

TTS 文本转音频功能使用与讲解

1. 功能简介

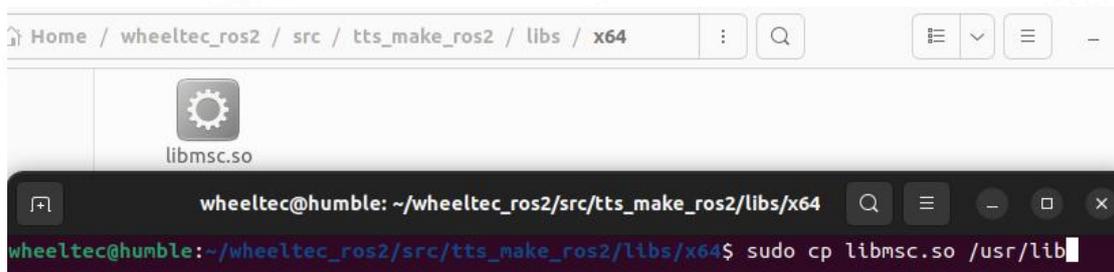
TTS 文本转音频功能是通过 TTS 功能包将我们所需要的文本转换并输出音频文件，并且我们可以设置输出音频的参数如音量、音调、语速等。

2. 使用方法

使用前需要确保上位机连接小车 WiFi，并已经 SSH 远程登录到小车端。

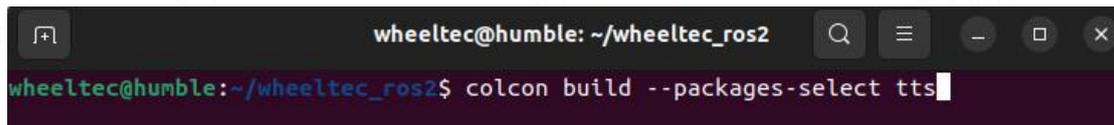
1) 编译功能包

编译功能包之前需要进行 so 库的配置，如下图所示，根据运行的架构配置不一样的 so 库如下图就是配置的 x64 架构，微主控配置的是 Arm64 架构。



配置 so 库

然后进入工作空间目录下进行功能包编译如下图所示。



编译 TTS 功能包

2) 合成音频的文本

运行功能之前，需要输入合成音频的文本，位置如下图所示。

```
1 import os
2 from launch import LaunchDescription
3 from launch_ros.actions import Node
4 from ament_index_python.packages import get_package_share_directory
5
6 def generate_launch_description():
7     ld = LaunchDescription()
8
9     tts_dir = get_package_share_directory('tts')
10    tts_launch_dir = os.path.join(tts_dir, 'launch')
11    tts_config = os.path.join(tts_dir, 'config', 'tts_params.yaml')
12    resource_param = {"source_path": tts_dir}
13
14    #合成语音的文本输入
15    tts_text = {"tts_text": "你好小微"}
16
17    tts_make = Node(
18        package="tts",
19        executable="tts_node",
20        output='screen',
21        parameters=[resource_param,
22                    tts_text,
23                    tts_config]
24    )
```

文本位置

如图可知合成文本为“你好小微”，TTS 功能运行后会生成一个包含此文本内容的一个音频文件。

3) 打开终端输入启动 TTS 文本转音频功能的指令

```
ros2 launch tts tts_make.launch.py
```

```
wheeltec@humble:~$ ros2 launch tts tts_make.launch.py
[INFO] [launch]: All log files can be found below /home/wheeltec/.ros/log/2022-06-01-10-01-33-320575-humble-3439
[INFO] [launch]: Default logging verbosity is set to INFO
[INFO] [tts_node-1]: process started with pid [3440]
[tts_node-1] [INFO] [1654048893.727286100] [tts_node]: tts_node node init!
[tts_node-1]
[tts_node-1] [INFO] [1654048893.959609874] [tts_node]: tts_node done!
[tts_node-1] [INFO] [1654048893.959661683] [tts_node]: tts_node over!
[tts_node-1]
[tts_node-1] #####
[tts_node-1] ## 语音合成 (Text To Speech, TTS) 技术能够自动将任意文字实时转换为连续的 ##
[tts_node-1] ## 自然语音, 是一种能够在任何时间、任何地点, 向任何人提供语音信息服务的 ##
[tts_node-1] ## 高效便捷手段, 非常符合信息时代海量数据、动态更新和个性化查询的需求。 ##
[tts_node-1] #####
[tts_node-1]
[tts_node-1] 开始合成 ...
[tts_node-1] 正在合成 ...
[tts_node-1]
[tts_node-1] 合成完毕
[INFO] [tts_node-1]: process has finished cleanly [pid 3440]
wheeltec@humble:~$
```

