

智能语音

SDK 文档

产品文档



腾讯云

【版权声明】

©2013-2019 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或模式的承诺或保证。

文档目录

SDK 文档

流式语音合成 SDK

Android SDK

iOS SDK

C++ SDK

Java SDK

PHP SDK

Python SDK

SDK 文档

流式语音合成 SDK

Android SDK

最近更新时间：2019-08-21 11:30:20

开发相关

开发准备

- 支持 Android 4.0 以上版本 API LEVEL 16，支持手机与平板。
- 合成实时流式语音，需要手机能够连接网络（2/3/4G 或 Wi-Fi 网络等）。
- 建议使用最新版本 Android Studio 进行开发。
- 腾讯云控制台获取 AppID、SecretID、SecretKey，详情参考 [基本概念](#)。
- 服务端 [API 文档](#)。

下载安装 SDK

- 语音合成 Android SDK [下载地址](#)。
- 解压得到 tts-sdk-android 文件夹，即是示例代码工程，工程目录 app/libs 下的 aar 格式 SDK 包。
- 用 Android Studio 打开此工程查看语音合成示例代码。
- 流式接口：实例代码参考 RealtimeTtsActivity 类，语音实时返回，支持不超过300中文字符或900英文字符，不支持暂停与恢复。
- 长文本接口：实例代码参考 LongTextTtsActivity 类，支持长文本，支持播放暂停与恢复。

参数说明

参数名称	类型	必填	说明
appId	int	是	腾讯云 ID，即 AppID， 获取地址
secretId	String	是	腾讯云安全凭证， 获取地址
secretKey	String	是	腾讯云安全凭证，获取地址同上
sessionId	String	否	一次请求对应一个 SessionId，会原样返回
projectId	String	否	项目 ID，用户自定义，默认为 0， 获取地址

参数名称	类型	必填	说明
speed	int	否	语速，范围：[-2, 2]，分别对应不同语速：0.6倍、0.8倍、1.0倍、1.2倍、1.5倍，默认为0
voiceType	int	否	tts音色，默认女声，亲和风格
language	int	否	主语言类型，默认中文

快速入门

初始化 TtsController 示例

构造 ttsController：长文本使用 LongTextTtsController；实时流式使用 RealtimeTtsController，2选1。

```
TtsController mTtsController = new TtsController();
```

在使用云 API 之前，请前往腾讯云 API 密钥页面申请安全凭证。安全凭证包括 SecretId 和 SecretKey。

SecretId 用于标识 API 调用者身份。

SecretKey 用于加密签名字符串和服务器端验证签名字符串的密钥。

注意：

这里只是示例，请根据用户实际申请的 SecretId 和 SecretKey 进行后续操作。

```
mTtsController.init(  
    1257709062L, //腾讯云 appId  
    "AKIDzllbgVXMPC**QaT6TZOWDF1WktQr4", //腾讯云 secretId  
    "6xYsxngLo45sT**ORFuMZZLs9BzXt" //腾讯云 secretKey  
);
```

设置自定义参数

```
public enum VoiceType {  
    VOICE_TYPE_AFFNITY_FEMALE(0, "亲和女声(默认)"),  
    VOICE_TYPE_AFFNITY_MALE(1, "亲和男声"),  
    VOICE_TYPE_MATURE_MALE(2, "成熟男声"),  
    VOICE_TYPE_VIBRANT_MALE(3, "活力男声"),  
    VOICE_TYPE_WARM_FEMALE(4, "温暖女声"),  
}
```

```
VOICE_TYPE_Emotional_FEMALE(5, "情感女声"),  
VOICE_TYPE_Emotional_MALE(6, "情感男声"),  
}  
  
public enum VoiceSpeed {  
    VOICE_SPEED_VERY_SLOW(-2, 0.6f, "0.6倍"),  
    VOICE_SPEED_SLOWDOWN(-1, 0.8f, "0.8倍"),  
    VOICE_SPEED_NORMAL(0, 1.0f, "正常语速(默认)"),  
    VOICE_SPEED_ACCELERATE(1, 1.2f, "1.2倍"),  
    VOICE_SPEED_VERY_FAST(2, 1.5f, "1.5倍"),  
}  
  
public enum VoiceLanguage {  
    VOICE_LANGUAGE_CHINESE(1, "中文 (默认)"),  
    VOICE_LANGUAGE_ENGLISH(2, "英文"),  
}
```

示例

```
//设置语速  
mTtsController.setVoiceSpeed(speed);  
  
//设置音色  
mTtsController.setVoiceType(voice);  
  
//设置音量  
mTtsController.setVoiceVolume(volume);  
  
//设置语言  
mTtsController.setVoiceLanguage(language);  
  
//设置ProjectId  
mTtsController.setProjectId(0);
```

语音合成

```
mTtsController.startTts(ttsText, mTtsExceptionHandler, new QCloudTTSPayer.QCloudTTSPayerCallba  
ck() {  
  
    //播放开始  
    @Override  
    public void onTTSPplayStart() {  
        Log.d("tts", "onPlayStart");  
    }  
}
```

```
//音频缓冲中
@Override
public void onTTSPPlayWait() {
    Log.d("tts", "onPlayWait");
}

//缓冲完成，继续播放
@Override
public void onTTSPPlayResume() {
    Log.d("tts", "onPlayResume");
}

//连续播放下一句
@Override
public void onTTSPPlayNext() {
    Log.d("tts", "onPlayNext");
}

//播放中止
@Override
public void onTTSPPlayStop() {
    Log.d("tts", "onPlayStop");
}

//播放结束
@Override
public void onTTSPPlayEnd() {
    Log.d("tts", "onPlayEnd");
}
});
```

