IAML Project#1 Music Genre Classification

Team 5 : 윤 현, 이 은, 한창진

1. Objective

본 프로젝트의 목표는 30초의 노래가 주어졌을 때 Convolutional Neural Network를 이용한 Genre Classification 모델을 만드는 것이다.

2. Dataset

30초 길이의 노래 7198곡으로 label은 8개의 Genre(Hiphop, Pop, Folk, Rock, Experimental, International, Electronic, Instrumental)로 이루어져 있다. Training set은 그 중 6398개를 사용하였고 각 장르별로 799~800개를 사용하였다. Validation set은 그 외 곡들로 각 장르별 99~100개로 이루어져 있다.

3. Feature

Mel-spectrogram Feature와 cqt를 이용하여 학습을 진행하였다. Audio classification task나 tagging task에서 흔하게 쓰이는 feature이다. MFCC에 비해서 time-frequency representation 방법이 더 좋은 성능을 보인다고 다양한 연구에서 제시되었다. 또한 mel-scale spectrogram이 다양한 model에서 안정적으로 우수한 성능을 보이며 일부 task에서는 cqt가 더 좋은 성능을 보인다는 연구[1]와 CQT가 Mel-scale에 비해 pitch를 더 잘 잡아낸다는 연구[2]에 기반하여 각각을 통해 feature extraction이 가능하도록 Mel-spectrogram Feature과 cqt feature 모두를 이용하기로 하였다.

신호 처리 방법은 기존에 제시된 뼈대 코드의 Mel-spectrogram과 cqt feature을 이용하였다.

4. Model Structure

일반적인 CNN서는 first layer에서 하나의 작은 filter size를 쓰는 것이 일반적이다. 그러나 Audio Domain에서 spectrogram에 대해 vertical filter을 사용할 경우 spectral representation을 학

[1] M. Huzaifah(2017), "Comparison of Time-Frequency Representations for Environmental Sound Classification using Convolutional Neural Networks".

[2] Thomas Lidy, Alexander Schinder(2016), "CQT-BASED CONVOLUTIONAL NEURAL NETWORKS FOR AUDIO SCENE CLASSIFICATION".

습하고[3], horizontal filter을 사용할 경우 longer temporal cues를 학습한다는 연구가 진행되었다.[4] 다양한 필터 사이즈를 사용할 경우 더 다양한 feature extraction이 가능하다는 결과 역시 있었다.[5] 그렇기에 본 프로젝트에서는 이러한 연구를 기반으로 mel-spectrogram에 대해 vertical filter와 horizontal filter을 모두 이용하였다.

Mel-spectrogram에 대해선 이 두개의 filter size를 이용하여 vertical filter에 대해선 4개의 convolution layer와 3개의 max-pooling layer로 구성하였다. Horizon filter로는 7개의 convolution layer와 6개의 max-pooling layer로 구성하였다. CQT에서는 CNN에서 일반적으로 사용하는 small size의 kernel을 사용하였고 이 역시 7개의 convolution layer와 6개의 max-pooling layer로 구성하였다.

Convolution layer의 Activation function으로는 ReLU(Rectified Linear Unit)를 사용하였고 각 convolution layer 이후에 Batch normalization을 사용하였다. 또한 ReLU를 위한 weight initialize 방법으로는 tensorflow에서 초기값으로 제공하는 Xavier에 비해 He initialization이 더 괜찮은 성능을 보인다는 연구에 기반해 He initialization을 이용하였다.[6] 이는 Batch Normalization이 제시된 논문에서는 activation 이전에 사용할 것을 명시했지만[7] 최근 실험[8]에 따르면 activation function 이후에 batch normalization을 이용하는 것이 더 우수한 성능을 보인다는 결과와 기존 논문의 저자가 최근 연구에서는 activation 이후에 Normalization을 적용하도록 모델을 작성하고 있다는 주장[9]에 근거했다. Convolution – activation – batch normalization – max-pooling 과정을 거치도록 하였다.

이렇게 3개를 각각 Convolution layer들을 통과시킨 후 이들을 최종적으로 concat 한 후 fully connected layer에 dropout (keep prob = 0.5)을 적용하였다 이 때 softmax로 각 장르별 예측값을 output으로 한다. 아래 그림1은 model의 전체적인 구조를 나타낸다.

---

[3] Honglak lee, Yan largman, Peter Pham, and Andrew Y. Ng(2009), "Unsupervised feature learning for audio classification using convolutional deep belief networks", NIPS, pp.1096-1104.

