

# 【安卓端SDK】 AI智能体接口(对外版本)

## (一) 背景

本文档描述AI智能体接口文档，本SDK是用于正式对接版本；

### 1.1 账号信息

app\_id (仅仅用于本次测试，不能线上使用)： apppibr915pu75m

token： 可以默认输入 "no\_token"

正式版本app\_id 需要在百度智能云控制台开通RTC服务申请；

地址：<https://cloud.baidu.com/doc/RTC/index.html>

## 1.2 测试环境地址

沙盒地址：<https://ai.agent.kaywang.cn/>

正式环境地址后续RD给出

创建智能体：</api/v1/aiagent/generateAIAgentCall>

**【Demo演示环境客户端调用创建智能体，正式环境需要业务服务器调用该接口】**

请求参数，**config 支持是json string; 当前支持配置如下；**

1	名称	类型	是否必填	示例值	描述
---	----	----	------	-----	----

2	app_id	String	是	xxx	BRTC应用ID
3	quick_start	bool	否	true	快速启动（不需要调用start方法） true
4	config	String	否	{  "lang": "zh", "location": "北京市海淀区", "tts_sayhi": "主动打招呼的话，“你好！”， "role": "你是旅游达人", "audiocodec": "opus" }	ai instance配置  lang: zh 中文， en 英文 location: 位置信息 tts_sayhi: 联通后首次播报内容 role: 人设 audiocodec: 音频编码器， 低延迟 pcmu

## 1.3 SDK下载地址

最新版本SDK 参考提供最新源码包;

gradle 依赖SDK

```
</>Java
1
2 implementation 'com.baidubce.mediasdk:brtc:3.4.0.8a'
3 implementation 'com.baidubce.mediasdk:lib_agent:1.0.0.6'
4
```

手动下载SDK

```
</>Java
1
```

```
2 https://repo1.maven.org/maven2/com/baidubce/mediasdk/brtc/3.4.0.8a/brtc-3.4.0.8a.aar
3
4 https://repo1.maven.org/maven2/com/baidubce/mediasdk/lib_agent/1.0.0.6/lib_agent-1.0
  .0.6.aar
5
6
```

## (二) SDK接口封装

### 2.1 初始化

## &lt;/&gt; 初始化AIEngine

Java

```
1 public static class AIEngineParams {
2     /** appid */
3     public String appId = "";
4     /**  workflow类型, 默认用voiceChat */
5     public String workflow = Constants.WorkFlowType.VOICE_CHAT;
6     /** 实例id  服务器端下发, 当前调试默认即可 */
7     public long aiEngineInstanceId = 0;
8     /** 服务器端接口下发 cid, token,  当前调试传入空即可 { "cid": 1, "token": "xxx-aa-
    bb" }*/
9     public String context = "";
10    /** 调试开关 */
11    public boolean verbose = false;
12    /** 开启终端侧TTS */
13    public boolean enableTerminalTTS = false;
14    /** dump 采集音频 */
15    public boolean dumpAudioInput = false;
16
```

```
17      /**容许外部输入音频 */
18      public boolean enableExternalAudioInput = false;
19
20      /**容许外部输出音频 */
21      public boolean enableExternalAudioOutput = false;
22
23      /** 音频source */
24      public int inputAudioSource = -1;
25
26      /** 软件回音消除 */
27      public boolean forceSoftEchoCancellation = false;
28
29      /** 回声消除等级 */
30      public int aecLevel = -1;
31
32      /** 禁止蓝牙设备 */
33      public boolean disableBluetooth = false;
34      public String audioCodec = RtcParameterSettings.AudioCodecId.OPUS;
35
36      /**
```

```
37      * 海外服务器 配置，中国大陆使用不用填该配置；私有化环境不用配置；
38      * 欧美 {@link Constants.AgentRegionType#europeAmerica }
39      * */
40      public String regionServer;
41      /** 授权key 信息 */
42      public String licenseKey;
43      /** 用户id */
44      public String userId;
45
46      /** 自定义信令服务器地址 */
47      public String signalServerUrl;
48  }
49
50  public static AIAgentEngineImpl init(Context context, AIAgentEngineParams params)
```

## 2.2 开启对话



&lt;/&gt;

Java

```
1  /**
2   * 开启对话
3   * @param token 鉴权token
4   * @param instanceId 实例id
5   */
6   public abstract void call(String token, long instanceId);
```

## 2.3 结束对话

&lt;/&gt;

Java

```
1   public abstract void hangup();
```

## 2.4 设置回调

&lt;/&gt;

Java

```
1      public abstract void setCallback(AIAgentEngineCallback callback);
```

## 2.5 发送文本

&lt;/&gt;

