Snap Point

TEAM JitterBug

18기 오중균
18기 김창영

I. Hyper Parameter

II. Modeling

III. Q&A

Hyper Parameter

Librosa

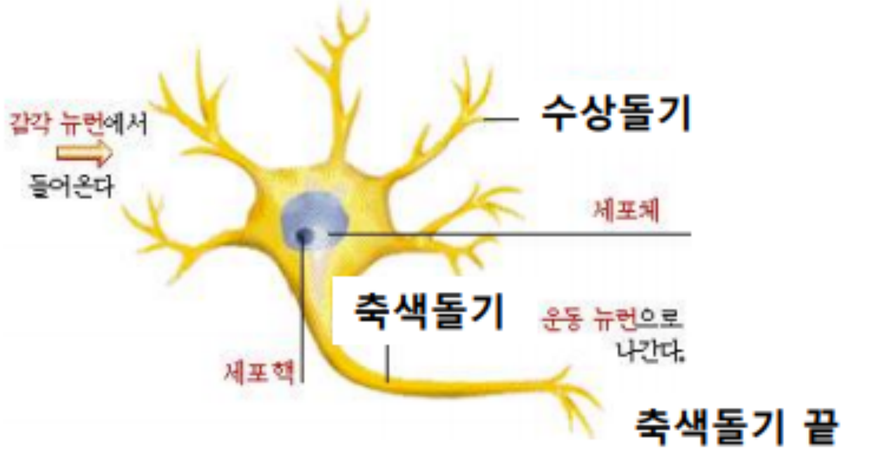
Python Package
for Music and Audio Analysis

```
# Sampling Rate = 20000
# Hop Length = 500 (samples) = 0.025ms
# # of MFCC = 40
# 10 Frames (0.25sec)
# Data Augmentation =
# (+0.1, +0.5, +1.0, -0.1, -0.5, -1.0)
```

원본 음성 데이터 (.wav)

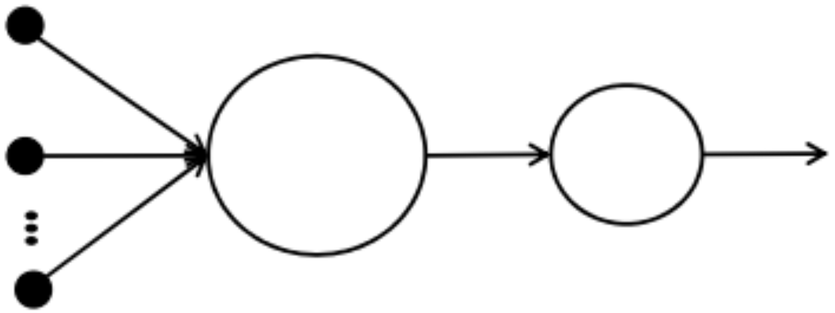
```
y, sr = librosa.load(file, sr = 20000)

y_mfcc = librosa.feature.mfcc(y=y,
                              sr=sr,
                              n_mfcc=40,
                              hop_length=500)

for frame_number in beat_frames:
        lighten(y_mfcc[frame_number-3:frame_number+7])
```

