



3、aiuisdk使用介绍



aiuisdk使用介绍

0: AIUI语音相关的代码在com.demo.aiuisdk目录下

1: 初始化语音相关的模块，包括持续语音识别、语音合成、语义理解功能

```
SpeechManager.getInstance();
```

2: 语音识别文字和语义结果的回调接口

```
SpeechManager.getInstance().setSpeechCallback(speechCallback);

BaseSpeechCallback speechCallback = new BaseSpeechCallback() {
    /**
     * 通过识别语音得到文字
     *
     * @param text 识别后的文本
     */
    @Override
    public void recognizeResult(String text) {
        // 语音识别结果
    }

    /**
     * @param text 返回三方语义结果，可直接speak出来
     * @param json 返回三方语义结果json串，可根据需求自行解析
     */
    @Override
    public void nlpResult(String text, String json) {
        // 语义理解结果
        SpeechManager.onSpeaking(text);
    }
};
```



3: 设置错误回调接口

```
SpeechManager.getInstance().setErrorInfoCallback(iErrorInfoCallback);  
/**  
 * @param code: 错误码  
 * @param info: 错误信息  
 */  
private IErrorInfoCallback iErrorInfoCallback = (code, info) -> {  
    // 处理了一部分错误同时将相关的错误信息抛出来，方便上层自己处理  
};
```

4: 合成发音人设置

```
SpeechManager.getInstance().setTtsLanguage(ITTSLanguage.jiajia);
```

5: 设置当前项目使用的哪种情景模式,如果当前项目不同于其他项目配置,需要自己先在讯飞平台上自定义一个情景模式,设置自己项目特有的参数,然后在枚举SCENE中增加自己的情景模式,通过调用setScene方法来声明自己使用哪种情景模式

```
SpeechManager.setScene(SpeechManager.SCENE.mainNoTuling);
```

6: 合成TTS接口

合成加回调

```
public static void onSpeaking(String text, Runnable onComplete)
```

兼容旧版本, 后期弃用

```
public static void onSpeaking(String text, MySynthesizerListener onComplete)
```

只合成, 无回调

```
public static void onSpeaking(String text)
```

兼容旧版本, 后期弃用

```
public static void onSpeaking(String text, boolean b)
```

7: 获取缓存的语义结果

```
SpeechManager.getNlpResultCache();
```

8: 清空语义缓存结果

```
SpeechManager.setNlpResultCacheEmpty();
```

9: 判断是否正在说话中

```
public static boolean isSpeaking()
```



10: 旧项目里有以下相关类时需要修改下包名，尤其是SpeechManager、BaseSpeechCallback、MySynthesizerListener这三个类需要特别注意

```
import com.demo.aiuisdk.BaseSpeechCallback;
import com.demo.aiuisdk.IErrorInfoCallback;
import com.demo.aiuisdk.MySynthesizerListener;
import com.demo.aiuisdk.SpeechManager;
import com.demo.aiuisdk.config.ITTSLanguage
```



11: 打断方案

- 10.1 加入唤醒词：这种方式不太推荐，安静 可能会增加误唤醒概率；
- 10.2 使用自定义技能：这个处理方式跟自定义问答是一样的
- 10.3 目前采用自定义问答在线的
- 10.4 离线命令词识别

12: 离线语音合成TTS不用讯飞语记

13: 需要等2小时测试一下，但是热词并不是绝对的，只是增加热词的被识别出来的概率。

14: 清除历史

AIUI支持多轮对话，如在问合肥今天的天气怎么样之后，再询问明天的呢，AIUI会结合上一句询问合肥今天 天气的历史，就会回答合肥明天的天气。

AIUI默认在休眠后唤醒会清除交互历史，在交互状态下唤醒，则不会清除交互历史。

AIUI清除历史的方式是可配置的，默认为auto即是上面描述的模式。当配置成user值后，用户可以通过发送CMD_CLEAN_DIALOG_HISTORY在任何时候手动清除交互的历史。

即使上面两种情况下，客户端没有主动清除交互历史，服务端保存用户交互历史的时间也是有限的，当用户交互超过5轮后，服务端也会将交互历史清空。

15: AIUI的SDK出现网络10120问题，实际是anzer5G网络来回切换导致网络不稳定，换了高功率的HRR-Test（anzer2017）后正常。

16: 上传热词无法通过客户端SDK进行，单纯的热词仅可通过网页端进行上传，如果您逐步深入了解了所见即可说、动态实体等高级特性就会发现上传的实体等资源本身就是热词的一种。

17: AIUI主动在线合成有时会出现卡顿，建议使用离线合成

18: AIUI不支持识别语种的切换，只有通过情景模式来切换自己想要的语种

