

百度语音识别 SDK Android 离在线融合版 开发手册

V1.0



北京百度网讯科技有限公司

(版权所有, 翻版必究)

北京百度网讯科技有限公司

目录

目录.....	2
1 概念解释.....	4
2 简介.....	4
2.1 功能介绍.....	4
2.2 兼容性.....	4
2.3 开发包说明.....	5
2.4 总体框图.....	5
3 集成指南.....	5
3.1 注册百度开发者.....	5
3.2 创建应用.....	6
3.3 开通语音识别服务.....	7
3.4 申请离线识别授权.....	8
3.5 导入 SDK.....	9
3.6 配置 manifest.....	9
3.7 权限声明.....	10
4 识别参数.....	11
5 语音识别控件.....	14
6 API 方式识别.....	15
6.1 语音识别器.....	15
6.1.1 创建实例.....	15
6.1.2 设置监听.....	15
6.1.3 开始识别.....	16
6.1.4 停止录音.....	16
6.1.5 取消识别.....	16
6.1.6 销毁.....	16
6.2 监听器.....	16
6.2.1 准备就绪.....	16
6.2.2 开始说话.....	16
6.2.3 音量变化.....	17
6.2.4 原始语音.....	17
6.2.5 说话结束.....	17
6.2.6 识别出错.....	17
6.2.7 识别结果.....	17
6.2.8 临时结果.....	17
6.2.9 识别事件.....	17

7 离线识别参数设置.....	18
7.1 注意事项.....	18
7.2 资源文件设置.....	18
7.3 离线垂类槽数据设置.....	18
8 离线识别支持语法说明.....	19
8.1 说明.....	19
8.2 20000（输入法）.....	19
8.3 10060（地图）.....	19
8.4 10001（音乐）.....	20
8.5 10003（应用）.....	21
8.6 10008（打电话/发短信）.....	22
8.7 100014（联系人）.....	24
8.8 100016（手机设置）.....	25
8.9 100018（电视指令）.....	26
8.10 100019（播放器指令）.....	27
8.11 100020（收音机指令）.....	27
8.12 100021（用户命令）.....	28
9 结果解析.....	29
10 错误码.....	29
10.1 详细错误信息.....	30
10.2 详细错误码.....	30
11 常见问题.....	31
Q1: DEMO 工程无法使用离线识别？.....	31

1 概念解释

对本文中将提到的名词约定如下：

语音识别：也被称为自动语音识别 Automatic Speech Recognition (ASR)，其目标是将人类的语音中的词汇内容转换为计算机可读的输入，例如按键、二进制编码或者字符序列。

自然语言理解：Natural Language Understanding (NLU) 俗称人机对话，人工智能的分支学科。研究用电子计算机模拟人的语言交际过程，使计算机能理解和运用人类社会的自然语言如汉语、英语等，实现人机之间的自然语言通信，以代替人的部分脑力劳动，包括查询资料、解答问题、摘录文献、汇编资料以及一切有关自然语言信息的加工处理。

BDASRv2：语音识别 SDK 离在线融合版的简称，详见下条。

语音识别 SDK 离在线融合版：即本开发包，文中简称为 BDASRv2。BDASRv2 是一个封装了语音采集、语音预处理、云端识别、离线识别等功能的语音识别解决方案。使用 BDASRv2 可以快速在应用程序中集成语音识别功能。

应用程序：在开发中使用了 BDASRv2，具有语音识别功能的产品线产品。

开发者：想使用 BDASRv2 到应用程序中，正在阅读本文档的开发人员。

2 简介

百度语音 Android 版本 SDK 离在线融合版是运行在 Android 平台的一体化语音识别解决方案，以 JAR 包 + 动态链接库的形式发布。基于该方案，开发者可以轻松的构建功能完备、交互性强的语音识别应用程序。