接收异常

```
//接收接口异常
private final TtsController.TtsExceptionHandler mTtsExceptionHandler = new TtsController.TtsExceptionHandler() {
    @Override
    public void onRequestException(TtsController.TtsException e) {
        Log.e(TAG, "tts onRequestException");
        Toast.makeText(TtsActivity.this, e.getErrMsg(), Toast.LENGTH_SHORT).show();
    }
};
```

暂停、恢复或停止语音播放

```
mTtsController.pause();
```

```
mTtsController.resume();
```

```
mTtsController.stop();
```

错误码

请参考 [语音合成 API 文档](#)。

iOS SDK

最近更新时间：2019-08-21 11:30:28

开发相关

开发准备

- 支持 iOS 8.0 及以上版本，不支持 bitcode 版本。
- 合成实时流式语音，需要手机能够连接网络（2/3/4G 或 Wi-Fi 网络等）。
- 语音合成 iOS SDK [下载地址](#)。
- 腾讯云控制台获取 AppID、SecretID、SecretKey，详情参考 [基本概念](#)。
- 服务端 [API 文档](#)。

导入 SDK

iOS SDK 文件夹名称为：QCloud_TTS。其中包含一个 .a 静态库和一个头文件夹 header。

配置工程

- 在 Build Settings 中设置 Other Linker Flags，加入参数 -ObjC。
- 在工程中添加依赖库，在 build Phases Link Binary With Libraries 中添加以下库：libc++.tbd

初始化

引入上传 SDK 中的头文件 QCloudTTS.h，需要先实例化 QCloudTTS 对象。

参数说明

参数名称	类型	必填	说明
appId	NSInteger	是	腾讯云 ID，即 AppID， 获取地址
secretId	NSString	是	腾讯云安全凭证， 获取地址
secretKey	NSString	是	腾讯云安全凭证，获取地址同上
sessionId	NSString	否	一次请求对应一个 SessionId，会原样返回
projectId	NSString	否	项目 ID，用户自定义，默认为 0， 获取地址
speed	NSInteger	否	语速，范围：[-2, 2]，分别对应不同语速：0.6倍、0.8倍、1.0倍、1.2倍、1.5倍，默认为 0
voiceType	NSInteger	否	tts音色，默认女声，亲和风格

参数名称	类型	必填	说明
language	NSInteger	否	主语言类型，默认中文

快速入门

初始化 QCloudTTS 并注册 QCloudTTSTDelegate

```
QCloudTTS * apiObj = [[QCloudTTS alloc] initWithAppId:appId secretId:secretId secretKey:secretKey];  
apiObj.ttsDelegate = self;
```

设置自定义参数 QCloudTTSConfig

TTS音色类型

```
typedef NS_ENUM(NSUInteger, VoiceType) {  
    VoiceTypeAffinityFemale = 0, // 0 : 亲和女声 (默认)  
    VoiceTypeAffinityMale = 1, // 1 : 亲和男声  
    VoiceTypeMatureMale = 2, // 2 : 成熟男声  
    VoiceTypeVibrantMale = 3, // 3 : 活力男声  
    VoiceTypeWarmFemale = 4, // 4 : 温暖女声  
    VoiceTypeEmotionalFemale = 5, // 5 : 情感女声  
    VoiceTypeEmotionalMale = 6, // 6 : 情感男声  
};
```

TTS语速

```
typedef NS_ENUM(NSUInteger, SpeedType) {  
    SpeedTypeVerySlow = -2, // 0.6 倍  
    SpeedTypeSlowDown = -1, // 0.8 倍  
    SpeedTypeNormal = 0, // 1.0 倍 (默认)  
    SpeedTypeAccelerate = 1, // 1.2 倍  
    SpeedTypeVeryFast = 2, // 1.5 倍  
};
```

TTS主语言类型

```
typedef NS_ENUM(NSUInteger, PrimaryLanguage) {  
    PrimaryChinese = 1, // 1 : 中文 (默认)  
    PrimaryEnglish = 2, // 2 : 英文  
};
```

示例

```
QCloudTTSTConfig *config = [QCloudTTSTConfig getInstance];
config.projectId = 1234567;
config.volume = VolumeLevelOne;
config.speed = SpeedTypeNormal;
config.voiceType = VoiceTypeFemale;
config.language = PrimaryChinese;
```

监听 QCloudTTSTDelegate 播放状态**

```
//开始播放
- (void) onTTSPplayStart{

    NSLog(@"onTTSPplayStart");

}

//音频缓冲中

- (void) onTTSPplayWait{

    NSLog(@"onTTSPplayWait");

}

//缓冲完成，继续播放

- (void) onTTSPplayContinue{

    NSLog(@"onTTSPplayContinue");

}

//播放中止

- (void) onTTSPplayStop{

    NSLog(@"onTTSPplayStop");

}

//播放结束
```

```
- (void) onTTSPplayEnd{  
  
    NSLog(@"onTTSPplayEnd");  
  
}
```

语音合成

```
(BOOL)startTTS:(NSString *)text fail:(TTSEExceptionHandler)fail;
```

示例

在使用云API之前，请前往 [腾讯云API密钥页面](#) 申请安全凭证。安全凭证包括 SecretId 和 SecretKey。

SecretId 用于标识 API 调用者身份。

SecretKey 用于加密签名字符串和服务器端验证签名字符串的密钥。

注意：

这里只是示例，请根据用户实际申请的 SecretId 和 SecretKey 进行后续操作。

```
NSInteger appId = 1257709062; //腾讯云 appId  
NSString *secretId = @ "AKIDzllbgVXMPC*****QaT6TZOWDF1WktQr4" ; //腾讯云 secretId  
NSString *secretKey = @ "6xYsxngLo45sT*****ORFuMZZLs9BzXt" ; //腾讯云 secretKey  
  
QCloudTTS* _apiObj = [[QCloudTTS alloc] initWithAppId:appId secretId:secretId secretKey:secretKey];  
  
apiObj.ttsDelegate = self;  
[apiObj startTTS:ttsText fail:^(NSString *code, NSString *errMsg) {  
    //接口异常处理  
    NSLog(@"error:%@" ,errMsg);  
}];
```

