

tensorflowTTS 설치 및 실행

(<https://github.com/TensorSpeech/TensorFlowTTS>)

(<https://enchiridion.tistory.com/69>)

Features

- High performance on Speech Synthesis.
- Be able to fine-tune on other languages.
- Fast, Scalable, and Reliable.
- Suitable for deployment.
- Easy to implement a new model, based-on abstract class.
- Mixed precision to speed-up training if possible.
- Support Single/Multi GPU gradient Accumulate.
- Support both Single/Multi GPU in base trainer class.
- TFlite conversion for all supported models.
- Android example.
- Support many languages (currently, we support Chinese, Korean, English, French and German)
- Support C++ inference.
- Support Convert weight for some models from PyTorch to TensorFlow to accelerate speed.

환경

Python 3.7+	
Cuda 10.1	
CuDNN 7.6.5	
Tensorflow 2.2/2.3/2.4/2.5/2.6	# 또는 tensorflow-gpu 2.6.0
Tensorflow Addons >= 0.10.0	

c:

cd \ai

python 3.7로 가상환경 생성

conda create -n tensorflowtts git python=3.7

소스 git

git clone <https://github.com/TensorSpeech/TensorFlowTTS.git>

소스 폴더로 이동

cd tensorflowtts

가상환경 활성화

conda activate tensorflowtts

관련 패키지 설치

pip install .

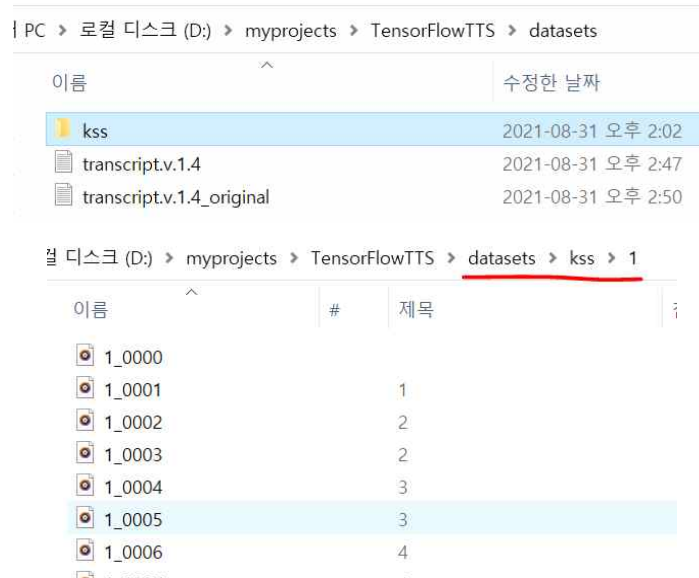
HDF5(계층적데이터형식, 빅데이터 처리를 위한 파일 형식)을 python에서 사용하기 위한 모듈 설치
conda install h5py

#약어, 숫자, 타임 스탬프 등을 포함한 독일어 텍스트를 정리하고

음역 (예 : 정규화)하는 Python 모듈 설치

pip install git+https://github.com/repodiac/german_transliterate

데이터 구조



이름	수정된 날짜
kss	2021-08-31 오후 2:02
transcript.v.1.4	2021-08-31 오후 2:47
transcript.v.1.4_original	2021-08-31 오후 2:50

이름	#	제목
1_0000		
1_0001	1	
1_0002	2	
1_0003	2	
1_0004	3	
1_0005	3	
1_0006	4	

Preprocessing

The preprocessing has two steps:

1. Preprocess audio features

- Convert characters to IDs
- Compute mel spectrograms
- Normalize mel spectrograms to [-1, 1] range
- Split the dataset into train and validation
- Compute the mean and standard deviation of multiple features from the **training split**

2. Standardize mel spectrogram based on computed statistics

전처리 데이터 정리- 첫 번째 줄 삭제 - 전처리 시 에러 방지

transcript.v.1.4 - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

1/1_0000.wav|그는 관창은 척하려고 애쓰는 것 같았다.|그는 관창은 척하려고
1/1_0001.wav|그녀의 사랑을 얻기 위해 애썼지만 헛수고였다.|그녀의 사랑을
1/1_0002.wav|용돈을 아껴 써라.|용돈을 아껴 써라.|
1/1_0003.wav|그는 아내를 많이 아낀다.|그는 아내를 많이 아낀다.|
1/1_0004.wav|그 애 전화번호 알아?|그 애 전화번호 알아?|
1/1_0005.wav|차에 대해 잘 아세요?|차에 대해 잘 아세요?|
1/1_0006.wav|거기 도착하면 나한테 알려 줘.|거기 도착하면 나한테 알려 줘.|
1/1_0007.wav|그들은 내가 시험에 떨어졌다고 알려 왔다.|그들은 내가 시험에
1/1_0008.wav|나는 살아오면서 감기를 앓은 적이 한 번도 없다.|나는 살아오면
1/1_0009.wav|사흘 동안 심하게 몸살을 앓았어요.|사흘 동안 심하게 몸살을 앓

