Tesorflow Speech Recognition Challenge

최 윤 철 (ycheol.choi@gmail.com)

Introduction

1초 길이의 오디오 데이터를 12가지 라벨 중 하나로 분류하는 문제

- **Training Data**
  - 라벨링 되어 있는 1초 길이의 오디오 파일 **64,728**개
  - 1분 길이의 background noise (6종류)

- **Test Data**
  - 라벨링 되어 있지 않은 1초 길이의 오디오 파일 **158,538**개

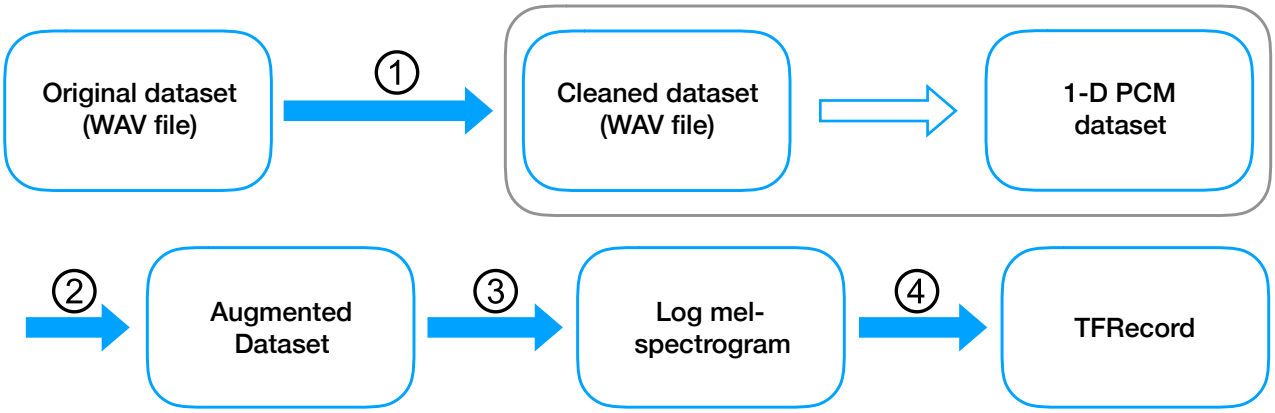
개발환경

- 서버 : AWS EC2 p3.2xlarge
  - OS: Ubuntu 16.04.3
  - GPU: 1 (16G)
  - CPU: 8
  - Memory: 61G,

- 언어 : Python 3.6.3

- 주요 라이브러리 :
  - tensorflow 1.4
  - librosa 0.5.1
  - numpy 1.13.3
  - pandas 0.21.0

Data Preprocessing



① Removing Abnormal Data

- Training Data에서 라벨과 일치하지 않거나 노이즈인 데이터를 제거
- 데이터를 spectrogram으로 변환한 뒤, 육안으로 데이터 이상 여부를 확인

② Data Augmentation

- 데이터 용량 및 학습시간을 고려해 원본데이터의 8배 크기로 augmentation
- **Speed tuning** : 원본 속도의 0.9 ~ 1.1배 범위에서 랜덤하게 속도 변경
- **Pitch tuning** : 한 옥타브를 24단계로 나누고 원본의 ±4단계 범위에서 랜덤하게 피치를 변경
- **Background noise mixing** : 6종류의 background noise 중에서 하나를 임의로 선택하여 원본 데이터와 합성