暂停、恢复或停止语音播放

```
[apiObj pauseTTS];  
  
[apiObj resumeTTS];  
  
[apiObj stopTTS];
```

错误码

请参考 [语音合成 API 文档](#)。

C++ SDK

最近更新时间：2019-06-18 19:53:38

流式语音合成C++ SDK [下载地址](#)。

此 SDK 目前仅支持在 Linux 平台上使用。

接口请求域名：aai.cloud.tencent.com/tts

腾讯云语音合成技术（TTS）可以将任意文本转化为语音，实现让机器和应用张口说话。腾讯 TTS 技术可以应用到很多场景，例如，移动 App 语音播报新闻；智能设备语音提醒；支持车载导航语音合成的个性化语音播报。本接口内测期间免费使用。

开发环境

基础编译环境

安装gcc g++

```
yum install -y gcc gcc-c++ make automake
//安装 gcc 等必备程序包（已安装则略过此步）
yum install -y wget
```

安装 CMake 工具

```
// cmake 版本要大于3.5
wget https://cmake.org/files/v3.5/cmake-3.5.2.tar.gz
tar -zxvf cmake-3.5.2.tar.gz
cd cmake-3.5.2
./bootstrap --prefix=/usr
gmake
gmake install
```

依赖库安装及编译

curl

客户需自行安装版本大于7.44.0：

下载 [curl 文件](#) 解压进入源码目录。

```
sudo ./configure
sudo cmake ./
sudo make
sudo make install
```

openssl

本 SDK 提供，目录为：c++_tts_sdk/lib

如果不适合客户系统，自行安装方法，版本1.0.2f:

下载 [wget 源码](#)。

1.更新zlib

RedHat系列:yum **install** -y zlib

Debian系列:sudo apt-**get install** zlib1g zlib1g.dev

2.安装

tar xzf openssl-1.0.2f.tar.gz

cd openssl-1.0.2f

sudo ./config **shared** zlib

sudo make

sudo make **install**

自行替换 c++_tts_sdk/extern/openssl_1.0.2f/lib下面的库文件

jsoncpp

本SDK提供，目录为：extern/json/lib

存在64位版本和32位版本，客户可自行根据自身操作系统更换链接

opus

本sdk提供，目录为:

extern/opus

如果不适合客户系统，自行安装方法:

下载 [opus 源码](#) 解压进入源码目录。

```
sudo ./configure --prefix=/usr
```

```
sudo make
```

```
sudo make install
```

自行替换 c++_tts_sdk/extern/opus/lib下面的库文件

获取用户信息

获取 AppID , SecretId 与 SecretKey

- 关于云 AP I 账号中的 AppID , SecretId 与 SecretKey 查询方法，可参考 [签名签权](#)。
- 具体路径为：单击 [腾讯云控制台](#) 右上角您的账号，选择【访问管理】>【访问密钥】>【API 密钥管理】界面查看 AppID 和 key。

更改用户信息配置文件

将查询到的用户信息更改到 conf/tcloud_auth.ini 配置文件中。

用户需要替换成自己账号信息

AppId=1259***** ****

SecretId=AKIDo*****

SecretKey=kFpw*****

开发相关

请求参数

参数名称	必选	类型	描述
Action	是	String	本接口取值：TextToStreamAudio，不可更改
AppId	是	Int	用户在腾讯云注册账号的AppId，具体可以参考 获取用户信息 。
SecretId	是	String	用户在腾讯云注册账号AppId对应的SecretId，获取方法同上。
Text	是	String	合成语音的源文本，最大支持800字符。
SessionId	是	String	一次请求对应一个 SessionId，会原样返回，建议传入类似于 uuid 的字符串防止重复。
ModelType	否	Int	模型类型，1：默认模型，此字段只需设置为1即可。
Volume	否	Float	音量大小，范围：[0, 10]，分别对应11个等级的音量，默认值为0，代表正常音量。没有静音选项。 输入除以上整数之外的其他参数不生效，按默认值处理。
Speed	否	Int	语速，范围：[-2, 2]分别对应不同语速： -2代表0.6倍 -1代表0.8倍 0代表1.0倍（默认） 1代表1.2倍 2代表1.5倍 输入除以上整数之外的其他参数不生效，按默认值处理。

参数名称	必选	类型	描述
VoiceType	否	Int	音色选择： 0：亲和女声（默认） 1：亲和男声 2：成熟男声 3：活力男声 4：温暖女声 5：情感女声 6：情感男声
PrimaryLanguage	否	Int	主语言类型： 1：中文（默认） 2：英文
SampleRate	否	Int	音频采样率： 16000：16k（默认） 8000：8k
Codec	否	String	返回音频格式： opus：返回多段含 opus 压缩分片音频，数据量小，建议使用（默认）。 pcm：返回二进制 pcm 音频，使用简单，但数据量大。
ProjectId	否	Int	项目ID，可以根据控制台-账号中心-项目管理中的配置填写，如无配置请填写默认项目ID:0。
Timestamp	是	Int	当前 UNIX 时间戳，可记录发起 API 请求的时间。如果与当前时间相差过大，会引起签名过期错误。SDK会自动赋值当前时间戳。
Expired	是	Int	签名的有效期，是一个符合 UNIX Epoch 时间戳规范的数值，单位为秒；Expired 必须大于 Timestamp 且 Expired-Timestamp 小于 90天。SDK默认设置1小时。