Java

```
1      /** 发送文本给智能体 */  
2      public abstract void setTextToAIAgent(String text);  
3  
4      /** 发送文本给智能体，并且打断当前播报 */
```

```
5    public abstract void sendTextToAIAgentAndInterrupt(String text);
6
7    /** 发送文本给TTS, 直接进行播报 */
8    public abstract void sendTextToTTS(String text);
```

## 2.6 声音配置

&lt;/&gt;

Java

```
1    /** 静音播放
2     * @param isMute 是否静音
3     * */
4    public abstract void mutePlayback(boolean isMute);
5
6    /** 静音mic
7     * @param isMute 是否静音
8     * */
```

```
9      public abstract void muteMic(boolean isMute);
10
11     /** 切换扬声器
12      * @param speaker 是否扬声器
13      * */
14     public abstract void switchToSpeaker(boolean speaker);
```

## 2.7 主动打断

&lt;/&gt;

Java

```
1      public abstract void interrupt();
```

## 2.8 销毁

&lt;/&gt;

Java

```
1    public abstract void destroy();
```

## 2.9 注册外部TTS

&lt;/&gt;

Java

```
1    /** TTS 服务 */
2    public static interface TTSService {
3        /** 停止播放TTS, 打断语音TTS播报时候调用,建议mute speaks 1000ms */
4        public void onStop();
```

```
5
6     /** 开始TTS */
7     public void onStart(String msg);
8 }
9
10 /**
11  * 注册外部TTS服务
12  * @param service 外部TTS服务
13  * */
14 public abstract void registerExternalTTS(Constants.TTSService service);
```

## 2.10 发送FunctionCall结果



Java

```
1  /**
2   * 发送functionCall 结果
3   * @param id 唯一标识
4   * @param result 结果 ， 例如: {"result":"ok"}
5   * */
6  public abstract void sendFunctionCallResult(String id, String result);
```

## 2.11 上传文件

### 注意

暂时仅仅支持 JPEG/PNG/JPG 三种格式图片文件，并且文件大小不超过7.5Mb；

文件上传成功后会有回调消息 **AIAgentEngineCallback#onUploadFileStatus**

```
</> java
1      /**
2      * 上传文件
3      * @param path 文件路径，必须有读权；
4      * @param expire 过期时间 单位秒，最大值默认180秒（图片支持多轮次会话引用），最小值0（图
   片仅仅支持一轮会话引用）；
5      * */
6      public abstract boolean uploadFile(String path, int expire);
```

## 2.12 音频外部采集

接入方负责将音频输入PCM 输入进入（当前要求16K音频采样，单声道，  
AudioFormat.**ENCODING\_PCM\_16BIT**）

```
</> Bash
```



```
1      /**
2      * 外部输入音频，音频要求输入长度10ms，320字节长度
3      * {@link AIAgentEngine.AIAgentEngineParams#enableExternalAudioInput} 打开音频外部
   采集后可以使用该接口；
4      * @param data 音频数据
5      * @param timestamp 时间戳
6      * @param sampleRate 采样率
7      * @param channels 声道数
8      * */
9      public abstract void pushAudioFrame(byte[] data, long timestamp, int sampleRate,
   int channels);
10
11     /**
12     * 外部输入音频，音频要求输入长度10ms，320字节长度
13     * {@link AIAgentEngine.AIAgentEngineParams#enableExternalAudioInput} 打开音频外部
   采集后可以使用该接口；
14     * @param data 音频数据
15     * @param timestamp 时间戳
16     * @param sampleRate 采样率
17     * @param channels 声道数
```

```
18      * */
19      public abstract void pushAudioFrame(ByteBuffer data, long timestamp, int
sampleRate, int channels);
20
21      /**
22       * 设置云渲染 view 参数
23       * @param viewGroup 父类容器
24       * @param params 参数
25       * @param type 缩放模式， 常用的两种模式；RTCVideoView.ScalingType.SCALE_ASPECT_FIT
| RTCVideoView.ScalingType.SCALE_ASPECT_FILL
26       * */
27      public abstract void setCloudRenderDisplay(ViewGroup viewGroup,
ViewGroup.LayoutParams params, RTCVideoView.ScalingType type);
28
29
30
31      /**
32       * 设置数字人 view 参数
33       * @param viewGroup 父类容器
34       * @param params 参数
```

```
35      * @param type 缩放模式，常用的两种模式；RTCVideoView.ScalingType.SCALE_ASPECT_FIT  
    |   RTCVideoView.ScalingType.SCALE_ASPECT_FILL  
36      * */  
37      public abstract void setDigitalDisplay(ViewGroup viewGroup,  
    |   ViewGroup.LayoutParams params, RTCVideoView.ScalingType type);  
38  
39  
40  
41
```