③ Making log mel-spectrograms

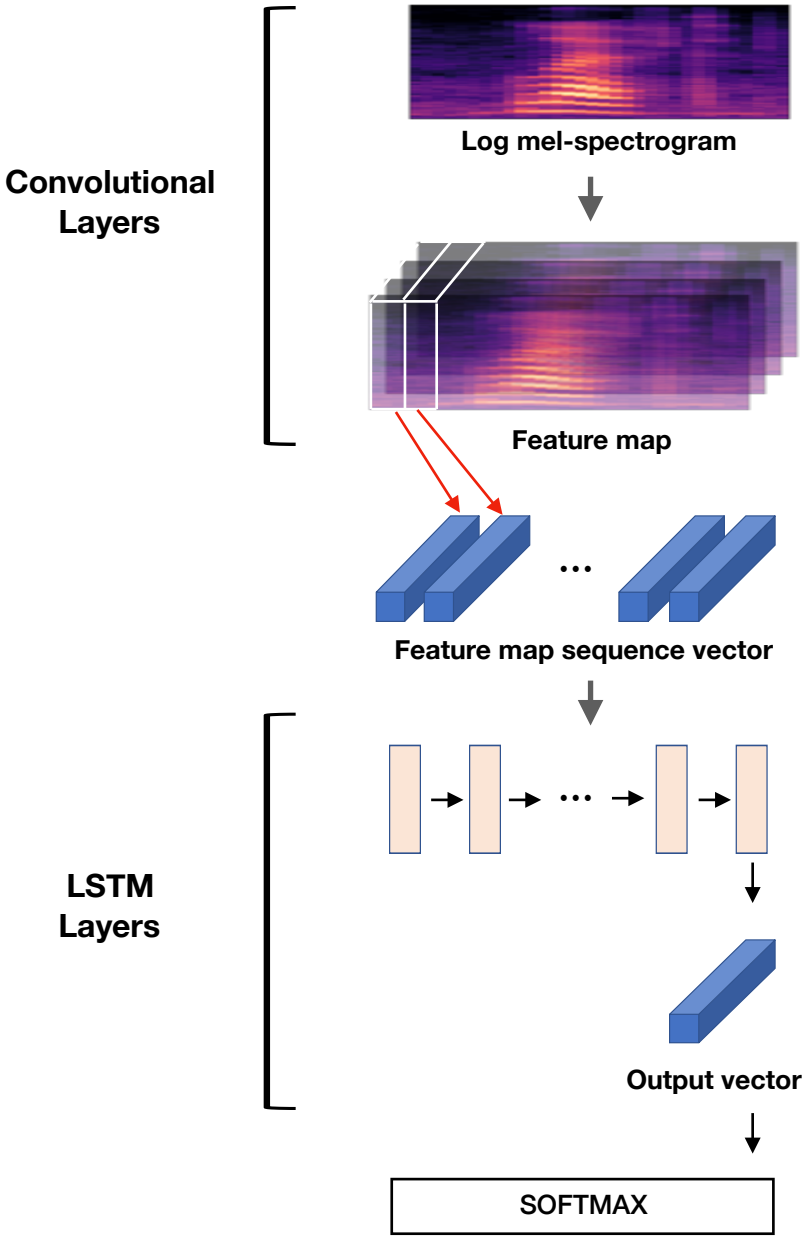
- Augmentation된 데이터셋을 log mel-spectrogram으로 변환
- librosa.feature.melspectrogram 메소드 사용