具体参数配置可以参考：[conf/request_parameter.ini](#) 的默认配置

请求接口

TCloudTTS::InitCommonParam

```

/* 初始化公共请求参数，此类参数较稳定不变
** configPath 参数文件路径
** time 鉴权的有效时间
** Output int 返回结果
*/
int InitCommonParam(string configPath);

```

此接口用来初始化请求的公共请求参数，参数参考配置文件说明。

路径为：conf/request_parameter.ini

TCloudTTS::InitAuth

通过配置文件初始化，建议选择此接口

/ 初始化用户信息*

*** configPath 参数文件路径*

*** time 鉴权的有效时间*

*** Output int 返回结果*

**/*

int InitAuth(string configPath, int time = 60 * 60);

通过结构体初始化

/ 初始化用户信息*

*** stAuth 用户信息结构体*

*** time 鉴权的有效时间，默认为 1h*

*** Output int 返回结果*

**/*

int InitAuth(TCloudTTSAuth stAuth, int time = 60 * 60);

此接口用来初始化用户信息，参数参考配置文件说明。信息获取参考章节 二。

路径为: conf/tcloud_auth.ini

TCloudTTS::CreateRequest

初始化公共参数后不需要更改则只需要通过text来创建请求，建议使用此接口：

/ 创建请求*

*** text 请求文本，不改变其他请求参数*

*** Output int 返回结果*

**/*

int CreateRequest(string text);

通过完整的请求参数信息来创建请求：

/ 创建请求*

*** stReq 完整的请求结构体信息*

*** Output int 返回结果*

**/*

int CreateRequest(TCloudTTSReq stReq);

此接口用来创建一个完整的请求信息。

TCloudTTS::RequestToJson

```
/* 请求转换成json
** stReq 请求结构体
** Output string json结果
*/
string RequestToJson(TCloudTTSReq stReq);
```

此接口将请求转换成json body,在post时作body传到服务端。

TCloudTTS::Process

```
/* 执行请求
** strRsp 请求的json body
** Output int 返回结果
*/
int Process(string &strRsp);
```

此接口将执行一次请求。

SplitString (可选功能)

```
/* 将文本根据标点符号切割
** strText 需要切割的原始文本
** model为标点符合的集合 例如 : ".,!?"
** Out put 为切割完成的文本片
*/
vector<string> SplitString(string strText, string model);
```

此接口提供对文本进行标点切割功能。用户可选择使用，建议参考demo的思路。

请求Demo

前提已经更改用户正确的AppId信息

```
cmake ./
make
./tts_test
```

简单开发流程介绍

以下存在多种方法的建议选择方法一，较优，其他方法供接入用户多选

创建请求 TCloudTTS ttsReq:

初始化用户信息

方法一（推荐此方法）：

调用InitAuth 通过配置文件初始化，默认路径 conf/tcloud_auth.ini

方法二:

调用 InitAuth 通过 TCloudTTSAuth 结构体初始化

参考 [TCloudTTS::InitAuth](#) 接口。

初始化请求参数

公共请求参数大多只需要设置一次，除了Text，SessionId参数

方法一（推荐此方法）:

调用InitCommonParam通过配置文件 conf/request_parameter.ini

调用CreateRequest传入Text。

方法二:

通过完整的请求结构体来初始化请求

调用InitCommonParam通过TCloudTTSTReq请求体来初始化。

参考 [TCloudTTS::InitCommonParam](#) 接口。

分片（可选功能）

初次接入不建议选择。

本sdk提供根据标点符号将整个文本进行分片的方法：

通过调用SplitString（Text）获取整个文本分片的结果，每片单独调用获取音频再合并。

[SplitString](#) 接口。

执行任务

调用**Process**函数获取音频结果，结果为**pcm**格式的音频串。
结果存储在 **TCloudTTS::m_strRsp**;

SDK 已提供各个接口源码，用户可根据自身需要进行更改。

C++ 快速入门示例

参考 Demo/demo.cpp

```
int main(){
    TCloudTTS tts;
    //初始化请求参数
    tts.InitCommonParam("./conf/request_parameter.ini");
    //初始化用户信息
    tts.InitAuth("./conf/tcloud_auth.ini");
    //文本过长可以自行切片，将返回strRsp合并即为完整音频
    //Text限制800字符以内
    string strRsp;
```

```
int ret = 0;
string strText = "尊敬的旅客您好，很抱歉的通知您：您在去哪网购买的2018年7月11日首尔到大连OZ301航班发生变动，最新的飞行计划为2018年7月11日9时35分的OZ301航班，从仁川国际机场起飞，2018年7月11日9时50分到达大连周水子国际机场，起降时间均为当地时间，我们已将详情发送至您的手机和邮箱，如需协助请拨打去哪网热线电话95117，谢谢。";

//创建具体文本请求
tts.CreateRequest(strText);
ret = tts.Process(strRsp);

//将pcm音频保存为wave格式
FILE *f = fopen("./test.wav", "wb");
unsigned char *wav_buffer = (unsigned char *) malloc(4 * 1024 * 1024);
//添加wave头
pcm16le_to_wave_buf((const int16_t *)strRsp.c_str(), strRsp.size(), 1, 16000, wav_buffer);
fwrite(wav_buffer, sizeof(char), strRsp.size() + 44, f);
fclose(f);
free(wav_buffer);

return 0
}
```

