

tf.train.ProfilerHook

Contents

Class ProfilerHook

Methods

`__init__`

`after_create_session`

Class ProfilerHook

Inherits From: [SessionRunHook](#)

Defined in [tensorflow/python/training/basic_session_run_hooks.py](#).

Captures CPU/GPU profiling information every N steps or seconds.

This produces files called "timeline.json", which are in Chrome Trace format.

For more information see: <https://github.com/catapult-project/catapult/blob/master/tracing/README.md>

Methods

`__init__`

```
__init__(
    save_steps=None,
    save_secs=None,
    output_dir='',
    show_dataflow=True,
    show_memory=False
)
```

Initializes a hook that takes periodic profiling snapshots.

`options.run_metadata` argument of `tf.Session.Run` is used to collect metadata about execution. This hook sets the metadata and dumps it in Chrome Trace format.

Args:

- `save_steps`: `int`, save profile traces every N steps. Exactly one of `save_secs` and `save_steps` should be set.
- `save_secs`: `int` or `float`, save profile traces every N seconds.
- `output_dir`: `string`, the directory to save the profile traces to. Defaults to the current directory.
- `show_dataflow`: `bool`, if True, add flow events to the trace connecting producers and consumers of tensors.
- `show_memory`: `bool`, if True, add object snapshot events to the trace showing the sizes and lifetimes of tensors.

after_create_session

```
after_create_session(  
    session,  
    coord  
)
```

Called when new TensorFlow session is created.

This is called to signal the hooks that a new session has been created. This has two essential differences with the situation in which `begin` is called:

- When this is called, the graph is finalized and ops can no longer be added to the graph.
- This method will also be called as a result of recovering a wrapped session, not only at the beginning of the overall session.

Args:

- `session`: A TensorFlow Session that has been created.
- `coord`: A Coordinator object which keeps track of all threads.

after_run

```
after_run(  
    run_context,  
    run_values  
)
```

before_run

```
before_run(run_context)
```

begin

```
begin()
```

end

```
end(session)
```

Called at the end of session.

The `session` argument can be used in case the hook wants to run final ops, such as saving a last checkpoint.

If `session.run()` raises exception other than `OutOfRangeError` or `StopIteration` then `end()` is not called. Note the difference between `end()` and `after_run()` behavior when `session.run()` raises `OutOfRangeError` or `StopIteration`. In that case `end()` is called but `after_run()` is not called.

Args:

- `session`: A TensorFlow Session that will be soon closed.
-

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated November 2, 2017.

Stay Connected

[Blog](#)

[GitHub](#)

[Twitter](#)

Support

[Issue Tracker](#)

[Release Notes](#)

[Stack Overflow](#)

English

[Terms](#) | [Privacy](#)