

TTS 3차시 <모델 학습>

AICS 윤서영 매니저

CONTENTS

1. 학습 준비

- 음성 합성을 위해 필요한 데이터 구성
- 필수 리눅스 명령어

2. TTS 학습

- Train, test, validation
- Hyper parameter
- Tacotron2
- Waveglow

1. 학습 준비

1. 필요 기본 지식

1. 음성 합성을 위해 필요한 데이터 구성

Wav	txt
22k (22050), wav format	UTF-8 without BOM

2. 학습을 위해 필수로 알아야 하는 리눅스 명령어

- 1) cd.. : 이전 디렉토리로 이동
- 2) cd 경로 : 해당 경로로 이동
- 3) cp 파일 경로/이름: 원하는 파일을 원하는 경로에 붙여넣기
- 4) mv 파일 이동경로: 원하는 파일을 원하는 경로에 옮기기
- 5) ls: 현재 디렉토리의 파일 보기
- 6) pwd: 현재 위치한 경로 확인
- 7) Ctrl + c : 종료 / 나가기 / 취소
- 8) vi : python, json, yaml 파일 등을 보고 수정하는 명령어

2. TTS docker 설정

1. Docker 버전 확인 방법

- 버전 확인 페이지 (repo) : <https://docker.maum.ai:8443/>

2. Docker image 받기

- 1) Tacotron2 : `docker pull docker.maum.ai:443/brain/tacotron2:1.2.7`
- 2) Waveglow : `docker pull docker.maum.ai:443/brain/waveglow:1.2.7`
※ 1.2.7 = image version

참고) Docker 명령어

- 1) `docker images` : 현재 서버에 설치 되어있는 docker image list를 보여줌
- 2) `docker ps` : 현재 서버에 떠있는 docket container list를 보여줌 (running)
- 3) `docker ps -a` : 서버에 있는 모든 docker container list를 보여줌
- 4) `docker stop` : 현재 떠있는 docker container를 죽임
- 5) `docker rm` : 죽어있는 docker container를 삭제 (죽어있는 container에만 적용됨)
- 6) `docker exec -it container_name bash` : 컨테이너 접속
- 7) `Ctrl + p / q` : container에서 나가기

2. TTS 학습

1. TTS docker 설정

1. Docker container 만들기

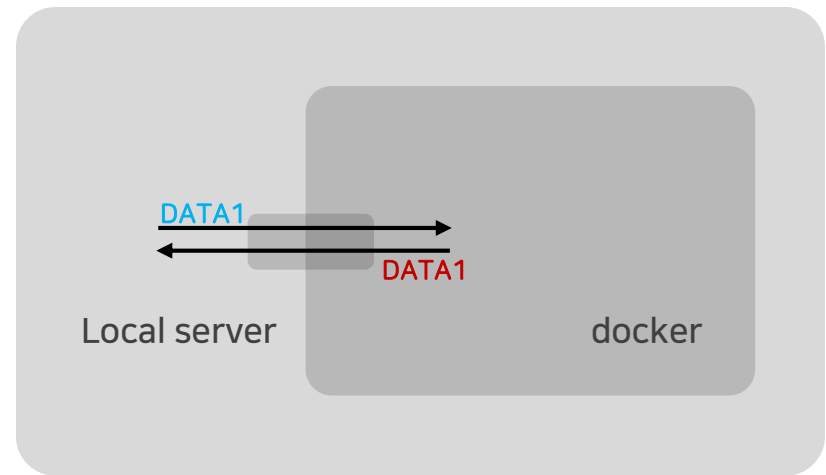
1) tacotron2

```
docker run -itd --ipc host --gpus '"device=0,1"' -v /DATA1:/DATA1 -p 6006:6006 -e LC_ALL=C.UTF-8 --name  
container_name docker.maum.ai:443/brain/tacotron2:1.2.7
```

2) Waveglow

```
docker run -itd --ipc host --gpus '"device=0,1"' -v /DATA1:/DATA1 -p 6007:6006 -e LC_ALL=C.UTF-8 --name  
container_name docker.maum.ai:443/brain/waveglow:1.2.7
```

- gpus : 사용할 gpu number
 - v : local server와의 공유 폴더
 - p : local server와의 공유 포트
 - name : 사용할 container 이름
- ※ local server 및 container 이름은 겹치면 안됨