Java SDK

最近更新时间：2019-06-18 19:53:44

流式语音合成 Java SDK [下载地址](#)。

接口请求域名：aai.cloud.tencent.com/tts。

腾讯云语音合成技术（TTS）可以将任意文本转化为语音，实现让机器和应用张口说话。腾讯 TTS 技术可以应用到很多场景，例如，移动 App 语音播报新闻；智能设备语音提醒；支持车载导航语音合成的个性化语音播报。本接口内测期间免费使用。

开发环境

基础编译环境

jdk1.8及以上

本文件夹包含了 Jar 包和源码。源码可通过 Eclipse 直接打开，或将 src 拷至 IDEA 等软件中。

JAR 包使用步骤举例如下：

1. 找到：out 和 lib 文件夹中的 jar 文件，一共16个(包括5个source jar)。
2. 将 jar 复制到您的工程文件夹中。如果第三方 Jar 和您已使用的 Jar 有重复，可选择其一。
3. 右键单击 Eclipse，选择【您的项目】>【Properties】>【Java Build Path】>【Add JARs】。
4. 将 jar 文件 include 到您的项目中。包括：out 文件夹中的：tts-sdk-1.0.jar 和第三方的10个依赖工具包。

添加完成后，用户就可以在工程中使用TTS语音合成SDK了。

获取用户信息

获取 AppID，SecretId 与 SecretKey

- 关于云 API 账号中的 AppID，SecretId 与 SecretKey 查询方法，可参考 [签名鉴权](#)。
- 具体路径为：单击 [腾讯云控制台](#) 右上角您的账号，选择【访问管理】>【访问密钥】>【API 密钥管理】界面查看 AppID 和 key。

更改用户信息配置文件

将查询到的用户信息更改到 SDK 中。

```
/**  
 * 用户的语音请求基本参数，仅需配置一次，全局使用。通常是注册腾讯云账号后，登录控制台，从“个人 API
```

密钥”中获得。

* 获取方法请查看：签名鉴权 获取签名所需信息

* 请用户务必将自己的各项参数值赋值到本类对应变量中。

*/

```
public class AsrBaseConfig {  
    public static String secretId = "AKID31NbfXbpBLJ4kGJrytc9UfgVAIGltJJ8";  
    public static String secretKey = "kKm26uXCgLtGRWVJvKtGU0LYdWCgOvGP";  
    public static String appId = "1255628450";  
}
```

开发相关

请求参数

参数名称	必选	类型	描述
Action	是	String	本接口取值：TextToStreamAudio，不可更改。
AppId	是	Int	用户在腾讯云注册账号的 AppId，具体可以参考 获取用户信息 。
SecretId	是	String	用户在腾讯云注册账号 AppId 对应的 SecretId，获取方法同上。
Timestamp	是	Int	当前 UNIX 时间戳，可记录发起 API 请求的时间。如果与当前时间相差过大，会引起签名过期错误。SDK会自动赋值当前时间戳。
Expired	是	Int	签名的有效期，是一个符合 UNIX Epoch 时间戳规范的数值，单位为秒；Expired 必须大于 Timestamp 且 Expired-Timestamp 小于 90天。SDK 默认设置 1 h。
Text	是	String	合成语音的源文本，最大支持800字符。
SessionId	是	String	一次请求对应一个 SessionId，会原样返回，建议传入类似于 uuid 的字符串防止重复。
ModelType	否	Int	模型类型，1：默认模型，此字段只需设置为1即可。
Volume	否	Float	音量大小，范围：[0，10]，分别对应11个等级的音量，默认值为 0，代表正常音量。没有静音选项。 输入除以上整数之外的其他参数不生效，按默认值处理。

参数名称	必选	类型	描述
Speed	否	Int	语速，范围：[-2，2]分别对应不同语速： -2代表0.6倍 -1代表0.8倍 0代表1.0倍（默认） 1代表1.2倍 2代表1.5倍 输入除以上整数之外的其他参数不生效，按默认值处理。
ProjectId	否	Int	项目 ID，可以根据控制台-账号中心-项目管理中的配置填写，如无配置请填写默认项目ID:0。
VoiceType	否	Int	音色选择： 0：亲和女声（默认） 1：亲和男声 2：成熟男声 3：活力男声 4：温暖女声 5：情感女声 6：情感男声
PrimaryLanguage	否	Int	主语言类型： 1：中文（默认） 2：英文
SampleRate	否	Int	音频采样率： 16000:16k（默认） 8000:8k
Codec	否	String	返回音频格式： opus：返回多段含 opus 压缩分片音频，数据量小，建议使用（默认）。 pcm：返回二进制 pcm 音频，使用简单，但数据量大。

请求接口

initBaseParameters

```
/*  
** 初始化基础参数, 请将下面的参数值配置成你自己的值。配置可中途修改，正常情况下立即生效。  
** 参数获取方法可参考：<a href="https://cloud.tencent.com/document/product/441/6203">签名鉴权  
获取签名所需信息</a>  
*/  
private static void initBaseParameters() {  
    // Required
```

```
// AsrBaseConfig.applId = "YOUR_APP_ID_SET_HERE";
// AsrBaseConfig.secretId = "YOUR_SECRET_ID";
// AsrBaseConfig.secretKey = "YOUR_SECRET_KEY";

// optional, 根据自身需求设置配置值, 不配则使用默认值。
TtsConfig.VOLUME = 5; // 音量大小, 范围[0, 10], 默认为0, 表示正常音量。
// TtsConfig.REQUEST_ENCODE = RequestEncode.UTF_8; // 传入的文字所采用的编码, 默认为utf-8
// TtsConfig.SPEED = 0; // 语速, 范围[-2, 2]. -2: 0.6倍; -1: 0.8倍; 0:1.0倍 (默认); 1: 1.2倍; 2: 1.5倍。其他
// 值: 1.0 倍。
// TtsConfig.VOICE_TYPE = 0; // 音色: 0: 亲和女声 (默认) 1: 亲和男声 2: 成熟男声 3: 活力男声 4: 温
// 暖女声 5: 情感女声 6: 情感男声
// TtsConfig.SAMPLE_RATE = 16000; // 音频采样率: 16000: 16k (默认); 8000: 8k
// TtsConfig.PRIMARY_LANGUAGE = 1; // 主语言类型: 1-中文 (默认) 2-英文
// TtsConfig.CODEC = CodeC.PCM; // 无需修改。暂未支持Opus方式。
}
```