[4] Jan Schluter, Sebastian Bock(2014). "Improved musical onset detection with convolutional neural networks", IEEE.

[5] Jordi Pons, Oliga Silzovskaia, Rong Gong, and Emilia Gomew(2017), "Timbre Analysis of Music Audio Signals with Convolutional Neural Networks", EUSIPCO.

6 Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun (2015), "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification", ICCV.

7 Sergey Ioffe, Christian Szegedy(2015), "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift".

8 https://github.com/ducha-aiki/caffenet-benchmark/blob/master/batchnorm.md

9 https://github.com/keras-team/keras/issues/1802

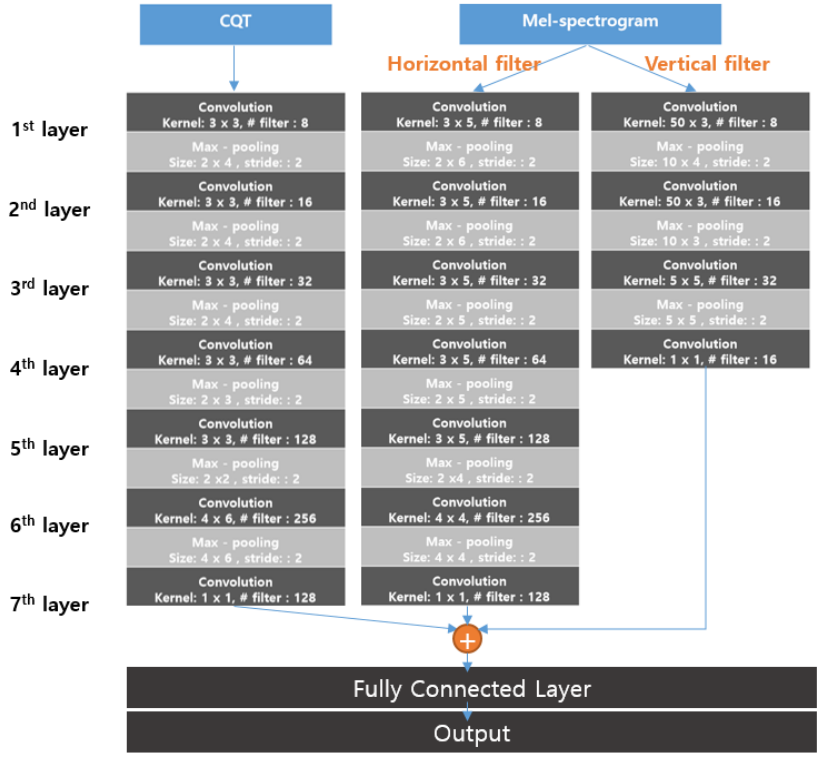

그림 1. 모델 구조

5. Model Training

Optimizer로는 Adam optimizer를 이용하였고 learning rate의 초기값으로 0.001을 사용하였는데 learning rate decay 방식을 적용하여 매 training step 마다 learning rate가 0.3배가 되도록 조정하였다. Epoch은 46, Batch size는 32로 설정하여 training을 진행했다.

6. Result

학습된 모델로 Validation accuracy 가 0.555로 나왔다. 본 모델의 학습 곡선은 아래와 같다.

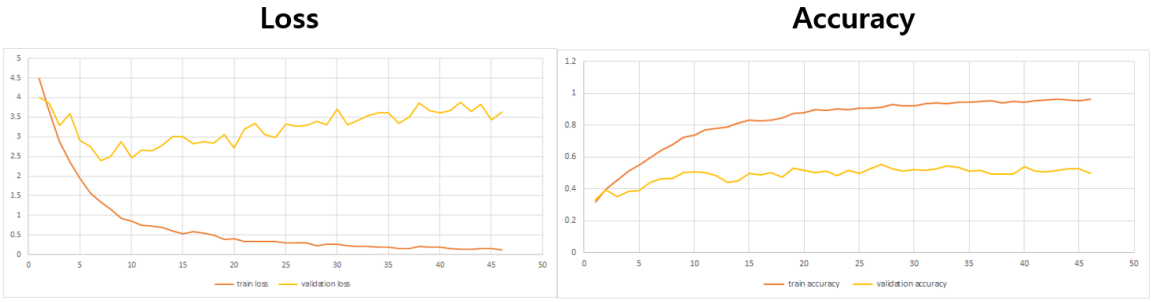

그림 2. 학습 곡선