2. TTS docker 학습

1. tacotron2 학습

- tacotron2의 경우 1, 2차 학습으로 나뉘짐
- 경로) `cd /root/tacotron2`
- `python -m multiproc -l log_path -o ckpt_path trainerd.py -c baseline_root --warm_start --hparams=distributed_run=True,resource_root='~resources_root',training_files='data_folder/train.txt',validation_files='data_folder/val.txt',speaker=['speaker ID'],mask_padding=True,use_eos=False,batch_size=16`
- 1차 학습시 `'--warm_start'`를 붙여 학습하며, 보통 checkpoint 500 ~ 1000 사이에 학습 종료
- 2차 학습시 `'--warm_start'`를 제거 후, 1차 학습에서 생성된 checkpoint를 baseline으로 변경하여 학습
- l : log를 쌓을 경로
- o : checkpoint를 쌓을 경로
- c : 사용할 baseline 경로
- resources_root : 원본 데이터 경로 ~ resources/
- training_files : ~train.txt가 있는 상대경로 (예, X/X_train.txt)
- validation_files : ~val.txt가 있는 상대경로 (예, Y/Y_train.txt)
- speaker = 사용할 speaker ID
- ※ baseline : 2019_09_02_fp16_22k_kor_base_2_178000

참고) sample rate error

- sample rate error가 발생한다면, wav 데이터가 있는 내부 폴더에서 하단 명령어 실행
- `for i in *wav; do ffmpeg -i "$i" -ar 22050 -hide_banner -loglevel panic /output경로/"$i"; done`

2. TTS docker 학습

1. waveglow 학습

- python distributed.py -c train/config.json -l log_path -o checkpoint_path
- l : log를 쌓을 경로
- o : checkpoint를 쌓을 경로

```
    "checkpoint_path": "/DATA1/yoong/tts/trained/waveglow/waveglow_194000_m_re",  
  },  
  "data_config": {  
    "resource_root": "/DATA1/yoong/tts/resources/",  
    "training_files": "/DATA1/yoong/tts/resources/Eric/Eric_train.txt",  
    "segment_length": 16000,  
    "sampling_rate": 22050,  
    "filter_length": 1024,  
    "hop_length": 256,  
    "win_length": 1024,  
    "mel_fmin": 0.0,  
    "mel_fmax": 8000.0,  
    "load_mel_from_disk": false  
  },  
  "validation_files": "/DATA1/yoong/tts/resources/Eric/Eric_val.txt",
```

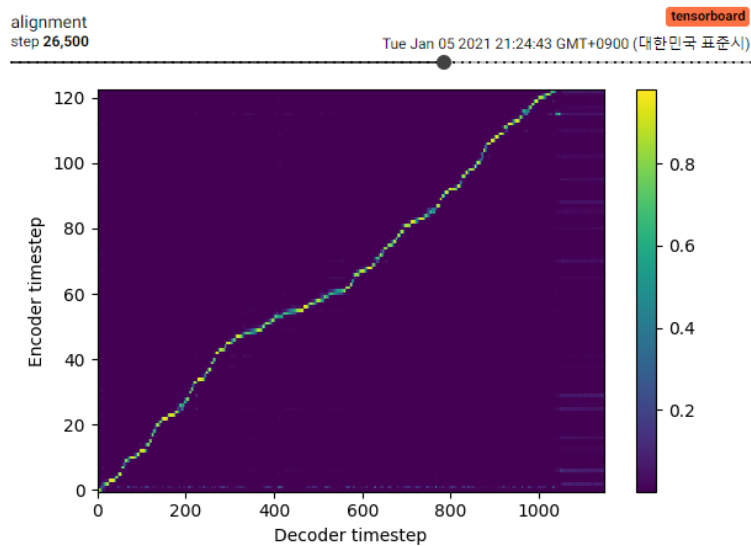
참고) baseline

- 남성) waveglow_194000_m_re
- 여성) waveglow_226000_kss_kva_22k_re / waveglow_260000_eng_22k_re

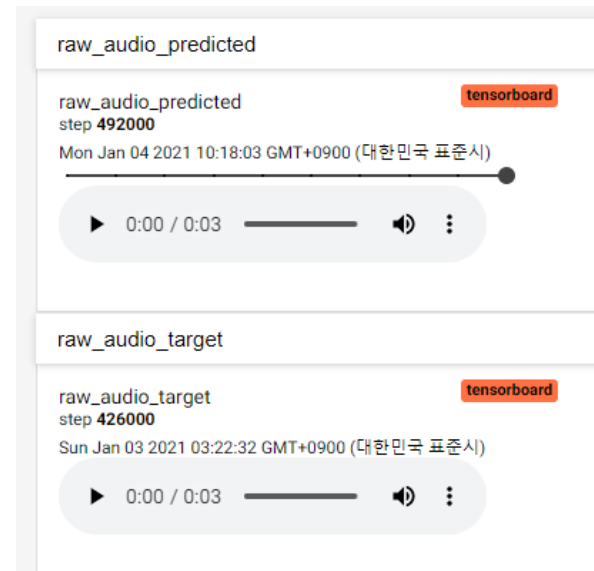
3. tensorboard 띄우기

1. Tensorboard 띄우기

- 경로 : 띄우고자 하는 모델의 log 경로
- `tensorboard --samples_per_plugin=scalars=500,images=0 --logdir=.`



Tacotron2 tensorboard 예시



waveglow tensorboard 예시

Thank you