Modeling

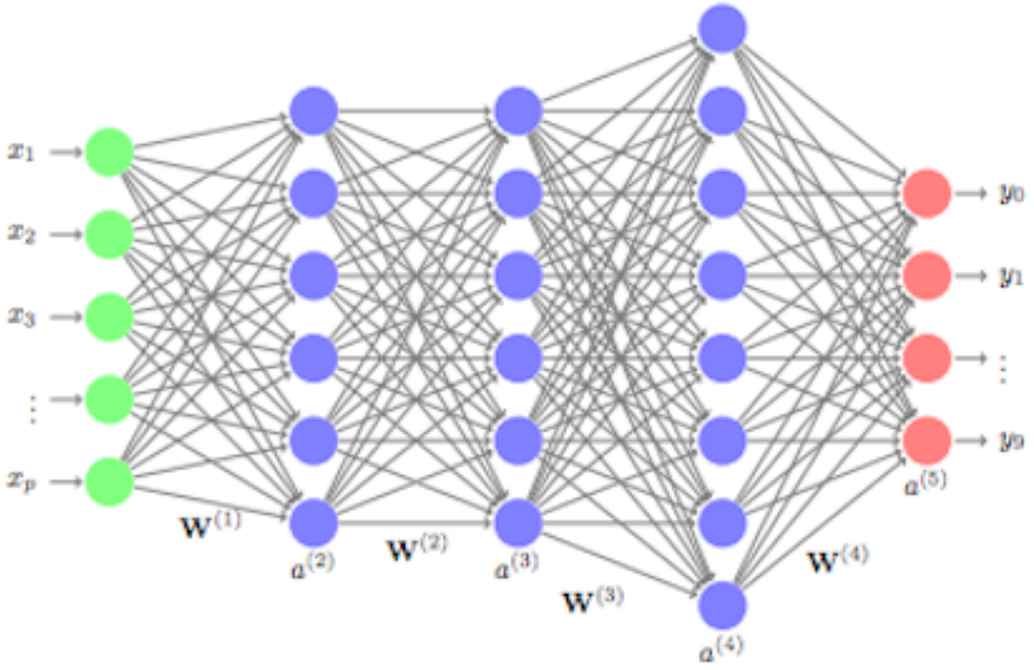


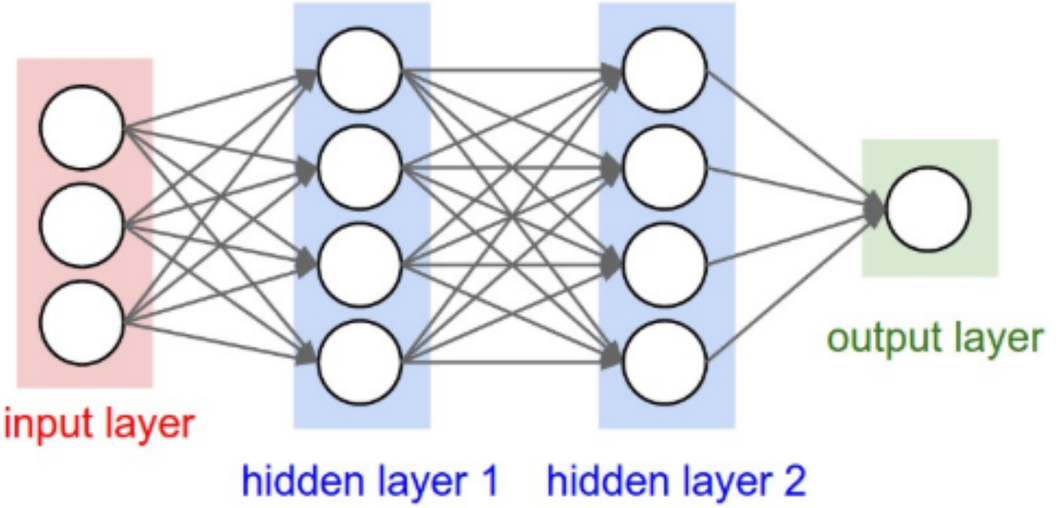
모델링

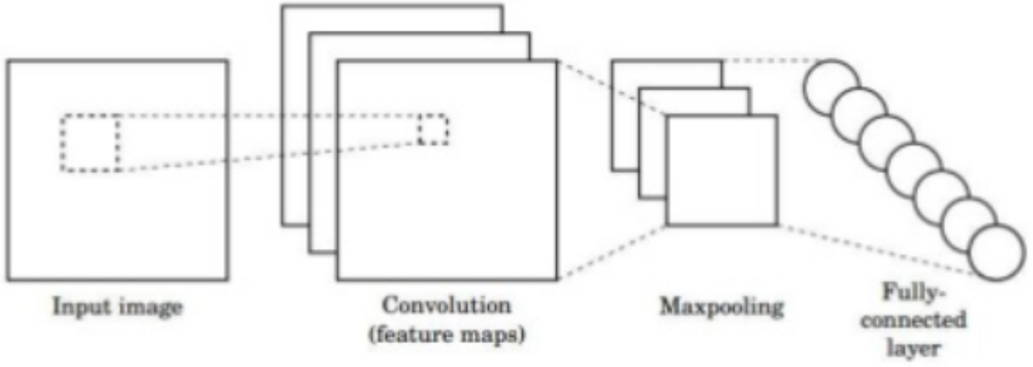
모델; 실제보다 간단하게 만든 것

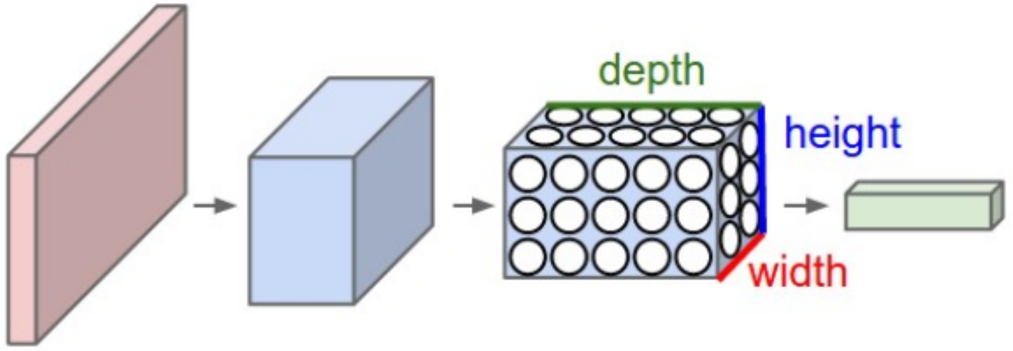


신경세포 1개를 모델링하여 만든 하나의 멋진 그림 모델


x_1
x_2
x_3
x_p
$\mathbf{W}^{(1)}$
$a^{(2)}$
$\mathbf{W}^{(2)}$
$a^{(3)}$
$\mathbf{W}^{(3)}$
$a^{(4)}$
$\mathbf{W}^{(4)}$
$a^{(5)}$
y_0
y_1
y_9


input layer
hidden layer 1
hidden layer 2
output layer


Input image
Convolution
(feature maps)
Maxpooling
Fully-
connected
layer


depth
height
width

```
model = tf.keras.models.Sequential()
model.add(Conv2D(8, kernel_size, padding='same', input_shape=(10,40,1)))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=pool_size))
model.add(Dropout(0.5))

model.add(Conv2D(16, kernel_size, padding='same'))
model.add(Activation('relu'))
model.add(MaxPooling2D(pool_size=pool_size))
model.add(Dropout(0.5))

model.add(Flatten())
model.add(Dense(600))
model.add(Activation('relu'))
model.add(Dropout(0.5))
model.add(Dense(10))
model.add(Activation('softmax'))