[1] 전처리

```
tensorflow-tts-preprocess --rootdir ./datasets --outdir ./datasets/dump_kss --config
preprocess/kss_preprocess.yaml --dataset kss
```

```
(tensorflowtts) D:\myprojects\TensorFlowTTS>tensorflow-tts-preprocess --rootdir ./datasets --outdir ./dump_kss --config
preprocess/kss_preprocess.yaml --dataset kss
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data] C:\Users\pkkim\AppData\Roaming\nltk_data...
[nltk_data] Unzipping taggers\averaged_perceptron_tagger.zip.
[nltk_data] Downloading package cmudict to
[nltk_data] C:\Users\pkkim\AppData\Roaming\nltk_data...
[nltk_data] Unzipping corpora\cmudict.zip.
2021-08-31 14:22:03,082 (preprocess:378) INFO: Selected 'kss' processor.
2021-08-31 14:22:03,183 (preprocess:417) INFO: Training items: 12211
2021-08-31 14:22:03,184 (preprocess:418) INFO: Validation items: 643
[Preprocessing train]: 3% | 355/12211 [00:30<16:37, 11.89it/s]W
ARNING:root:1_0922 causes clipping. It is better to reconsider global gain scale value.
[Preprocessing train]: 3% | 411/12211 [00:34<24:36, 7.99it/s]W
ARNING:root:1_0014 causes clipping. It is better to reconsider global gain scale value.
[Preprocessing train]: 4% | 468/12211 [00:38<10:49, 18.07it/s]W
ARNING:root:1_0554 causes clipping. It is better to reconsider global gain scale value.
[Preprocessing train]: 5% | 561/12211 [00:44<17:30, 11.09it/s]W
ARNING:root:1_0696 causes clipping. It is better to reconsider global gain scale value.
[Preprocessing train]: 5% | 627/12211 [00:48<07:57, 24.27it/s]W
ARNING:root:1_0668 causes clipping. It is better to reconsider global gain scale value.
[Preprocessing train]: 9% | 1105/12211 [01:20<06:48, 27.18it/s]
```

.....

```
[Preprocessing train]: 94% | 11448/12210 [13:03<00:37, 20.39it/s]W
ARNING:root:1_0668 causes clipping. It is better to reconsider global gain scale value.
[Preprocessing train]: 98% | 11914/12210 [13:34<00:11, 26.88it/s]W
ARNING:root:2_0609 causes clipping. It is better to reconsider global gain scale value.
[Preprocessing train]: 100% | 12210/12210 [13:54<00:00, 14.63it/s]
2021-08-31 15:01:33,056 (preprocess:478) INFO: Saving computed statistics.
[Preprocessing valid]: 7% | 48/643 [00:01<00:18, 32.92it/s]W
ARNING:root:1_0654 causes clipping. It is better to reconsider global gain scale value.
[Preprocessing valid]: 9% | 55/643 [00:02<00:50, 11.54it/s]W
ARNING:root:1_0386 causes clipping. It is better to reconsider global gain scale value.
[Preprocessing valid]: 100% | 643/643 [00:41<00:00, 15.49it/s]
```

PC > 로컬 디스크 (D:) > myprojects > TensorFlowTTS > datasets



datasets 검색

이름	수정한 날짜	유형	크기
kss	2021-08-31 오후 2:02	파일 폴더	
transcript.v.1.4	2021-08-31 오후 2:47	텍스트 문서	3,225KB
transcript.v.1.4_original	2021-08-31 오후 2:50	텍스트 문서	3,225KB
<u>dump_kss</u>	2021-08-31 오후 3:01	파일 폴더	

[2] Normalize

```
tensorflow-tts-normalize --rootdir ./datasets/dump_kss --outdir ./datasets/dump_kss --config preprocess/kss_preprocess.yaml --dataset kss
```

