tf.keras.callbacks.TensorBoard

Contents

Class `TensorBoard`

Inherits From: `Callback`

Defined in `tensorflow/python/keras/_impl/keras/callbacks.py`.

Tensorboard basic visualizations.

This callback writes a log for TensorBoard, which allows you to visualize dynamic graphs of your training and test metrics, as well as activation histograms for the different layers in your model.

TensorBoard is a visualization tool provided with TensorFlow.

If you have installed TensorFlow with pip, you should be able to launch TensorBoard from the command line:

```
tensorboard --logdir=/full_path_to_your_logs
```

You can find more information about TensorBoard here.

Arguments:

- `log_dir`: the path of the directory where to save the log files to be parsed by TensorBoard.
- `histogram_freq`: frequency (in epochs) at which to compute activation and weight histograms for the layers of the model. If set to 0, histograms won't be computed. Validation data (or split) must be specified for histogram visualizations.
- `write_graph`: whether to visualize the graph in TensorBoard. The log file can become quite large when write_graph is set to True.
- `write_grads`: whether to visualize gradient histograms in TensorBoard. `histogram_freq` must be greater than 0.
- `batch_size`: size of batch of inputs to feed to the network for histograms computation.
- `write_images`: whether to write model weights to visualize as image in TensorBoard.
- `embeddings_freq`: frequency (in epochs) at which selected embedding layers will be saved.
- `embeddings_layer_names`: a list of names of layers to keep eye on. If None or empty list all the embedding layer will be watched.
- `embeddings_metadata`: a dictionary which maps layer name to a file name in which metadata for this embedding layer is saved. See the details about metadata files format. In case if the same metadata file is used for all embedding layers, string can be passed.

Methods

`__init__`

```
__init__(
    log_dir='./logs',
    histogram_freq=0,
    batch_size=32,
    write_graph=True,
    write_grads=False,
    write_images=False
)
```

`on_batch_begin`

```
on_batch_begin(
    batch,
    logs=None
)
```

`on_batch_end`

```
on_batch_end(
    batch,
    logs=None
)
```

`on_epoch_begin`

```
on_epoch_begin(
    epoch,
    logs=None
)
```

`on_epoch_end`

```
on_epoch_end(
    epoch,
    logs=None
)
```

`on_train_begin`

```
on_train_begin(logs=None)
```

`on_train_end`

```
on_train_end(_)
```

`set_model`

```
set_model(model)
```

`set_params`

```
set_params(params)
```

Last updated November 2, 2017.

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | **Privacy**