### (三) 事件回调

## 3.1 AIAgent回调

&lt;/&gt;

Java

```
1
2  /** 智能体音频状态 */
3  @Retention(RetentionPolicy.SOURCE)
4  public @interface AIAgentAudioStateType {
5      /** 大模型停止说话 */
6      int STOPPED = 1;
7      /** 大模型说话 */
8      int SPEAKING = 2;
9  }
10 /** 用户音频状态 */
11 @Retention(RetentionPolicy.SOURCE)
12 public @interface UserAudioStateType {
13     /** 用户停止说话*/
14     int STOPPED = 1;
15     /** 用户说话*/
```

```
16         int SPEAKING = 2;
17     }
18
19
20 /** AIAgentEngineCallback 事件回调 */
21
22 public abstract class AIAgentEngineCallback {
23
24     /**
25      * 错误回调，出错后需要结束通话。
26      * @param error 错误码
27      * @param msg 错误消息
28      * @param bundle 错误描述详细信息
29      */
30     public void onError(int error, String msg, Bundle bundle) {
31
32     }
33
34     /**
35      * 状态变化
```

```
36      * @param state 状态
37      * 智能体流程开始 {@link Constants.CallState#ON_CALL_BEGIN }
38      * 智能体流程结束 {@link Constants.CallState#ON_CALL_END }
39      */
40      public void onCallStateChange(int state) {
41
42      }
43
44      /**
45      * 链接状态变化
46      * @param state 状态 {@link Constants.ConnectionState }
47      * */
48      public void onConnectionStateChange(int state) {
49
50      }
51
52
53
54      /**
55      * 用户端音频ASR结果
```

```
56      * @param text 文本
57      * @param isFinal 标记是否最终结果
58      * */
59      public void onUserAsrSubtitle(String text, boolean isFinal) {
60
61      }
62
63      /**
64       * AI智能体结果
65       * @param text 文本
66       * @param isFinal 标记是否最终结果
67       * */
68      public void onAIAgentSubtitle(String text, boolean isFinal) {
69
70      }
71
72      /**
73       * AI智能体音频状态变化
74       * @param newState 状态变化
75       * */
```

```
76     public void onAIAgentAudioStateChange(@Constants.AIAgentAudioStateType int
      newState) {
77
78     }
79
80     /**
81     * 用户声音状态变化
82     * @param newState 状态变化
83     * */
84     public void onUserAudioStateChange(@Constants.UserAudioStateType int newState) {
85
86     }
87
88     /**
89     * 智能体音频能量，间隔100ms 回调一次
90     * @param level 音频能量
91     * */
92     public void onAIAgentAudioLevel(int level) {
93
94     }
```



```
95
96  /**
97   * 用户音频能量，间隔100ms 回调一次
98   * @param level 音频能量
99   * */
100 public void onUserAudioLevel(int level) {
101
102 }
103
104 /**
105  * functionCall
106  * @param id functionCall 单次唯一标识
107  * @param params json参数, 例如: {"function_name":"phone_call","parameter_list":
108 [{"called_number":"1891017000"}, {"called_name":"我的父亲"}
109  * */
110 public void onFunctionCall(String id, String params) {
111
112 }
113 /**
```

```
114      * 文件长传状态
115      * @param code 状态 成功 {@link Constants#NO_ERROR }, 其他失败
116      * @param msg 消息, 成功时候返回文件名, 失败返回错误消息
117      * */
118      public void onUploadFileStatus(int code, String msg) {
119
120      }
121
122      /**
123      * 音频输出接口
124      * 注意: 预期这里直接将音频输入给AudioTrack, 如果不输入到AudioTrack, 这里需要
      TimeUnit.MILLISECONDS.sleep(10);
125      * 数据直接写入AudioTrack;
126      * {@link AIEngineParams#enableExternalAudioOutput} 打开音频外部输出时候有该回调
127      * @param data 音频数据
128      * @param sampleRate 采样率
129      * @param channelCount 声道数
130      * */
```

```
131     public void onPlaybackAudioFrame(byte[] data, int sampleRate, int channelCount)
132     {
133     }
134
135     /**
136     * agent意图
137     * @param type 意图类型, 暂时支持,
138     *     图片上传 {@link Constants.AgentIntentType#IMAGE_UPLOAD }
139     *     播放音频 {@link Constants.AgentIntentType#AGENT_EVENT_PLAY_AUDIO }
140     * @param bundle 其他附加信息
141     * */
142     public void onAgentIntent(String type, Bundle bundle) {
143
144     }
145
146 }
```