

Manual

cnn_classification.py

Usage

The CNN tools has the following help text, which can be shown by using

- `python cnn_classification.py --help`

```
usage: CNN-Bug-Report-Classification [-h] [--dataset DATASET] [--pretrained-embedding-path PRETRAINED_EMBEDDING_PATH] [--manual-tuned-model] [--repetitions REPETITIONS] [--generate-hyperparameters]
```

options:

<code>-h, --help</code>	show this help message and exit
<code>--dataset</code>	Dataset to use (e.g. tensorflow/keras) (default: tensorflow)
<code>--pretrained-embedding-path</code>	Path to custom pretrained embeddings (default: None)
<code>--manual-tuned-model</code>	Use a manually tuned module, instead of hyperparameter tuning (default: False)
<code>--repetitions</code>	Specify the number of repetitions (default: 10)
<code>--generate-hyperparameters</code>	Overwrite, and (re)run Hyperband to search for optimal hyperparameters (not recommended) (default: False)

When Hyperband is used (which is by default) the raw results are saved to **results_cnn/hyperband/{dataset}.csv**.

The averages of the metrics are printed and are also saved to **results_cnn/hyperband/averages.csv**.

When the manually tuned model is used (which is achieved by using the **--manual-tuned-model** argument) the raw results are saved to the **results_cnn/manual/{dataset}.csv**.

The averages of the metrics are printed and are also saved to **results_cnn/manual/averages.csv**.

Options

--dataset allows you to use any dataset within the **datasets** directory

To use a dataset (e.g. tensorflow.csv), you specify the name excluding “.csv”

--repetitions allows you to specify how many times the training and evaluation process is repeated, with an average of the metrics taken after.

For example, to generate results for a 5-average using the incubator-mxnet dataset, you would execute the following command.

- **python cnn_classification.py --dataset incubator-mxnet --repetitions 5**

--generate-hyperparameters allows you to (re)run the Hyperband hyperparameter tuning process to (re)generate optimal hyperparameters. This is **not** recommended, as this is time consuming, and has already been run for all the datasets. The hyperparameters are saved to the **/tuners/{dataset}/{dimension}/{hyperband}/** directory.

Defaults

By default,

- `python cnn_classification.py`

Will use the following default parameters

- `--dataset tensorflow`
- `--repetitions 10`

statistical_test.py

Usage

The CNN tools has with the following help text, which can be shown by using

- `python statistical_test.py --help`

```
usage: Statistical Test [-h] [--out-dir OUT_DIR] [--level-of-significance
LEVEL_OF_SIGNIFICANCE] x_dir y_dir
```

positional arguments:

x_dir	First results directory
y_dir	Second results directory

options:

-h, --help	show this help message and exit
--out-dir OUT_DIR	Directory to save the results (default: statistical_tests)
--level-of-significance	Level of significance (default: 0.05)

The tool accepts two results directories (e.g. **results_baseline** and **results_cnn/hyperband**) and calculates pvalues for each metric, within each matching dataset.

These results are saved to **x_dir_vs_y_dir** directory within the **out_dir** directory (which by default is **statistical_tests**).

Defaults

By default,

- `python statistical_tests.py`

Will use the following default parameters

- `--out-dir statistical_tests`
- `--level-of-significance 0.05`