sendStringRequest

```
/*
** 从字节数组读取语音数据, 发送请求。
*/
private void sendStringRequest() {
// 方法1:
TtsSynSender ttsSynSender = new TtsSynSender(); // 创建之后可重复使用
String text = "早上好, 今天天气真不错.";
TtsResponse response = ttsSynSender.request(text, "session-id-123");
// TtsResponse response2 = ttsSynSender.sendRequest(text);
printAndSaveReponse(response);

// 方法2:
String filePath = "testTtsFiles/test_article.txt";
List<TtsResponse> list = ttsSynSender.requestFromFile(filePath, StandardCharsets.UTF_8, "Session-Id-xx");
for (TtsResponse ttsResponse : list) {
    printAndSaveReponse(ttsResponse);
}
}
```

简单开发流程介绍

先初始化一次基本参数:

```
AsrBaseConfig.applId = "YOUR_APP_ID_SET_HERE";
AsrBaseConfig.secretId = "YOUR_SECRET_ID";
AsrBaseConfig.secretKey = "YOUR_SECRET_KEY";
TtsConfig.VOLUME = 5; // 音量大小
```

.....

然后开始调用：

方法一. 同步调用：

```
TtsSynSender ttsSynSender = new TtsSynSender();
TtsResponse response = ttsSynSender.request(text, "session-id-123");
byte[] pcmBytes = response.getResponseBytes();
String sessionId = response.getSessionId(); // 即： "session-id-123"
或者：
List<TtsResponse> list = ttsSynSender.requestFromFile(filePath, "session-id-xxx");
详见实例类：
/TtsSdkJava/src/com/tencent/cloud/asr/tts/sdk/TtsRequestSample.java
```

方法二. 异步多线程调用：

线程A不断add文本，线程B收取合成的语音结果字节数组。详见实例类：
/TtsSdkJava/src/com/tencent/cloud/asr/tts/sdk/TtsAsynRequestSample.java

Java 快速入门示例

```
public class TtsRequestSample {

    static {
        initBaseParameters();
    }

    public static void main(String[] args) {
        TtsRequestSample ttsRequestSample = new TtsRequestSample();
        ttsRequestSample.start();
    }

    private void start() {
        this.sendStringRequest();
        System.exit(0);
    }

    /**
     * 从字节数组读取语音数据，发送请求。
     */
    private void sendStringRequest() {
        // 方法1：
        TtsSynSender ttsSynSender = new TtsSynSender(); // 创建之后可重复使用
```

```
String text = "早上好，今天天气真不错。";
TtsResponse response = ttsSynSender.request(text, "session-id-123");
// TtsResponse response2 = ttsSynSender.sendRequest(text);
printAndSaveReponse(response);

// 方法2：
String filePath = "testTtsFiles/test_article.txt";
List<TtsResponse> list = ttsSynSender.requestFromFile(filePath, StandardCharsets.UTF_8, "Session-Id-xx");
for (TtsResponse ttsResponse : list) {
    printAndSaveReponse(ttsResponse);
}

private void printAndSaveReponse(TtsResponse response) {
    if (response != null) {
        new File("logs").mkdirs();
        File pcmFile = new File("logs/" + response.getSessionId() + ".pcm");
        this.savePcmFile(response.getResponseBytes(), pcmFile);
        File wavFile = new File("logs/" + response.getSessionId() + "_Convert.wav");
        this.saveToWavFile(response.getResponseBytes(), pcmFile, wavFile);
        System.out.println("Response: " + response.getSessionId() + ", length: "
            + response.getResponseBytes().length + ", result saved at: " + pcmFile.getAbsolutePath());
    } else
        System.out.println("Result is null.");
}

/**
 * 初始化基础参数, 请将下面的参数值配置成你自己的值。配置可中途修改，正常情况下立即生效。
 *
 * 参数获取方法可参考：<a href="https://cloud.tencent.com/document/product/441/6203">签名鉴权
 * 获取签名所需信息</a>
 */
private static void initBaseParameters() {
    // Required
    // AsrBaseConfig.applId = "YOUR_APP_ID_SET_HERE";
    // AsrBaseConfig.secretId = "YOUR_SECRET_ID";
    // AsrBaseConfig.secretKey = "YOUR_SECRET_KEY";

    // optional，根据自身需求设置配置值，不配则使用默认值。
    TtsConfig.VOLUME = 5; // 音量大小, 范围[0, 10]，默认为0，表示正常音量。
    // TtsConfig.REQUEST_ENCODE = RequestEncode.UTF_8; // 传入的文字所采用的编码，默认为utf-8
    // TtsConfig.SPEED = 0; // 语速，范围[-2, 2]。-2: 0.6倍; -1: 0.8倍; 0: 1.0倍（默认）; 1: 1.2倍; 2: 1.5倍。其他
    // 值：1.0 倍。
    // TtsConfig.VOICE_TYPE = 0; // 音色：0：亲和女声（默认）1：亲和男声 2：成熟男声 3：活力男声 4：温
```