④ Converting dataset into TFRecord

- 학습시간 단축과 편의를 위해 ndarray 데이터를 TFRecord로 저장

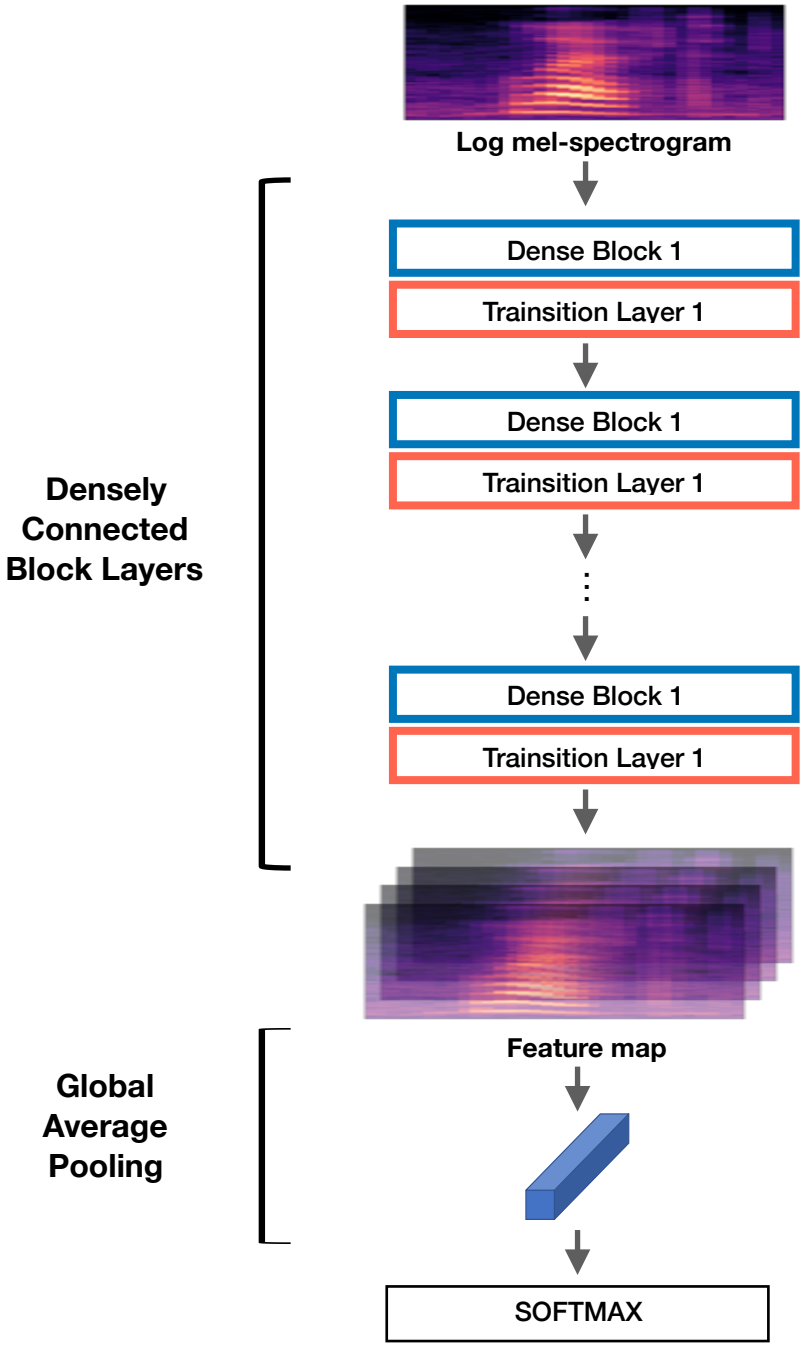
Model1: CNN + LSTM

- Log mel-spectrogram을 convolution layer에 통과시켜 featura map 생성
- 생성된 feature map들을 시간 순서를 갖는 여러 sequence vector로 resize
- sequence vector들을 many-to-one 방식의 LSTM layer에 feed
- LSTM을 통과한 vector를 softmax 함수를 이용하여 분류
- CNN과 LSTM을 결합하여 CNN만으로는 잡아내지 못하는 시계열 특징을 반영



Model2: DenseNet

- 여러 개의 composite layer를 묶어서 하나의 dense block 생성
- 총 4개의 dense block을 구성하고, dense block 사이에 transition layer 구성
- 생성된 feature map들을 global average pooling
- global average pooling으로 생성된 벡터를 softmax 함수로 분류
- 이미지 인식에 있어 성능이 좋은 알고리즘 중 하나로 스펙트럼 인식에도 우수한 결과를 기대



Conclusion

Model	Description	Accuracy
CNN	Convolution Layer 4층, Dense Layer 2층으로 구성된 신경망	0.82767
4Conv + 1LSTM	Convolution Layer 4층, LSTM 1층으로 구성된 신경망	0.87266
6Conv + 2LSTM	Convolution Layer 6층, LSTM 2층으로 구성된 신경망	0.87677
DenseNet(121)	총 4개의 블록, 121층으로 구성된 신경망	0.87231
Bagging	예측한 확률을 단순 평균	0.88546
Minmax-median	예측한 확률이 모두 높거나 낮으면 채택, 아니면 중앙값 채택	0.88793

- 단순 CNN에 LSTM을 결합하여 성능향상을 이뤘음
- DenseNet보다 CNN + LSTM 모델의 성능이 더 좋았음
- 기본 bagging model보다 성능이 더 잘 나오는 minmax-median 앙상블 이용
- 모든 모델들이 강하게 확신하는 부분(0.8이상, 0.2이하)은 가장 높거나 낮은 proba를 채택
- 강하게 확신하지 않는 부분은 중앙값을 proba로 사용

- 최종 스코어 : **0.88793**
- 리더보드 순위 : **132** / 1315