

标贝科技虚拟数字人

Android SDK 使用说明文档 (1.0.0)

Name	Date	Reason For Changes	Version
	2021.12.02	创建文档，编写使用说明。	1.0.0

标贝（北京）科技有限公司 DataBaker(Beijing)technology co.,LTD

北京市海淀区西小口路 66 号中关村东升科技园 B-6 号楼 C 座 6 层, 010-58465943

目录

1. Android Studio 集成 SDK（参考 demo）	3
2. SDK 关键类	4
3. 调用说明.....	4
4. 方法说明.....	6
4.1 SmartGuyEngine 基本调用方法说明	6
4.2 OffLineConvertCallBack 回调类方法说明	7
4.3 发生异常时返回的 code 对应表	7

1. Android Studio 集成 SDK（参考 demo）

1.1 在工程下的 build.gradle 文件中添加以下代码。

```
allprojects {  
    repositories {  
        ...  
        maven { url 'https://jitpack.io' }  
    }  
}
```

1.2 将 BakerEngine1.aar、BakerEngine2.aar 文件拷贝到 module 下的 libs 文件夹内，另在该 module 的 build.gradle 文件中添加以下依赖库包。

```
dependencies {  
    implementation fileTree(dir: 'libs', include: ['*.aar'])  
    implementation 'com.github.data-baker.BakerAndroidSdks:bakerasr:2.0.13'  
    implementation 'com.github.data-baker.BakerAndroidSdks:synthesizer:2.0.13'  
    implementation 'com.github.data-baker.BakerAndroidSdks:bakerbasecomponent:2.0.13'  
    implementation 'com.github.data-baker.BakerAndroidSdks:bakerhttp:2.0.13'  
    implementation 'com.squareup.okhttp3:okhttp:4.2.2'  
    implementation 'com.google.code.gson:gson:2.8.6'  
}
```

或者将对应.jar/.aar 包添加至工程 module 下，lib 文件夹里。

注意：SDK 内使用了 okhttp、gson 所以需要开发者添加以这两个库，若工程内已存在，无需重复添加。okhttp 请使用 4.x 版本，以免出现版本兼容 bug。

1.3 在 Module 的 AndroidManifest.xml 文件中 application 节点下添加：

```
android:usesCleartextTraffic="true"
```

1.4 在 Module 的 AndroidManifest.xml 文件中添加权限。6.0 以上需要动态申请 RECORD_AUDIO、READ_PHONE_STATE 权限。

```
<uses-permission android:name="android.permission.RECORD_AUDIO" />  
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />  
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
```

1.5 关于混淆

SDK 中用到了 okhttp 和 gson，所以需要将这两个包的混淆代码添加上。具体混淆代码可以去官方文档上查阅。如果项目中已经有这两个包的混淆代码，不必重复添加。请加上我们 SDK 其他类的混淆代码，如下：

```
-keep class com.baker.smart.guy.lib.fu.** { *; }
-keep class com.faceunity.** {*; }
-keep class com.baker.smart.guy.lib.asr.** { *; }
-keep class com.baker.speech.asr.basic.**{*; }
-keep class com.baker.speech.asr.bean.**{*; }
-keep class com.baker.speech.asr.utils.**{*; }
-keep class com.databaker.synthesizer.bean.** { *; }
-keep class com.baker.smart.guy.lib.callback.SmartGuyEngineCallback{*; }
-keep class com.baker.smart.guy.lib.SmartGuyEngine{*; }
-keep class com.baker.sdk.http.bean.** { *; }
-keep class com.baker.smart.guy.lib.nlp.** { *; }
-keep class com.baker.sdk.basecomponent.util.** {*; }
-keep class com.baker.sdk.basecomponent.bean.** {*; }
-keep class com.baker.sdk.basecomponent.writelog.WriteLog{*; }
-keep class com.baker.sdk.basecomponent.BakerBaseConstants{*; }
```