暖女声 5 : 情感女声 6 : 情感男声

```
// TtsConfig.SAMPLE_RATE = 16000;// 音频采样率： 16000 : 16k ( 默认 ) ; 8000 : 8k  
// TtsConfig.PRIMARY_LANGUAGE = 1;// 主语言类型： 1-中文 ( 默认 ) 2-英文  
// TtsConfig.CODEC = CodeC.PCM; // 无需修改。暂未支持Opus方式。  
}
```

```
private void savePcmFile(byte[] response, File file) {
```

```
try {
```

```
FileOutputStream out = new FileOutputStream(file, false);
```

```
out.write(response);
```

```
out.close();
```

```
} catch (IOException e) {
```

```
System.err.println("Failed save data to: " + file + ", error: " + e.getMessage());
```

```
}
```

```
}
```

```
/**  
 * 将Pcm文件转换成wav文件保存起来。请将方法中的参数改成自己的语音文件对应的值，本方法仅供参考。  
 *  
 * 如需改成追加形式输出，请自行修改convert2Wav()方法中new FileOutputStream的参数。  
 */
```

```
private void saveToWavFile(byte[] responseBytes, File pcmFile, File wavFile) {
```

```
int bitNum = TtsConfig.SAMPLE_RATE == 16000 ? 16 : 8;
```

```
PcmUtils.convert2Wav(pcmFile, wavFile, TtsConfig.SAMPLE_RATE, 1, bitNum);
```

```
}
```

```
}
```

PHP SDK

最近更新时间：2019-06-18 19:53:54

接口请求域名：aai.cloud.tencent.com/tts

腾讯云语音合成技术（TTS）可以将任意文本转化为语音，实现让机器和应用张口说话。腾讯 TTS 技术可以应用到很多场景，例如，移动 App 语音播报新闻；智能设备语音提醒；支持车载导航语音合成的个性化语音播报。本接口内测期间免费使用。

开发环境

具体路径为：单击 [腾讯云控制台](#) 右上角您的账号，选择【访问管理】>【访问密钥】>【API 密钥管理】界面查看 AppID 和 key。

此版本 SDK 适用于 PHP5.4.16 及以上版本。

下载 SDK

流式语音合成 PHP SDK [下载地址](#)。

安装 SDK

源码安装。

根据下载地址下载源码，将源码中的 *.php 复制到项目中即可使用。

卸载 SDK

卸载方式即删除 *.php 即可。

获取用户信息

获取 AppID , SecretId 与 SecretKey

- 关于云 API 账号中的 AppID , SecretId 与 SecretKey 查询方法，可参考 [签名鉴权](#)。
- 具体路径为：单击 [腾讯云控制台](#) 右上角您的账号，选择【访问管理】>【访问密钥】>【API 密钥管理】界面查看 AppID 和 key。

更改用户信息配置文件

将查询到的用户信息更改到 php_tts_sdk/Config.php 中。

#需要配置成用户账号信息

```
static $SECRET_ID = "AKID*****";
static $SECRET_KEY = "kKm2*****";
static $APPID = 1255*****;
```

开发相关

请求参数

参数名称	必选	类型	描述
Action	是	String	本接口取值：TextToStreamAudio，不可更改。
AppId	是	Int	用户在腾讯云注册账号的 AppId，具体可以参考 获取用户信息 。
SecretId	是	String	用户在腾讯云注册账号 AppId 对应的 SecretId，获取方法同上。
Text	是	String	合成语音的源文本，最大支持800字符。
SessionId	是	String	一次请求对应一个 SessionId，会原样返回，建议传入类似于 uuid 的字符串防止重复。
ModelType	否	Int	模型类型，1：默认模型，此字段只需设置为1即可。
Volume	否	Float	音量大小，范围：[0，10]，分别对应11个等级的音量，默认值为0，代表正常音量。没有静音选项。 输入除以上整数之外的其他参数不生效，按默认值处理。
Speed	否	Int	语速，范围：[-2，2]分别对应不同语速： -2代表0.6倍 -1代表0.8倍 0代表1.0倍（默认） 1代表1.2倍 2代表1.5倍 输入除以上整数之外的其他参数不生效，按默认值处理。
VoiceType	否	Int	音色选择： 0：亲和女声（默认） 1：亲和男声 2：成熟男声 3：活力男声 4：温暖女声 5：情感女声 6：情感男声
PrimaryLanguage	否	Int	主语言类型： 1：中文（默认） 2：英文

参数名称	必选	类型	描述
SampleRate	否	Int	音频采样率： 16000:16k（默认） 8000:8k
Codec	否	String	返回音频格式：pcm：返回二进制 pcm 音频，使用简单，但数据量大。
ProjectId	否	Int	项目ID，可以根据控制台-账号中心-项目管理中的配置填写，如无配置请填写默认项目ID:0。
Timestamp	是	Int	当前 UNIX 时间戳，可记录发起 API 请求的时间。如果与当前时间相差过大，会引起签名过期错误。SDK 会自动赋值当前时间戳。
Expired	是	Int	签名的有效期，是一个符合 UNIX Epoch 时间戳规范的数值，单位为秒；Expired 必须大于 Timestamp 且 Expired-Timestamp 小于 90天。SDK 默认设置1小时。

PHP 快速入门示例

参考 `php_tts_sdk/TCloudTTS.php`

```
<?php
require ('TTSUtil.php');

# 1. 先修改好Config.php文件中的配置值。
# 2. TEXT为每次请求的文本，SESSION_ID建议每次请求修改成唯一id，例如uuid。
Config::$TEXT = "您好，五一节准备去哪里玩啊";
Config::$SESSION_ID = guid();
//echo "Session id : " . Config::$SESSION_ID . "\n";

# 2. 调用获取pcm格式音频
$result = getVoice();
$pcm_file = fopen('./test.pcm', "w");
fwrite($pcm_file, $result);
?>
```