model.compile (optimizer= 'adam', loss='binary_crossentropy', metrics = ['accuracy'])
```

```
Epoch 10/12
1513/1513 - 0s - loss: 0.0690 - accuracy: 0.9818 - val_loss: 0.2090 - val_accuracy: 0.9515
Epoch 11/12
1513/1513 - 0s - loss: 0.0527 - accuracy: 0.9851 - val_loss: 0.1925 - val_accuracy: 0.9530
Epoch 12/12
1513/1513 - 0s - loss: 0.0617 - accuracy: 0.9832 - val_loss: 0.1659 - val_accuracy: 0.9578
```

Out[255]:
```
<tensorflow.python.keras.callbacks.History at 0x1aa329a6e88>
```

In [256]:
```
test_loss, test_acc = model.evaluate(test_x, test_y, verbose=2)
print('test_loss = ', test_loss, 'test_acc = ', test_acc)

predictions = model.predict(test_x)
```

```
335/1 - 0s - loss: 0.1276 - accuracy: 0.9534
test_loss =  0.19244698857638373 test_acc =  0.9534328
```

Ver 1.0 (Snap Point)

핑거스냅 - 1
이외 모든 비트 - 0

Ver 1.1 (Snap & Clap) - Later

박수소리 - 2
핑거스냅 - 1
이외 모든 소리 - 0

Contributors

김서영 - 29	정선우 - 363	남수민 - 79	김종근 - 75
오중균 - 149	신재욱 - 3	김창영 - 286	양승우 - 15
김정인 - 14	이승학 - 111	박상원 - 53	김양하 - 92

Q&A

Thank You