


[TTS] VITS DEMO

REFERENCE

<https://github.com/jaywalnut310/vits>

Google Colaboratory

 https://colab.research.google.com/drive/1CO61pZizDj7en71NQG_aqqKdGaA_SaBf?usp=sharing



GOOGLE COLAB

- `!pip install -r requirements.txt` 이후 런타임 재시작해야 함
- 하지만 알 수 없는 이유로 세션 재시작이 안됨
- 주피터에서 시도하기

JUPYTER

설치가 안됨

sudo 접근 권한이 없음

가상환경 파이썬 버전 설정을 해도 3.9가 됨

1 가상환경 : python 3.7

① 가상환경 생성

```
conda create --name ttsenv python=3.7s
```

② 가상환경 활성화

```
source activate ttsenv
```

③ jupyter notebook 설치

```
pip install jupyter notebook
```

④ 커널 연결

```
python -m ipykernel install --user --name ttsenv --display-name
```

2 pre-requisites

requirements.txt 설치

- jupyter terminal 이동
- 가상환경 ttsenv 활성화
- requirements.txt 설치

```
pip install -r requirements.txt
```

3 python 3.7 설치

① 터미널에서 wget으로 python 3.7.16 설치

```
pip install wget
```

```
wget https://https://www.python.org/ftp/python/3.7.16/Python-
```

혹은

```
python
```

```
wget.download("https://https://www.python.org/ftp/python/3.7.16/python-3.7.16.tgz")
```

② 압축 해지 및 컴파일 빌드

```
tar xvf Python-3.7.16.tgz
```

```
cd Python-3.7.16
```

```
./configure --enable-optimizations
```

③ 빌드

```
make -j 1
```

```
make altinstall
```

4 python 환경변수 설정

```
import os
new_python_path = "/home/j-i10c109/.conda/envs/ttsenv/bin"
current_path = os.environ.get('PATH', '')
os.environ['PATH'] = f"{new_python_path}:{current_path}"
```

- 파이썬 버전 확인

```
!python --version
```

- 환경 변수 확인

```
for x in os.environ:
    print((x, os.getenv(x)))
```

- !which python

확장 모듈 빌드

- 터미널에서 실행
- 가상환경 활성화

```
python setup.py build_ext --build-lib=.
```

espeak 설치 : 사용자 수준에서 설치하는 방법

- sudo 접근 권한 없음
- 전체 코드

```
git clone https://github.com/espeak-ng/espeak-ng.git
cd espeak-ng
./configure --prefix=$HOME/local
make
make install
```

1.

```
git clone https://github.com/espeak-ng/espeak-ng.git
cd espeak-ng
```

2.

```
./configure --prefix=$HOME/j-i10c109/.local
```

- configure 스크립트가 없음
- configure.ac 만 있음
- configure 스크립트 생성해야 함

configure 스크립트 생성

```
autoreconf -i
```

`./configure --prefix=$HOME/local` 재실행 ⇒ ERROR

● TROUBLE SHOOTING : config.status: error ●

error message

config.status: error: cannot find input file: 'Makefile.in'

- autoconf 및 automake를 사용자 수준에서 설치

autoconf, automake를 사용자 수준에서 설치

1. 사용자 홈 디렉토리에 설치할 디렉토리 생성

```
mkdir -p $HOME/j-i10c109/.local/bin
```

2. 이 디렉토리를 PATH에 추가

```
cd bin
```

```
echo 'export PATH=$HOME/j-i10c109/.local/bin:$PATH' >> ~/.
```

- 확인

```
echo $PATH
```

`/home/j-i10c109/.local/bin` 이 있어야 함

- 현재 쉘 세션에 있는 Bash 사용자 환경 설정을 업데이트한다

```
source ~/.bashrc
```

3. autoconf 다운로드 및 빌드

```
wget http://ftp.gnu.org/gnu/autoconf/autoconf-latest.tar.g
tar -xzf autoconf-latest.tar.gz
cd autoconf-*
./configure --prefix=$HOME/j-i10c109/.local
make
```

```
make install  
cd ..
```

4. automake 다운로드 및 빌드

```
wget http://ftp.gnu.org/gnu/automake/automake-latest.tar.g  
tar -xzvf automake-latest.tar.gz  
cd automake-  
./configure --prefix=$HOME/local  
make  
make install  
cd ..
```