Python SDK

最近更新时间：2019-06-18 19:53:59

流式语音合成 Python SDK [下载地址](#)。

接口请求域名：aai.cloud.tencent.com/tts

腾讯云语音合成技术（TTS）可以将任意文本转化为语音，实现让机器和应用张口说话。腾讯 TTS 技术可以应用到很多场景，例如，移动 App 语音播报新闻；智能设备语音提醒；支持车载导航语音合成的个性化语音播报。本接口内测期间免费使用。

开发环境

基本编译环境

python 2.7版本

requests

安装方法：pip install requests 或先下载，然后进入目录执行：python setup.py install

下载 [requests](#) 链接。

获取用户信息

获取 AppID , SecretId 与 SecretKey

- 关于云 API 账号中的 AppID , SecretId 与 SecretKey 查询方法，可参考 [签名鉴权](#)。
- 具体路径为：单击 [腾讯云控制台](#) 右上角您的账号，选择【访问管理】>【访问密钥】>【API 密钥管理】界面查看 AppID 和 key。

更改用户信息配置文件

将查询到的用户信息更改到 conf/tcloud_auth.ini 配置文件中。

#需要配置成用户账号信息

```
[authorization]
```

```
AppId=1259*****
```

```
SecretId=AKID*****
```

```
SecretKey=kFpwo*****
```

开发相关

请求参数

参数名称	必选	类型	描述
Action	是	String	本接口取值：TextToStreamAudio，不可更改。
AppId	是	Int	用户在腾讯云注册账号的 AppId，具体可以参考 获取用户信息 。
SecretId	是	String	用户在腾讯云注册账号 AppId 对应的 SecretId，获取方法同上。
Text	是	String	合成语音的源文本，最大支持800字符。
SessionId	是	String	一次请求对应一个 SessionId，会原样返回，建议传入类似于 uuid 的字符串防止重复。
ModelType	否	Int	模型类型，1：默认模型，此字段只需设置为1即可。
Volume	否	Float	音量大小，范围：[0, 10]，分别对应11个等级的音量，默认值为0，代表正常音量。没有静音选项。 输入除以上整数之外的其他参数不生效，按默认值处理。
Speed	否	Int	语速，范围：[-2, 2]分别对应不同语速： -2代表0.6倍 -1代表0.8倍 0代表1.0倍（默认） 1代表1.2倍 2代表1.5倍 输入除以上整数之外的其他参数不生效，按默认值处理。
VoiceType	否	Int	音色选择： 0：亲和女声（默认） 1：亲和男声 2：成熟男声 3：活力男声 4：温暖女声 5：情感女声 6：情感男声
PrimaryLanguage	否	Int	主语言类型： 1：中文（默认） 2：英文
SampleRate	否	Int	音频采样率： 16000:16k（默认） 8000:8k

参数名称	必选	类型	描述
Codec	否	String	返回音频格式： opus：返回多段含 opus 压缩分片音频，数据量小，建议使用（默认）。 pcm：返回二进制 pcm 音频，使用简单，但数据量大。
ProjectId	否	Int	项目ID，可以根据控制台-账号中心-项目管理中的配置填写，如无配置请填写默认项目ID:0。
Timestamp	是	Int	当前 UNIX 时间戳，可记录发起 API 请求的时间。如果与当前时间相差过大，会引起签名过期错误。SDK 会自动赋值当前时间戳。
Expired	是	Int	签名的有效期，是一个符合 UNIX Epoch 时间戳规范的数值，单位为秒；Expired 必须大于 Timestamp 且 Expired-Timestamp 小于 90天。SDK默认设置1小时。

具体参数配置可以参考：`conf/request_parameter.ini` 的默认配置

请求demo

```
python tcloud_tts.py
```

Python 快速入门例子

参考 `python_tts_sdk/tcloud_tts.py`

```
def task_process():
    req = request()
    req.init()
    auth = authorization()
    auth.init()

    #request_data = collections.OrderedDict()
    request_data = dict()
    request_data['Action'] = req.Action
    request_data['AppId'] = auth.AppId
    request_data['Codec'] = req.Codec
    request_data['Expired'] = int(time.time()) + auth.Expired
    request_data['ModelType'] = req.ModelType
    request_data['PrimaryLanguage'] = req.PrimaryLanguage
    request_data['ProjectId'] = req.ProjectId
```

```
request_data['SampleRate'] = req.SampleRate
request_data['SecretId'] = auth.SecretId
request_data['SessionId'] = req.SessionId
request_data['Speed'] = req.Speed
request_data['Text'] = req.Text
request_data['Timestamp'] = int(time.time())
request_data['VoiceType'] = req.VoiceType
request_data['Volume'] = req.Volume

signature = auth.generate_sign(request_data = request_data)
header = {
    "Content-Type": "application/json",
    "Authorization": str(signature)
}
url = "https://aai.cloud.tencent.com/tts"

r = requests.post(url, headers=header, data=json.dumps(request_data), stream = True)
'''
if str(r.content).find("Error") != -1 :
    print(r.content)
    return
'''

i = 1
wavfile = wave.open('test.wav', 'wb')
wavfile.setparams((1, 2, 16000, 0, 'NONE', 'NONE'))
for chunk in r.iter_content(1000):
    if (i == 1) & (str(chunk).find("Error") != -1) :
        print(chunk)
    return
    i = i + 1
    wavfile.writeframes(chunk)

wavfile.close()
```