TTS 功能运行终端

3. 注意事项

① 音频文件夹

生成完音频后会自动保存到/audio 文件夹下, 文件名为运行 TTS 功能时的时间, 位置如下图所示。

/ wheeltec_ros2 / src / tts_make_ros2 / audio

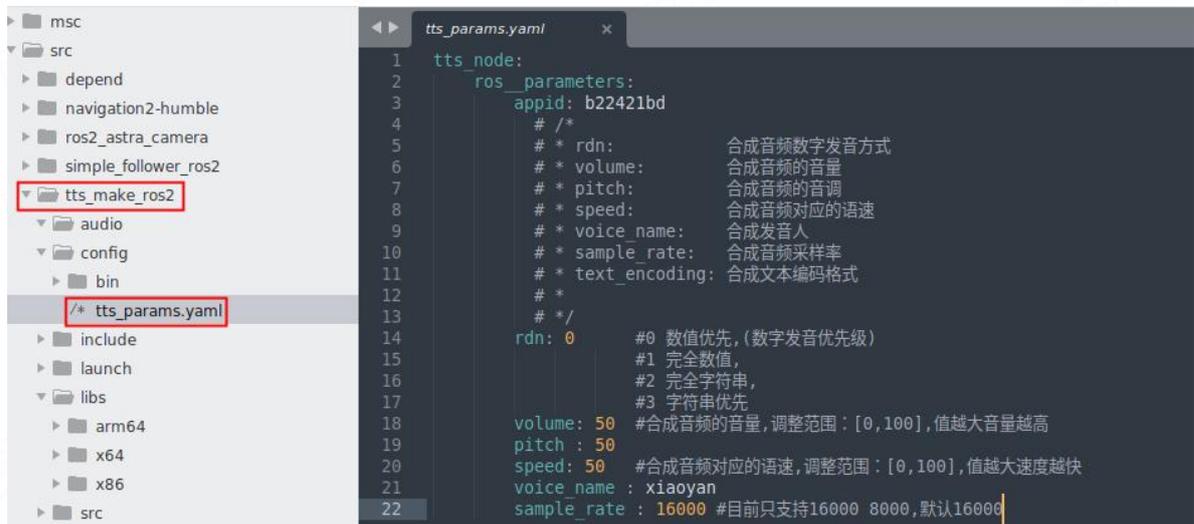


2022-06-01_10:01:33.wav

音频文件放置位置

② 音频参数

目前外置了合成音频的参数，例如合成音频的音量大小、语速等参数，可以直接在 `tts_params.yaml` 文件里面修改不需要进行编译。具体位置如下图所示。



音频参数文件

③ APPID 和离线资源替换

如果运行功能终端出现“11212”错误，则表明离线资源过期，此时便需要进行离线资源替换，下面为替换的整个流程。

Step1: 注册讯飞开放平台账号

打开讯飞开放平台网页，网址 <https://www.xfyun.cn>，注册账号。

Step2: 创建新应用

选择控制台——我的应用——创建新应用——填好对应资料后提交，如下图



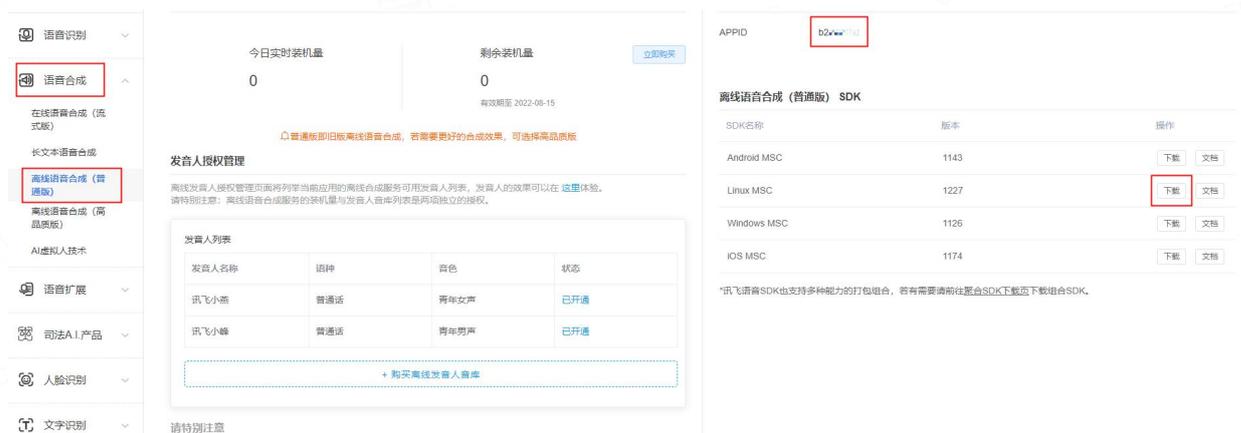
讯飞开放平台创建新应用步骤 1



讯飞开放平台创建新应用步骤 2

Step3: 下载 SDK

进入刚创建的应用，下载离线语音合成（普通版）SDK



下载离线语音合成（普通版）SDK

创建新应用会生成一个 APPID，与 SDK 中离线语音资源文件匹配。创建新应用试用期 90 天，若不购买服务，可以再次创建新应用重新替换 APPID 和离线

语音资源文件获得新的 90 天试用，每个实名制用户可以免费申请 5 个应用。

将 SDK 压缩包解压后，离线语音资源路径如下：

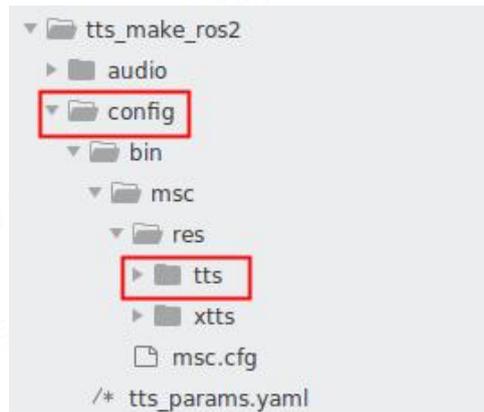


SDK 离线语音资源路径

注：如果官网应用中显示的 appid 与下载后 SDK 后缀带有的 appid 不符则以 SDK 后缀的 appid 为主。

Step4: 替换 common.jet

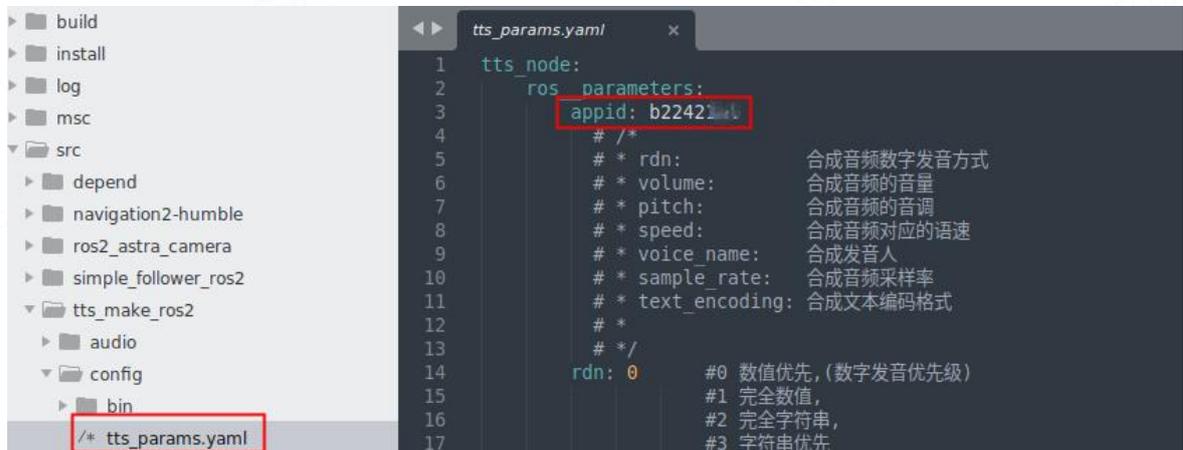
tts 功能包中离线语音资源路径如下：



tts 语音资源路径

Step5: 替换 APPID

存放 APPID 的文件路径如下图，修改为对应应用 SDK 中的 ID 后保存退出。