2.1 功能介绍

基本功能：录音，语音数据处理，端点检测、网络通讯和状态通知，返回识别结果。

语音识别控件：集成提示音、音量反馈动效整套交互的对话框控件，方便开发者快速集成。

播放提示音：在录音前后播放提示音，优化用户体验。

监听语音能量：实时反馈用户当前说话声音能量强度。

语义理解：将语音识别成领域相关的语义结果。

本文档适用于对 Android 应用开发有基本了解的开发人员。

2.2 兼容性

系统：支持 Android 2.3 (API Level 9) 及以上系统。需要开发者通过 `minSdkVersion` 来保证支持系统的检测。

机型：手机和平板皆可。

架构：支持 ARM 平台（只提供 armeabi 架构的动态库，可以兼容其他架构）。

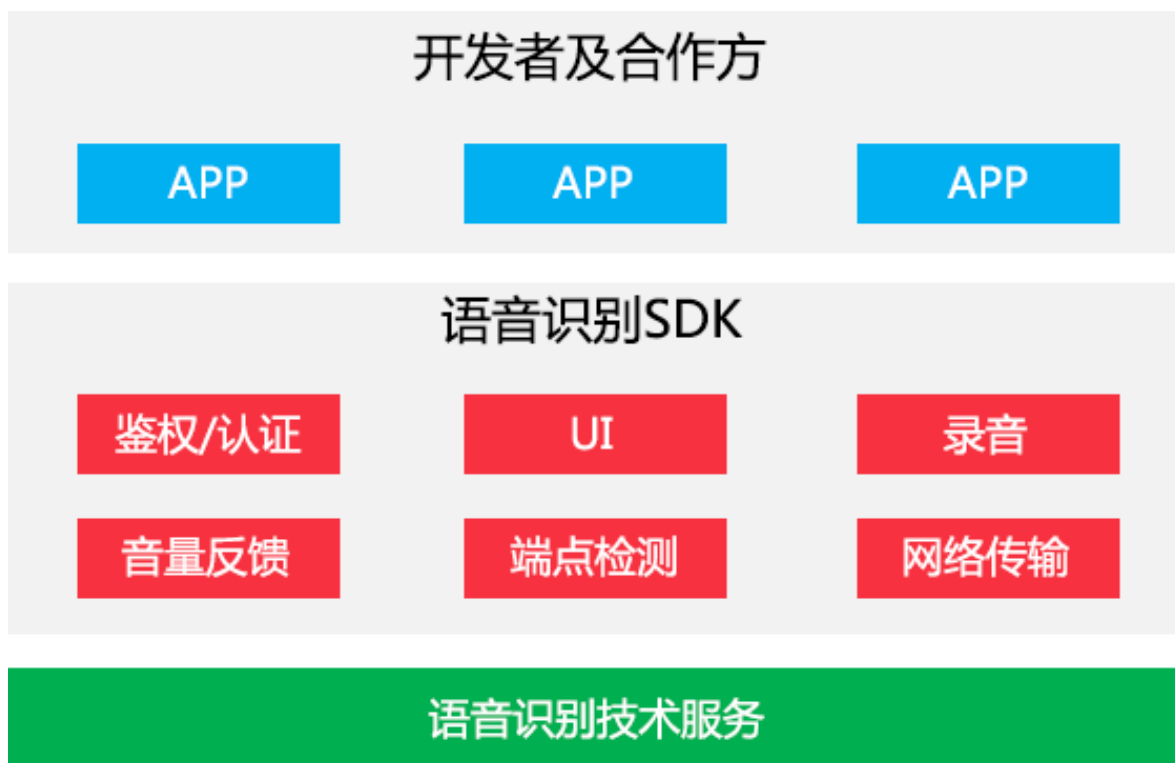
硬件要求：要求设备上有麦克风。

网络：支持 WIFI 及移动网络，移动网络支持使用 NET 网关及 WAP 网关（CMWAP、CTWAP、UNIWAP、3GWAP）。

2.3 开发包说明

文件/文件夹名	说明
/demo/	语音识别 SDK demo 工程
/libs/	包括 so 库及 jar 包。
/res	语音识别对话框资源文件及音效文件，如果不使用对话框可以不集成
/DEMO-VoiceRecognition-2.0.1.apk	Android 示例应用

2.4 总体框图



3 集成指南

3.1 注册百度开发者

使用百度语音技术服务首先需要注册成为百度开发者，如果您在登陆百度帐号后进入“应用管理”出现如下提示，说明您的百度帐号还没有注册成为百度开发者。请按要求填写开发者信息并完成百度开发者的注册。

* 类型: ☒ 个人 ☐ 公司

* 开发者来源: ⓘ

* 开发者姓名:

* 开发者简介:

* Email地址:

* 邮箱验证码:

* 手机号:

* 手机验证码:

3.2 创建应用

在首页登陆已经注册为开发者的百度账号，然后点击页面上方的“应用管理”进入应用管理页面。如果您之前在百度开发者中心创建过应用，则会出现您之前曾经创建过的应用；如果您之前没有创建过应用，则需要新创建一个应用。