컬 디스크 (D:) > myprojects > TensorFlowTTS > datasets > dump_kss dump_kss 검색

이름	수정한 날짜	유형	크기
train	2021-08-31 오후 3:19	파일 폴더	
valid	2021-08-31 오후 3:19	파일 폴더	
kss_mapper	2021-08-31 오후 2:47	JSON 파일	3KB
stats.npy	2021-08-31 오후 3:01	NPY 파일	1KB
stats_energy.npy	2021-08-31 오후 3:01	NPY 파일	1KB
stats_f0.npy	2021-08-31 오후 3:01	NPY 파일	1KB
train_utt_ids.npy	2021-08-31 오후 2:47	NPY 파일	287KB
valid_utt_ids.npy	2021-08-31 오후 2:47	NPY 파일	16KB

```
(tensorflowtts) D:\myprojects\TensorFlowTTS>tensorflow-tts-normalize --rootdir ./datasets/dump_kss --outdir ./datasets/dump_kss --config preprocess/kss_preprocess.yaml --dataset kss
2021-08-31 15:19:35.645 (preprocess:531) INFO: Files to normalize: 12946
[Normalizing]: 100% | 12946/12946 [02:55<00:00, 73.74it/s]
```

myprojects > TensorFlowTTS > datasets > dump_kss > train train

이름	수정한 날짜	유형
ids	2021-08-31 오후 3:01	파일 폴더
norm-feats ✓	2021-08-31 오후 3:23	파일 폴더
raw-energies	2021-08-31 오후 3:01	파일 폴더
raw-f0	2021-08-31 오후 3:01	파일 폴더
raw-feats	2021-08-31 오후 3:01	파일 폴더
wavs	2021-08-31 오후 3:01	파일 폴더

cudnn64_8.dll 파일 못찾는다는 에러 방지 위해 cudnn 설치

```
conda install cudnn cudatoolkit
```

pretrained model 복사

컬 디스크 (D:) > myprojects > TensorFlowTTS > model-20210831T050255Z-001 > model > tacotron2.v1

이름	수정한 날짜	유형	크기
model-40000.h5	2020-07-17 오후 8:07	H5 파일	124,835KB
✓ model-100000.h5	2020-07-20 오전 4:34	H5 파일	124,835KB

| PC > 로컬 디스크 (D:) > myprojects > TensorFlowTTS

이름	수정된 날짜	유형	크기
tacotron-100000.h5	2020-07-20 오전 4:34	H5 파일	124,835KB
README	2021-08-30 오후 9:51	MD 파일	23KB
LICENSE	2021-08-30 오후 9:51	파일	12KB
setup	2021-08-30 오후 9:51	Python 원본 파일	4KB
.gitignore	2021-08-30 오후 9:51	Git Ignore 원본 파일	1KB
dockerfile	2021-08-30 오후 9:51	파일	1KB
3.10	2021-08-30 오후 11:16	10 파일	1KB
setup	2021-08-30 오후 9:51	CFG 파일	1KB
docker-compose	2021-08-30 오후 9:51	Yaml 원본 파일	1KB
.gitattributes	2021-08-30 오후 9:51	Git Attributes 원본 ...	1KB
.git	2021-08-30 오후 9:51	파일 폴더	
.github	2021-08-30 오후 9:51	파일 폴더	
datasets	2021-08-31 오후 3:18	파일 폴더	

GPU 메모리 할당 문제 해결을 위하여 extract_duration.py에 소스 추가

```
extract_duration.py 4 ●
D: > myprojects > TensorFlowTTS > examples > tacotron2 > extract_duration.py > ...

21 import sys
22
23 sys.path.append(".")
24
25 import matplotlib.pyplot as plt
26 import numpy as np
27 import tensorflow as tf
28 import yaml
29 from tqdm import tqdm
30
31 from examples.tacotron2.tacotron_dataset import CharacterMelDataset
32 from tensorflow_tts.configs import Tacotron2Config
33 from tensorflow_tts.models import TFTacotron2
34
35 ### added by pk
36 config = tf.compat.v1.ConfigProto()
37 config.gpu_options.allow_growth = True
38 config.log_device_placement = True
39 session = tf.compat.v1.Session(config=config)
40 ###
41
42 @jit(nopython=True)
```

Duration 추출

```
python examples/tacotron2/extract_duration.py --rootdir ./datasets/dump_kss/train/ --outdir
./datasets/dump_kss/train/durations/ --checkpoint ./tacotron2-100k.h5 --use-norm 1
--config ./examples/tacotron2/conf/tacotron2.kss.v1.yaml --batch-size 32
```

```
python examples/tacotron2/extract_duration.py --rootdir ./datasets/dump_kss/valid/ --outdir
./datasets/dump_kss/valid/durations/ --checkpoint ./tacotron2-100k.h5 --use-norm 1
--config ./examples/tacotron2/conf/tacotron2.kss.v1.yaml --batch-size 32
```

학습

```
python examples/tacotron2/train_tacotron2.py --train-dir ./datasets/dump_kss/train/ --dev-dir
./datasets/dump_kss/valid/ --outdir ./examples/tacotron2/exp/train.tacotron2.kss.v1/ --config
./examples/tacotron2/conf/tacotron2.kss.v2.yaml --use-norm 1 --pretrained
./tacotron2-100000.h5
```