整为20的整倍数，例如77将被调整为60，如果您不了解此参数不建议更改
silentDetectTimeOut	Boolean	否	false	是否使能静音检测

参数名	参数类型	是否必填	默认值	参数描述
silentDetectTimeOutAutoStop	Boolean	否	true	触发静音超时后是否停止识别，默认为true:停止，setSilentDetectTimeOut为true时参数有效
audioFlowSilenceTimeOut	Integer	否	2000	静音检测超时可设置>2000ms，setSilentDetectTimeOut为true有效，超过指定时间没有说话超过指定时间没有说话收到onSilentDetectTimeOut回调；需要大于等于sliceTime，实际时间为sliceTime的倍数，如果小于sliceTime，则按sliceTime的时间为准
minVolumeCallbackTime	Integer	否	无	音量回调时间，需要大于等于sliceTime，实际时间为sliceTime的倍数，如果小于sliceTime，则按sliceTime的时间为准

参数名	参数类型	是否必填	默认值	参数描述
isCompress	Boolean	否	false	是否压缩音频。 默认压缩，压缩音频有助于优化弱网或网络不稳定时的识别速度及稳定性，SDK历史版本均默认压缩且未提供配置开关，如无特殊需求，建议使用默认值

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0;
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.type	String	类型，listenerSuccess：录音成功， onAudioRecognizeResult：录音识别结果， onAudioRecognizeState：录音状态

参数名	参数类型	参数描述
data.resultType	String	结果类型，onSliceSuccess：返回分片的识别结果，onSegmentSuccess：返回语音流的识别结果，onSuccess：识别结束回调，返回所有的识别结果，onFailure：识别失败，下面的为录音状态回调，onStartRecord：开始录音，onStopRecord：停止录音，onVoiceVolume：音量状态，onVoiceDb：音量分贝，onSilentDetectTimeOut：静音检测回调
data.result	String	识别结果
data.audioRecognizeRequest	Object	请求数据
data.audioRecognizeRequest.requestId	Integer	请求id
data.audioRecognizeRequest.convertNumMode	Integer	转换数字类型
data.audioRecognizeRequest.customizationId	Integer	自学习模型 id
data.audioRecognizeRequest.engineModelType	Integer	引擎
data.audioRecognizeRequest.extraUserAgent	Integer	-
data.audioRecognizeRequest.filterDirty	Integer	过滤脏话类型
data.audioRecognizeRequest.filterModal	Integer	过滤语气词类型
data.audioRecognizeRequest.filterPunc	Integer	过滤句末的句号类型
data.audioRecognizeRequest.hotWordId	Integer	热词id
data.audioRecognizeRequest.maxSpeakTime	Integer	强制断句功能

参数名	参数类型	参数描述
data.audioRecognizeRequest.needVad	Integer	开启 vad类型
data.audioRecognizeRequest.noiseThreshold	Integer	噪音参数阈值
data.audioRecognizeRequest.reinforceHotWord	Integer	开启热词增强功能
data.audioRecognizeRequest.timestamp	Integer	请求时间戳
data.audioRecognizeRequest.vadSilenceTime	Integer	语音断句检测阈值
data.audioRecognizeRequest.voiceFormat	Integer	识别音频的音频格式
data.audioRecognizeRequest.wordInfo	Integer	是否显示词级别时间戳
data.audioRecognizeResult	Object	识别结果对象
data.audioRecognizeResult.voiceId	String	语音id
data.audioRecognizeResult.text	String	识别结果内容
data.audioRecognizeResult.startTime	Integer	识别开始时间
data.audioRecognizeResult.endTime	Integer	识别结束时间
data.audioRecognizeResult.sliceType	Integer	识别结果类型
data.audioRecognizeResult.code	Integer	状态码，0代表正常，非0值表示发生错误
data.audioRecognizeResult.message	String	错误说明，发生错误时显示这个错误发生的具体原因，随着业务发展或体验优化，此文本可能会经常保持变更或更新
data.audioRecognizeResult.resultJson	String	识别结果JSON数据

参数名	参数类型	参数描述
data.audioRecognizeResult.seq	Integer	该分片所在句子的序号
data.seq	Integer	该分片所在句子的序号
data.response	String	服务端返回的json字符串（如果有）
data.clientException	String	客户端异常
data.serverException	String	服务端异常
data.recordPath	String	录音文件路径
data.volume	String	当前音量
data.voiceDb	String	当前分贝
data.minVoiceDb	String	最小分贝
data.maxVoiceDb	String	最大分贝
code	Integer	返回类型，0.成功，其他：失败

停止实时录音

方法名

stopRealtimeRecognizeRecord

用法

- 用法如下：

```
module.stopRealtimeRecognizeRecord(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

取消实时录音

方法名

cancelAudioRecognize

用法

- 用法如下：

```
module.cancelAudioRecognize(res => {  
  console.log(res)  
});
```

- 参数说明

无

回调

- 示例

```
{  
  "data": {},  
  "message": "",  
  "code": 0  
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败

pcm转wav

方法名

pcm2wav

用法

- 用法如下：

```
module.pcm2wav({
  //录音文件地址
  pcmPath: "",
}, res => {})
```

- 参数说明

参数名	参数类型	是否必填	默认值	参数描述
pcmPath	String	否	无	pcm文件路径

回调

- 示例

```
{
  "data": {
    "path": ""
  },
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
data.path	String	转换后的路径

参数名	参数类型	参数描述
code	Integer	返回类型，0.成功，其他：失败

实时语音识别释放资源

方法名

release

用法

- 用法如下：

```
module.release(res => {
  console.log(res)
});
```

- 参数说明

无

回调

- 示例

```
{
  "data": {},
  "message": "",
  "code": 0
}
```

- 回调说明：

参数名	参数类型	参数描述
message	String	消息提示
data	Object	数据对象
code	Integer	返回类型，0.成功，其他：失败