- 1、点击右上角 填写应用名称并选择应用垂类信息，即可创建一个新的应用。

创建新应用

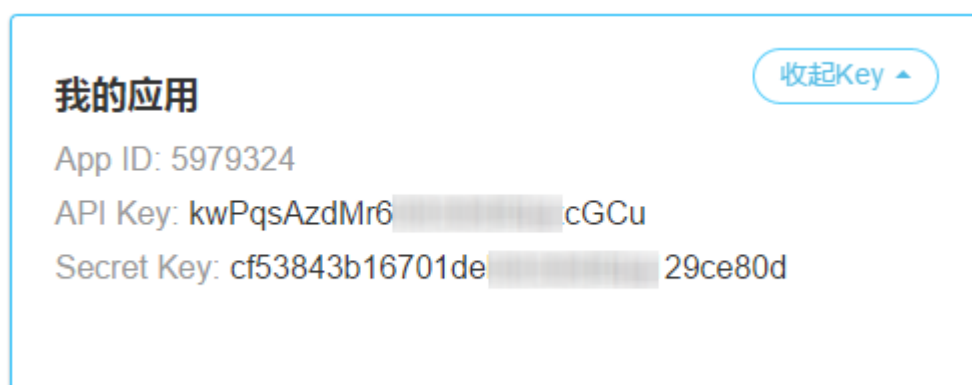
应用名称:

应用类别:

2、创建成功后，新创建的应用会出现在应用列表最上方的位置。



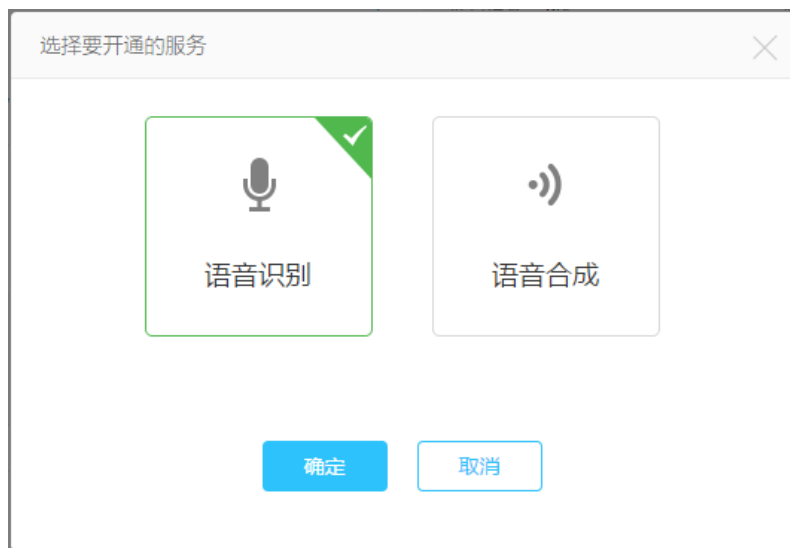
3、点击“查看 Key”可以查看当前应用的 AppID、API Key、Secret Key 信息。



3.3 开通语音识别服务

新创建或从未开通语音识别服务的应用，则需要开通语音识别服务之后才可以使用。

1、点击应用卡片上的  按钮，选择 “语音识别” 进行开通。



2、“语音识别” 服务开通成功后即可获得 50000 次/天 的在线调用次数配额。

3.4 申请离线识别授权

1、如果您使用的是离在线融合语音识别 SDK，并且需要使用离线识别，需要申请离线识别授权。点击“管理语音识别”选择“申请离线识别授权”。



2、选择应用类型为 Android 或者 iOS，填写并提交离线授权所需信息：Android 应用为应用包名，iOS 应用为 Bundle ID。提交成功后即获取离线识别正式授权。

注：请确保填写提交的信息准确，否则会无法通过离线授权验证导致离线识别不可用。

申请离线识别授权

你的应用类型

☒  Android

☐  iOS

请提交您的应用包名值，应用首次连网验证通过即可自动下载离线授权文件

应用包名：

如：com.yourdomain.yourapp

您提交的信息被作为验证密钥使用，请认真填写。提交后如果发现信息有误，可以再次申请更新信息。

提交

取消

暂时没有包名？

您可以先下载临时授权文件，集成到项目中进行开发、调测，临时授权文件的有效期为30天。
在您的应用正式对外发布前，请在左侧填写包名，并确保发版的apk中没有授权文件。

[下载临时授权文件](#)

[去开发文档看详细说明](#)

3、如果暂时没有包名信息，您可以先下载临时授权文件，集成到项目中进行开发、调测，临时授权文件的有效期为 30 天。在您的应用正式对外发布前，请在左侧填写包名信息，并确保发版的 apk 中没有授权文件。

4、离线识别授权信息提交成功后，通过开发环境运行或者打包安装到手机设备上，在设备联网的情况下会自动下载正式授权文件并激活离线识别功能，一定要确保应用包名填写正确。