2. SDK 关键类

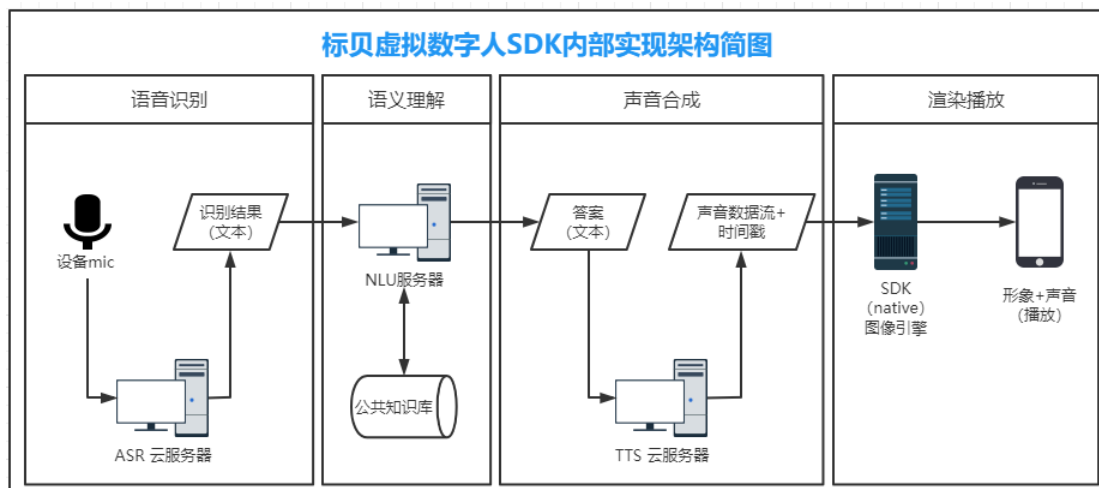
1. **SmartGuyEngine**：虚拟数字人业务处理类，全局单实例。其方法说明请参考表 4.1。
2. **SmartGuyEngineCallback**：虚拟数字人业务回调类。您可以在这些个回调方法中获取到当前各个进度的状态回调、各类数据结果，和错误信息。[注意回调方法都是运行在子线程的](#)。各回调方法说明请参考表 4.2。

3. 调用说明

3.1 虚拟数字人 SDK 架构简图。

SDK 大致分 4 个功能（过程），先开启 mic 进行语音识别，将识别得到的文字送给 nlu 服务去换取答案，再将答案送到语音合成服务进行合成，得到声音数据和时间戳信息送给形象引擎渲染和播放。

SDK 已经将 4 个功能整合成一个整体，开发者只需要开启和结束第一步，后面流程 sdk 内部会自动流转执行，并将各阶段进度和数据通过回调方法同步。



3.2 初始化。请在 app 启动后尽可能提前执行以下初始化代码，sdk 初始化会执行相关网络耗时操作，建议为 sdk 业务提前做好准备。Demo 中是放在了 application 内执行。Tts、Asr 的 ClientId、Secret 等授权信息请在标贝开放平台 (<https://ai.data-baker.com/>)获取。

//初始化引擎基础库

SmartGuyEngine.getInstance().initEngine(BaseApplication.this);

//设置 tts 授权，并获取授权

SmartGuyEngine.getInstance().initTts(ttsClientId, ttsSecret);

//设置 asr 授权，并获取授权

SmartGuyEngine.getInstance().initAsr(asrClientId, asrSecret);

3.2 在形象展示页获取引擎渲染落地控件，并进行设置，参考 MainActivity 中 onCreate 方法。

SmartGuyEngine.getInstance().initEngineView(GLTextureView mGLSurfaceView);

同步设置引擎的生命周期方法，参考 MainActivity 中的 onResume()、 onPause()、 onDestroy():

@Override

protected void onResume() {

super.onResume();

SmartGuyEngine.getInstance().onResume();

}

@Override

protected void onPause() {

super.onPause();

SmartGuyEngine.getInstance().onPause();

}

```
@Override
protected void onDestroy() {
    super.onDestroy();
    SmartGuyEngine.getInstance().onDestroy();
}
```

3.3 开启语音识别，同时设置回调对象。回调方法 `asrOnReadyOfSpeech()` 回调时，代表开启成功，并可以说话进行语音识别，如果错误会回调 `onError` 方法，并给出错误信息。语音合成音色名请用开放平台上 `tts` 已授权并支持的音色。

```
SmartGuyEngine.getInstance().startTalk(callback, voiceName);
```

3.4 结束语音识别，关闭 mic。直接进入 SDK 后续流程。

```
SmartGuyEngine.getInstance().stopTalk();
```

3.5 如果在同一个页面多次问答，请重复 3.3-3.4 步骤。引擎是单实例，同一时间只允许 1 个页面渲染，如果退出此页面在其他页面实现此功能，请实现 3.2-3.4 步骤。关于 `SmartGuyEngine` 类其他方法请参考表 4.1。

