

An Introduction to Deep Learning With Python

[7.2] Inspecting and monitoring deep-learning models using Keras callbacks and TensorBoard

Prof. Yuzo Iano

pgs: 249 - 252

The ModelCheckpoint and EarlyStopping Callbacks

```
In [ ]: import keras

callbacks_list = [keras.callbacks.EarlyStopping(monitor='acc',
                                                patience=1, ),
                  keras.callbacks.ModelCheckpoint(filepath='my_model.h5',
                                                  monitor='val_loss',
                                                  save_best_only=True)]

model.compile(optimizer='rmsprop',
              loss='binary_crossentropy',
              metrics=['acc'])

model.fit(x, y,
          epochs=10,
          batch_size=32,
          callbacks=callbacks_list,
          validation_data=(x_val, y_val))
```

The ReduceLROnPlateau Callback

```
In [ ]: callbacks_list = [keras.callbacks.ReduceLROnPlateau(monitor='val_loss',
                                                            factor=0.1,
                                                            patience=10)]

model.fit(x, y,
          epochs=10,
          batch_size=32,
          callbacks=callbacks_list,
          validation_data=(x_val, y_val))
```

Writing your own callback

```
In [1]: import keras
import numpy as np

class ActivationLogger(keras.callbacks.Callback):
    def set_model(self, model):
        self.model = model
        layer_outputs = [layer.output for layer in model.layers]
        self.activations_model = keras.models.Model(model.input, layer_outputs)

    def on_epoch_end(self, epoch, logs=None):
        if self.validation_data is None:
            raise RuntimeError('Requires validation_data.')
        validation_sample = self.validation_data[0][0:1]
        activations = self.activations_model.predict(validation_sample)
        f = open('activations_at_epoch_' + str(epoch) + '.npz', 'w')
        np.savez(f, activations)
        f.close()
```

Using TensorFlow backend.

Pablo Minango

- pablodavid218@gmail.com