3.5 导入 SDK

1. 将开发包中的 `libs` 目录合并到工程目录的 `libs` 目录，其中包括了各平台的 `so` 库，开发者视应用需要可以进行删减。
2. 将开发包中的 `res` 目录合并到工程目录的 `res` 目录

3.6 配置 manifest

```
1. ...
2. <uses-permission android:name="android.permission.RECORD_AUDIO" />
3. <uses-permission android:name="android.permission.INTERNET" />
4. <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

```
5.     <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
6.     <uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
7.     <uses-permission android:name="android.permission.READ_PHONE_STATE" />
8.     <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
9.     ...
10.    <application
11.        android:icon="@drawable/ic_launcher"
12.        android:label="@string/app_name">
13.
14.        <!-- begin: baidu speech sdk-->
15.        <!-- 请填写应用实际的 APP_ID -->
16.        <meta-data android:name="com.baidu.speech.APP_ID" android:value="your app id"/>
17.        <!-- 请填写应用实际的 API_KEY -->
18.        <meta-data android:name="com.baidu.speech.API_KEY" android:value="your api key"/>
19.        <!-- 请填写应用实际的 SECRET_KEY -->
20.        <meta-data android:name="com.baidu.speech.SECRET_KEY" android:value="your api secre
    t"/>
21.        <service android:name="com.baidu.speech.VoiceRecognitionService" android:exported="
    false" />
22.        <activity
23.            android:name="com.baidu.voicerecognition.android.ui.BaiduASRDigitalDialog"
24.            android:configChanges="orientation|keyboardHidden|screenLayout"
25.            android:theme="@android:style/Theme.Dialog"
26.            android:exported="false"
27.            android:screenOrientation="portrait">
28.            <intent-filter>
29.                <action android:name="com.baidu.action.RECOGNIZE_SPEECH" />
30.                <category android:name="android.intent.category.DEFAULT" />
31.            </intent-filter>
32.        </activity>
33.        <!-- end : baidu speech sdk-->
34.        ...
35.    </application>
36.    ...
```

3.7 权限声明

名称	用途
android.permission.RECORD_AUDIO	允许应用使用麦克风录音

名称	用途
android.permission.INTERNET	允许应用联网，发送语音数据至服务器，获得识别结果。
android.permission.ACCESS_NETWORK_STATE	获取当前的网络状态，优化录音参数及网络参数。
android.permission.READ_PHONE_STATE	获取用户手机的 IMEI，用来唯一的标识用户。
android.permission.WRITE_SETTINGS	存储用户的标识

4 识别参数

支持的识别参数

参数名	类型	值	描述
appid	int		开放平台创建应用后分配，设置后会覆盖 manifest 中配置的 APP_ID
key	String		开放平台创建应用后分配，设置后会覆盖 manifest 中配置的 API_KEY
secret	String		开放平台创建应用后分配，设置后会覆盖 manifest 中配置的 SECRET_KEY
sample	int	-	采样率
		16000	
		8000	离线暂不支持
sound_start	int	资源 ID	说话开始的提示音

参数名	类型	值	描述
sound_end	int	资源 ID	说话结束的提示音
sound_success	int	资源 ID	识别成功的提示音
sound_error	int	资源 ID	识别出错的提示音
sound_cancel	int	资源 ID	识别取消的提示音
infile	String	音频源	<p>该参数支持设置为：</p> <p>文件系统路径，如：”/sdcard/test/test.pcm”</p> <p>java 资源路径， 如：”res:///com/baidu.test/16k_test.pcm”</p> <p>数据源方法全名，格式 如：”#com.test.Factory.create8kInputStream()” （解释：Factory 类中存在一个返回 InputStream 的方法 create8kInputStream()）</p> <p>注意：必须以井号开始；方法原型必须为：public static InputStream your_method()</p>
outfile	String	文件路径	保存识别过程产生的录音文件
language	String	-	语种
		cmn-Hans-CN	中文普通话
		sichuan-Hans-CN	中文四川话（离线暂不支持）
		yue-Hans-CN	粤语（离线暂不支持）
		en-GB	英语（离线暂不支持）
nlu	String	-	语义解析设置
		disable	禁用语义解析

参数名	类型	值	描述
		enable	开启语义解析
vad	String	-	语音活动检测
		search	搜索模式，适合短句输入
		input	输入模式，适合短信、微博内容等长句输入
asr-base-file-path	String	文件路径	支持离线语音识别的基础包，下载方式参见[集成指南]
license-file-path	String	文件路径	可选，支持离线语音识别的授权文件，下载方式参见[集成指南]，如果不设置，将在联网时尝试在线获取
lm-res-file-path	String	文件路径	支持离线语音识别的附加包，包括输入法和地图资源，下载方式参见[集成指南]
slot-data	String	JSON 字符串	用于自定义离线识别，具体格式参见 demo
prop	int	-	垂直领域
		20000	输入（离线支持）
		10005	热词
		10060	地图（离线支持）
		10001	音乐（离线支持）
		10002	视频
		10003	应用（离线支持）
		10004	网页
		10006	购物
		10007	健康

参数名	类型	值	描述
		10008	电话（离线支持）
		10052	医疗
		10053	汽车
		10054	娱乐餐饮
		10055	财经
		10056	游戏
		10057	菜谱
		10058	助手
		100014	联系人（离线支持）
		100016	手机设置（离线支持）
		100018	电视指令（离线支持）
		100019	播放器指令（离线支持）
		100020	收音机指令（离线支持）
		100021	命令词（离线支持）

5 语音识别控件

语音识别控件提供了整套语音交互、提示音、音量反馈、动效反馈。开发者可以像调起一个普通的 Activity 一样简单的使用语音识别。

