

Readme.md

실행 방법

- Infer.py를 이용하여 실행 가능
- Argument 목록
 - `--model`
 - Default: LanguageBind/LanguageBind_Audio
 - 다른 사용가능한 모델로는 LanguageBind/LanguageBind_FT가 존재
 - `--checkpoint_path`
 - Default: None
 - Finetuning된 Checkpoint를 사용
 - `--file`
 - 처리할 음성 파일의 경로
 - wav, m4a 등 파일 확장자와 관계없이 음성파일이면 가능
 - `--split_time`
 - Audio 전처리 간 자를 시간 간격
 - Default의 값은 1로, 이는 1초간격으로 오디오를 잘라서 샘플링을 하는 것을 의미
 - `--k`
 - 추후 결과를 뽑을 때 상위 몇 개의 값을 뽑을지 정하는 숫자
 - `--k 2` 이면 상위 2개의 결과를 뽑아서 결과를 도출한다.
 - `--output`
 - 출력물의 경로
 - 따로 명시하지 않으면 `--file`과 동일한 경로
- 예시 명령어

```
python infer.py --checkpoint_path "final_models/final.pt" --f
```

Final Model / Files

- github와 google drive의 pt 파일을 다운받아서 pt 파일의 경로를 `--checkpoint_path`의 인자로 작성
- model.bin 같은 경우 원본 LanguageBind의 github에서 다운
- pt file link

https://drive.google.com/file/d/1pWM-cAbM1xpAtRWgWq-_ZtjT_mUhsyzH/view?usp=drive_link

- Github Link
<https://github.com/okaybody10/sed>

Metric

- Collars / PSDS
 - Collars
 - Google Drive에 보이는 파일 순 차례대로 Sample 1 ~ Sample 8까지 명명

▼ Sample 1

- Pretrained Model

```
"F1 of music": 0.08695652173913045,  
"Precision of music": 0.05555555555555555,  
"Recall of music": 0.2,
```

- Finetune Model

```
"F1 of music": 0.15384615384615385,  
"Precision of music": 0.125,  
"Recall of music": 0.2,
```

▼ Sample 2

- Pretrained Model

```
"F1 of music": 0.04347826086956522,  
"Precision of music": 0.022727272727272728,  
"Recall of music": 0.5,
```

- Finetune Model

```
"F1 of music": 0.018181818181818181,  
"Precision of music": 0.009259259259259259,  
"Recall of music": 0.5,
```

▼ Sample 3

- Pretrained Model

```
"F1 of music": 0.0,  
"Precision of music": 1.0,  
"Recall of music": 0.0,
```

- Finetune Model

```
"F1 of music": 0.015503875968992248,  
"Precision of music": 0.008264462809917356,  
"Recall of music": 0.125,
```

▼ Sample 4

- Pretrained Model

```
"F1 of music": 0.004854368932038835,  
"Precision of music": 0.0026041666666666665,  
"Recall of music": 0.03571428571428571,
```

- Finetune Model

```
"F1 of music": 0.014084507042253521,  
"Precision of music": 0.008771929824561403,  
"Recall of music": 0.03571428571428571,
```

▼ Sample 5

- Pretrained Model

```
"F1 of music": 0.08695652173913045,  
"Precision of music": 0.05555555555555555,  
"Recall of music": 0.2,
```

- Finetune Model

```
"F1 of music": 0.15384615384615385,  
"Precision of music": 0.125,  
"Recall of music": 0.2,
```

▼ Sample 6

- Pretrained Model

```
"F1 of music": 0.03296703296703297,  
"Precision of music": 0.02459016393442623,  
"Recall of music": 0.05,
```

- Finetune Model

```
"F1 of music": 0.03131991051454138,  
"Precision of music": 0.01808785529715762,  
"Recall of music": 0.11666666666666667,
```

▼ Sample 7

- Pretrained Model

```
"F1 of music": 0.0,  
"Precision of music": 1.0,  
"Recall of music": 0.0,
```

- Finetune Model

```
"F1 of music": 0.0,  
"Precision of music": 1.0,  
"Recall of music": 0.0,
```

▼ Sample 8

- Pretrained Model

```
"F1 of music": 0.03076923076923077,  
"Precision of music": 0.015873015873015872,  
"Recall of music": 0.5,
```

- Finetune Model

```
"F1 of music": 0.0,  
"Precision of music": 1.0,  
"Recall of music": 0.0,
```

▼ Eval시 사용한 코드

```
collar_based.best_fscore(  
    scores = scores,  
    ground_truth = os.path.splitext(args.file_name)[0],  
    onset_collar=collar, offset_collar=collar,  
    offset_collar_rate=offset_collar_rate,  
    time_decimals=time_decimals,  
    num_jobs=8,  
)
```

- collar = offset_collar_rate = 0.2

- time_decimals = 30
- PSDS
 - Pretrained된 모델 / Finetuning된 모델 모두 0.0 의 결과
 - ▼ Eval시 사용한 코드

```
intersection_based.psd(
    scores = scores,
    ground_truth = os.path.splitext(args.file_name)[0],
    audio_durations= os.path.splitext(args.file_name)[1],
    dtc_threshold=dtc_threshold, gtc_threshold=gtc_threshold,
    cttc_threshold=cttc_threshold,
    alpha_ct=alpha_ct, alpha_st=alpha_st,
    unit_of_time='hour', max_efpr=100.,
)
```

- dtc_threshold = gtc_threshold = 0.7
- cttc_threshold = None
- alpha_ct = 0, alpha_st = 1
- max_efpr = 100