4. 方法说明

4.1 SmartGuyEngine 基本调用方法说明

方法名	方法作用	说明
<code>initEngine(Context context)</code>	初始化引擎	<code>SmartGuyEngine</code> 类是单实例，此方法全局只需要调用一次，请尽可能在使用前提前调用，提前做好准备。
<code>initTts(String ttsClientId, String ttsSecret)</code>	设置 <code>tts</code> 授权信息，并获取授权	此方法会进行网络请求，全局只需要调用 1 次，请尽可能提前调用，提前做好准备。
<code>initAsr(String asrClientId, String asrSecret)</code>	设置 <code>asr</code> 授权信息，并获取授权	此方法会进行网络请求，全局只需要调用 1 次，请尽可能提前调用，提前做好准备。
<code>initEngineView(GLTextureView glSurfaceView)</code>	初始化引擎渲染落地控件	参考 <code>activity_main.xml</code> 布局文件，此控件是 <code>sdk</code> 自定义控件，需要用 <code>com.faceunity.ui.GLTextureView</code> 控件，目的是展示形象。
<code>startTalk(SmartGuyEngineCallback callback, mVoiceName)</code>	开启 <code>asr</code> 识别，开启任务	<code>Sdk</code> 内部流程做了全封装，只需要调用此方法开启任务。合成音色名请用开放平台上 <code>tts</code> 已授权并支持的音色。
<code>stopTalk()</code>	关闭 <code>asr</code> 录音，进行识别	<code>SDK</code> 未加 <code>VAD</code> 功能，所以需要调用此方法结束说话，后续流程会自动进行。

switchBundle(int bundleId)	切换形象，取值 0-2	目前形象数据设置了 3 个，切换形象用以用此方法。参数 bundleId 意义 0=红衣服小姐姐， 1=卡通女， 2=卡通男
setTtsVoiceName(String voiceName)	设置 tts 音色名，默认 Lingling	如果要设置最终合成时的声音效果，可以通过此方法设置，请在开启任务之前设置此参数。
stopPlay()	停止音频播放&形象口型播放	播放过程中若需要提前结束播放，可以调用此方法。
stopTask()	停止任务	如果需要取消任务，或停止任务，可以调用此方法。
onResume()	恢复形象动画等资源	引擎生命周期函数，请在页面对应生命周期中调用此方法。（必须调用）
onPause()	暂停形象动画等资源	引擎生命周期函数，请在页面对应生命周期中调用此方法。（必须调用）
onDestroy()	停止任务&释放形象动画等资源	引擎生命周期函数，请在页面对应生命周期中调用此方法。（必须调用）

4.2 SmartGuyEngineCallback 回调类方法说明

方法	作用	说明
asrOnReadyOfSpeech()	Mic 准备就绪，可以开始说话	可以在此方法回调后，唤起页面的语音识别 UI 交互效果等。所有回调方法均运行在子线程中。
asrOnVolumeChanged(int volume)	识别过程中的分贝值回调	这个回调会返回识别过程中声音的分贝值，返回时间间隔 100ms。可以根据此值绘制音轨图等效果。
asrOnResult(String bestText, boolean isLast)	实时返回识别结果（文本），会有多次回调。	当调用 stopTalk()之后，此方法会回调给出此次识别的最终结果。isLast=true 表示是最终结果。
nlpOnAnswer(String answerText)	将识别结果送给 nlu 服务器之后得到的答案	这个过程是 sdk 内部自动处理的，会将 nlu 之后的答案回调，用于页面展示提示等作用。合成播报也是这段文本内容。
onResumeByCloseEverything()	如果 sdk 内部出现适合恢复到默认状态的情况时，回调通知恢复到默认状态。	SDK 某些业务流程，会出现需要恢复到默认状态的情况，会通过此回调通知调用者恢复 UI 等到默认状态。比如：问答对话之后的结束对话场景。
onError(String errorCode, String errorMsg)	任务错误信息回调	String code 是错误码，错误码具体请参考以下错误码对照表。String message 是具体的错误信息日志。

4.3 发生异常时返回的 code 对应表

错误码	含义	类别
11180001	缺少 ClientId	TTS 相关错误码
11180002	缺少 Secret	
11180003	token 获取失败	
11180004	合成文本内容为空	
11180005	发音人参数错误	
11180006	返回结果解析错误	
11180007	合成失败，失败信息相关错误	
11180008	播放器相关错误	
11180009	合成文本内容转码错误	
11180010	返回结果解析错误;返回 null	
13180001	缺少 ClientId	ASR 相关错误码
13180002	缺少 Secret	
13180003	token 获取失败	
13180004	没有录音权限	
13180005	本地网络不可用	
13180006	网络未准备好，没有调用 start 方法	
13180007	录音尚未初始化	
13180008	正在录音	
13180009	发送的数据为空	
13180010	response is null	
13180011	gson to object error	
13180012	websocket 发送消息出错	
13180013	websocket onFailure error	
13180014	录音意外中断	
18180001	Nlu 相关错误信息	Nlu 相关错误信息
18180002	形象渲染播放相关错误信息	形象渲染播放相关错误信息
18180003	合成声音音色值为空	
	服务引擎等其他错误	请详看错误信息