```
1. public class DemoActivity extends Activity {  
2.  
3.     private static final int REQUEST_UI = 1;  
4.  
5.     private void start() {  
6.         Intent recognizerIntent = new Intent();
```

```
7.      // recognizerIntent.put("...", "...") TODO 为 recognizerIntent 设置参数，支持的参数见本
      文档的“识别参数”一节
8.      // 为了支持离线识别能力，请参考“离线语音识别参数设置”一节
9.      startActivityForResult(recognizerIntent, REQUEST_UI);
10.   }
11.
12.   @Override
13.   protected void onActivityResult(int requestCode, int resultCode, Intent data) {
14.       super.onActivityResult(requestCode, resultCode, data);
15.       if (resultCode == RESULT_OK) {
16.           ArrayList<String> results = data.getStringArrayList(SpeechRecognizer.RESULTS_RE
               COGNITION);
17.           // data.get... TODO 识别结果包含的信息见本文档的“结果解析”一节
18.       }
19.   }
20. }
```

6 API 方式识别

6.1 语音识别器

6.1.1 创建实例

可以通过 `android.speech.SpeechRecognizer` 的静态方法创建语音识别监听器，如：

```
1.   private android.speech.SpeechRecognizer mSpeechRecognizer;
2.
3.   @Override
4.   protected void onCreate(Bundle savedInstanceState) {
5.       super.onCreate(savedInstanceState);
6.       // ...
7.       mSpeechRecognizer = SpeechRecognizer.createSpeechRecognizer(this, new ComponentName
           (this, VoiceRecognitionService.class));
8.   }
```

6.1.2 设置监听

可以通过设置监听器监听识别过程中产生的消息和最终的识别结果（或错误码），具体见下文中“识别监听器”一节

```
1.      RecognitionListener listener = new RecognitionListener() {
2.          // ..., 略过具体实现
3.      };
4.      mSpeechRecognizer.setRecognitionListener(listener);
```

6.1.3 开始识别

```
1.      Intent recognizerIntent = new Intent();
2.      // recognizerIntent.putExtra ...
3.      // 为了支持离线识别能力，请参考“离线语音识别参数设置”一节
4.      mSpeechRecognizer.startListening(recognizerIntent);
```

6.1.4 停止录音

```
1.      mSpeechRecognizer.stopListening();
```

6.1.5 取消识别

识别取消后，将不再有任何回调返回

```
1.      mSpeechRecognizer.cancel();
```

6.1.6 销毁

```
1.      mSpeechRecognizer.destroy();
```

6.2 监听器

为了监听识别过程，需要创建 `android.speech.RecognitionListener` 的实例，该接口包含如下方法回调

6.2.1 准备就绪

`onReadyForSpeech(Bundle params)`

只有当此方法回调之后才能开始说话，否则会影响识别效果

6.2.2 开始说话

onBeginningOfSpeech()

当用户开始说话后，将会回调此方法

6.2.3 音量变化

onRmsChanged(float rmsdB)

引擎将对每一帧语音进行回调一次改方法

6.2.4 原始语音

onBufferReceived(byte[] buffer)

此方法会被回调多次，**buffer** 是当前帧对应的 PCM 语音数据，拼接后可得到完整的录音数据

6.2.5 说话结束

onEndOfSpeech()

当用户停止说话后，将会回调此方法

6.2.6 识别出错

onError(int error)

错误码详见错误码一节（注意：识别出错将不再回调 **onResults** 方法）

6.2.7 识别结果

onResults(Bundle results)

识别结果中包含：识别结果、原始结果等，具体见结果解析一节