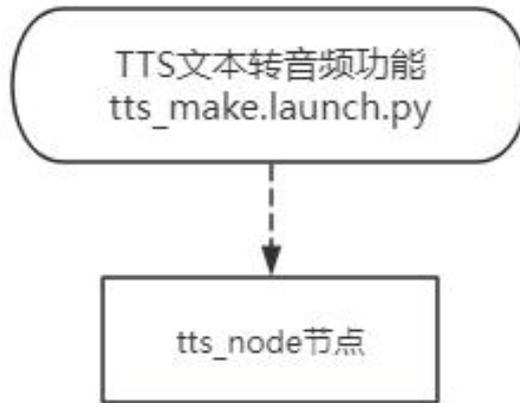


修改 APPID

4. 功能讲解

TTS 文本转音频功能是通过启动文件 `tts_make.launch.py` 来开启的

① 启动 TTS 文本转音频功能



TTS 文本转音频功能启动框架

② TTS 文本转音频功能启动文件介绍

```
import os
from launch import LaunchDescription
from launch_ros.actions import Node
from ament_index_python.packages import get_package_share_directory

def generate_launch_description():
    ld = LaunchDescription()

    tts_dir = get_package_share_directory('tts')
    tts_launch_dir = os.path.join(tts_dir, 'launch')
    tts_config = os.path.join(tts_dir, 'config', 'tts_params.yaml')
    resource_param = {"source_path": tts_dir}

    #合成语音的文本输入
    tts_text = {"tts_text": "TTS 功能运行后会生成一个包含此文本内容的一个音频文件"}

    tts_make = Node(
        package="tts",
        executable="tts_node",
        output='screen',
        parameters=[resource_param,
                    tts_text,
                    tts_config]
    )
```

```
ld.add_action(tts_make)
```

```
return ld
```

在 `tts_make.launch.py` 文件中，首先获取参数来设置合成音频的配置，如音量音调等，然后设置参数，包括 `tts` 功能包的路径和需要生成音频的文本，最后开启 `tts` 节点，整体功能比较简单。