```
1. // 多备选的最优结果
2. ArrayList<String> nbest = results.getStringArrayList("results_recognition");
3. // 原始结果
4. String originResult = results.getString("origin_result");
```

6.2.8 临时结果

onPartialResults(Bundle partialResults)

在最终识别结果返回前，可能会收到多个临时识别结果

6.2.9 识别事件

onEvent(int eventType, Bundle params)

7 离线识别参数设置

7.1 注意事项

由于离线识别仅支持部分识别参数，如果需要在支持离线识别能力，需要确保当前的参数设置被离线所支持（要查看完整的支持列表，请参考[“识别参数”](#)一节）：

```
1.      intent.putExtra("sample", 16000); // 离线仅支持 16000 采样率
2.      intent.putExtra("language", "cmn-Hans-CN"); // 离线仅支持中文普通话
3.      intent.putExtra("prop", 20000); // 输入
4.      //      intent.putExtra("prop", 10060); // 地图
5.      //      intent.putExtra("prop", 10001); // 音乐
6.      //      intent.putExtra("prop", 10003); // 应用
7.      //      intent.putExtra("prop", 10008); // 电话
8.      //      intent.putExtra("prop", 100014); // 联系人
9.      //      intent.putExtra("prop", 100016); // 手机设置
10.     //      intent.putExtra("prop", 100018); // 电视指令
11.     //      intent.putExtra("prop", 100019); // 播放器指令
12.     //      intent.putExtra("prop", 100020); // 收音机指令
13.     //      intent.putExtra("prop", 100021); // 命令词
```

7.2 资源文件设置

```
1.      // value 替换为资源文件实际路径
2.      intent.putExtra("asr-base-file-path", "/path/to/s_1");
3.      // value 替换为 license 文件实际路径，仅在使用临时 license 文件时需要进行设置，如果在[应用管理]
      中开通了离线授权，不需要设置该参数
4.      intent.putExtra("license-file-path", "/path/to/license-tmp-20150530.txt");
5.      if (prop == 10060) {
6.          // 地图类附加资源，value 替换为资源文件实际路径
7.          intent.putExtra("lm-res-file-path", "/path/to/s_2_Navi");
8.      } else if (prop == 20000) {
9.          // 语音输入附加资源，value 替换为资源文件实际路径
10.         intent.putExtra("lm-res-file-path", "/path/to/s_2_InputMethod");
11.     }
```

7.3 离线垂类槽数据设置

```
1.      JSONObject slotData = new JSONObject();
```

```
2.    JSONArray name = new JSONArray().put("张三").put("李四");
3.    JSONArray song = new JSONArray().put("七里香").put("冰雨");
4.    JSONArray artist = new JSONArray().put("周杰伦").put("刘德华");
5.    JSONArray app = new JSONArray().put("手机百度").put("百度地图");
6.    JSONArray usercommand = new JSONArray().put("关灯").put("开门");
7.    try {
8.        slotData.put("name", name);
9.        slotData.put("song", song);
10.       slotData.put("artist", artist);
11.       slotData.put("app", app);
12.       slotData.put("usercommand", usercommand);
13.    } catch (JSONException e) {
14.
15.    }
16.    intent.putExtra("slot-data", slotData.toString());
```

8 离线识别支持语法说明

8.1 说明

本文档列举了离在线融合语音识别离线部分支持的命令说法，并给出示例。说法是由槽组成的，每种命令首先列举出槽，然后说明由这些槽组成的哪些语法；

关于槽内内容，如果数量较少，本文档会一一列举出来，如果数量较多，本文档会举出一些例子。具有“包括”字样的就是将所有列举出来；

本文档为了便于理解，同时是为了说明支持的说法，命名和识别出的 json 结果可能有些出入，请开发者注意。

8.2 20000（输入法）

需要输入法资源文件支持，s_2_InputMethod [下载链接](#)，支持随便说。

8.3 10060（地图）

需要地图资源文件支持，s_2_Navi [下载链接](#)。

槽说明

槽名	说明
Cmd_to	表示寻找目的地的动词（包括导航，导航到，导航去，出发，走起，我要去，带我去，走，去，上，到，我想去，带我到，现在去，改去）
Arrival	任意地名
Nearby	表示附近的介词或者动词（包括附近的，周边的，旁边的，最近的，搜附近的，搜周边的，搜旁边的，搜最近的，我附近的，我周围的，导航去周边的，搜索周边的，我要去周边的，导航去附近的，搜索附近的，我要去附近的，导航去最近的，搜索最近的，我要去最近的）
NearbyCore	表示一类事物的名词，例如加油站，港口等；

支持说法

支持说法	说明举例
cmd_to Arrival	导航到百度大厦
Arrival	用户只说地名，例如 百度大厦
Nearby NearbyCore	附近的 加油站

8.4 10001（音乐）

槽说明

槽名	说明
----	----

槽名	说明
PlayCmd	播放命令（包括播放音乐，放音乐，请播放歌曲，请播放音乐，来一首，我要听，给我来一首，播
Song	歌曲名称，需要开发者传入
Artist	歌手名称，需要开发者传入
Tail	包括 的歌，的专辑

支持说法

支持说法	说明举例
PlayCmd Song	播放《小苹果》
PlayCmd Artist	播放周杰伦的歌

8.5 10003（应用）

槽说明

槽名	说明
App	应用名称，需要开发者传入
LaunchCmd	包括打开，开启，启动
UnInstallCmd	包括 删除，卸掉，卸载，删掉；

支持说法

支持说法	说明举例
App	直接说应用名称，例如，微信
LaunchCmd App	开启 手机百度
UnInstallCmd App	卸载 360 手机助手

8.6 10008（打电话/发短信）

槽说明

槽名	说明
Name	通讯录人名， 需要开发者传入
PhoneNum	电话号码
Call	表示打电话的命令（包括打电话，打个电话，我想打电话，我要打电话，帮我打电话，请打电话，拨号，请拨号，帮我拨号，请帮我拨号，我要拨号）
CallTo	表示“打电话给 XXX”的命令（例如，呼叫，拨打，拨号等）；
FirstCmd	一些前缀（包括给，我想给，我要给，帮我给，请给，请帮我给）；
LastCmd	包括 打电话，打个电话；

槽名	说明
PhoneType	号码类型（包括号码前缀，例如，132，188 等，包括运营商，例如移动联通电信，包括号码类型，例如，座机，手机，手机号码等）
View	包括我要看，看一下，我想看，看看，查看；
ObjectCall	包括没接通电话，未接来电，已接电话，通话记录；
Sms	表示发短信的命令（例如，发短信，发个短信，等等）；
SmsTo	表示“发短信给 XXX”的命令（发短信给，我想发给等）；
SmsLastCmd	包括 发短信，发个短信，发信息，发个信息 ；
ObjectSms	包括 未读短信，已读短信，全部短信；

支持说法

支持说法	说明举例
Call	表示用户打电话的意图，比如，打电话
CallTo PhoneNum	例如 打电话给 10086
CallTo Name	例如打电话给张三

支持说法	说明举例
CallTo Name PhoneType	例如，打电话给张三的移动号码，打电话给李四的 132 号码，等；
FirstCmd Name LastCmd	例如，请给张三打电话；
View ObjectCall	例如，查看未接来电；
Sms	表示用户打电话的意图，比如，发个短信
SmsTo PhoneNum	例如 发短信给 10086
SmsTo Name	例如 发短信给张三
SmsTo Name PhoneType	例如发短信给张三的移动号码，发个短信给李四的 132 号码，等；
FirstCmd SmsLastCmd Name	例如，请给张三发个短信；
View ObjectSms	例如，查看未读短信；

8.7 100014（联系人）

槽说明

槽名	说明
View	包括我要看，看一下，我想看，看看，查看；

槽名	说明
Name	通讯录人名， 需要开发者传入
PhoneType	号码类型（包括号码前缀，例如，132，188 等，包括运营商，例如移动联通电信，包括号码类型，例如，座机，手机，手机号码等）
Create	包括新建，添加，增加，加入，加上
Edit	包括编辑，更改，更新，修改；
Del	删除，删掉，去掉；
Contactator	联系人

支持说法

支持说法	说明举例
View Name PhoneType	查看张三的移动号码
Create (Contactator)	新建（联系人）
Edit Name PhoneType	更改张三的座机
Del Name PhoneType	删除张三的 132 号码

8.8 100016（手机设置）

槽说明

槽名	说明
PhoneSetting	电话设置，包括常用电话设置，包括设定时间，打开 wifi 等常用设置命令

支持说法

支持说法	说明举例
PhoneSetting	

8.9 100018（电视指令）

槽说明

槽名	说明
Change	换台命令，包括调台，切换，换台，我要跳台，我要切换，我要换台，我要看
Number	数字，范围 0~999
TvChannel	电视台，包括北京电视台等常用电视台；
Channel	包括，频道，台；

支持说法

支持说法	说明举例
Change Number Channel	我要看 74 频道

支持说法	说明举例
Change TvChannel	切换河北卫视

8.10 100019（播放器指令）

槽说明

槽名	说明
Setplayer	常用设置命令，包括推出播放，退出，继续播放，继续，暂停播放，暂停，上一首，下一首，设置
Play	包括播放，听，观看
Time	表示时间的槽
From	从

支持说法

支持说法	说明举例
Setplayer	继续播放
From Time Play	从 43 分 32 秒开始播放

8.11 100020（收音机指令）

槽说明

槽名	说明
OpenCmd	常见打开命令，例如，收听电台，听 FM，打开 AM 等
ChangeMode	用于切换 AM/FM 的指令，例如，打开 FM，切换到 AM 等
CloseCmd	关闭收音机，关机
Number	数字
Channel	频道，电台

支持说法

支持说法	说明举例
OpenCmd	打开电台
CloseCmd	关闭收音机
OpenCmd Number (Channel)	收听 FM 八七点五频道
ChangeMode	切换到 FM

8.12 100021（用户命令）

槽说明

槽名	说明
----	----

槽名	说明
UserCommand	需要开发者传入

支持说法

支持说法	说明举例
UserCommand	

9 结果解析

最终识别结果将通过 `onResults(Bundle results)` 方法返回，其中 `results` 支持以下字段：

参数名	类型	描述
results_recognition	ArrayList	识别结果
origin_result	String(JSONObject)	字符串形式的 JSON 结构体，其中包含原始的识别结果信息

10 错误码

错误码	常量名	描述
1	ERROR_NETWORK_TIMEOUT	网络超时
2	ERROR_NETWORK	网络错误
3	ERROR_AUDIO	录音错误

错误码	常量名	描述
4	ERROR_SERVER	服务端错误
5	ERROR_CLIENT	客户端调用错误
6	ERROR_SPEECH_TIMEOUT	超时
7	ERROR_NO_MATCH	没有识别结果
8	ERROR_RECOGNIZER_BUSY	引擎忙
9	ERROR_INSUFFICIENT_PERMISSIONS	缺少权限

10.1 详细错误信息

当语音识别发生错误时，将通过 `RecognitionListener#onError(int error)` 方法返回错误码，并且通过 `RecognitionListener#onEvent(int eventType, Bundle params)` 方法返回详细的错误信息。获取详细的错误信息示例如下：

```
1.  @Override
2.  public void onEvent(int eventType, Bundle params) {
3.      switch (eventType) {
4.          case 11: // eventType == 11 表是返回详细错误信息
5.              String reason = params.get("reason") + "";
6.              print("EVENT_ERROR, " + reason);
7.              break;
8.      }
9.  }
```

借助该方法返回的错误信息，开发者可以更快的发现和调试问题。

10.2 详细错误码

详细错误码	描述
-3002	服务器后端错误
-3003	没有识别结果
-3004	apikey/secretkey 无语音识别权限
-3005	语音质量过低
7051	没有在指定路径找到 s_1（离线识别基础资源包）
7052	没有在指定路径找到 s_2_xxx（prop=20000 或 10060 时对应的附加包）
7055	命令词超过数量限制（10 个）
7056	没有指定 prop 参数
7057	不支持的采样率（仅支持 16000 采样率）
7058	不支持的垂类（参见 “识别参数” ）
7059	设置的离线语义数据无效
7060	不支持的语言类型（参见 “识别参数” ）
7998	离线识别没有得到结果
9005/9101/9xxx	下载授权失败（没有申请离线授权或者提交的信息不正确，参见 “如何验证提交的签名信息是否正确？” ）

11 常见问题

Q1: DEMO 工程无法使用离线识别？

DEMO 工程要使用离线识别，需要注意以下三点：

1. 离线授权
2. 离线数据包
3. 识别参数是否支持离线

离线授权

- 方法一（**推荐**）：在 [应用管理](#) 中申请离线授权：
请参考“[集成指南](#)”一节申请离线授权，并且确认填写的包名或者 bundle ID 是否正确。
- 方法二：下载 30 天有效期的临时 license 文件，存放到设备的任意可读路径，并通过如下方法设置（此处示例为存放于 sdcard 根目录）：

```
1. intent.putExtra("license-file-path", "/sdcard/temp_license_2015-05-19");
```

离线数据包

在语音识别的“[相关下载](#)”里下载离线数据包，存放到设备的任意可读路径，并通过如下方法设置（此处示例为存放于 sdcard 根目录）：

```
1. intent.putExtra("asr-base-file-path", "/sdcard/s_1");
2. if (prop == 10060) { // 地图
3.     intent.putExtra("lm-res-file-path", "/sdcard/s_2_Navi");
4. } else if (prop == 20000) { // 输入法
5.     intent.putExtra("lm-res-file-path", "/sdcard/s_2_InputMethod");
6. }
```

识别参数是否支持离线

请参考“[离线识别参数设置](#)”-“[注意事项](#)”一节，确认当前设置的参数是否支持离线。