

tf.saved_model.builder.SavedModelBuilder

Contents

Class `SavedModelBuilder`

Methods

`__init__`

`add_meta_graph`

`add_meta_graph_and_variables`

`save`

Class `SavedModelBuilder`

Defined in [tensorflow/python/saved_model/builder_impl.py](#).

Builds the `SavedModel` protocol buffer and saves variables and assets.

The `SavedModelBuilder` class provides functionality to build a `SavedModel` protocol buffer. Specifically, this allows multiple meta graphs to be saved as part of a single language-neutral `SavedModel`, while sharing variables and assets.

To build a `SavedModel`, the first meta graph must be saved with variables. Subsequent meta graphs will simply be saved with their graph definitions. If assets need to be saved and written or copied to disk, they can be provided when the meta graph def is added. If multiple meta graph defs are associated an asset of the same name, only the first version is retained.

Each meta graph added to the `SavedModel` must be annotated with tags. The tags provide a means to identify the specific meta graph to load and restore, along with the shared set of variables and assets.

Typical usage for the `SavedModelBuilder`:

```
...
builder = tf.saved_model.builder.SavedModelBuilder(export_dir)

with tf.Session(graph=tf.Graph()) as sess:
    ...
    builder.add_meta_graph_and_variables(sess,
                                        ["foo-tag"],
                                        signature_def_map=foo_signatures,
                                        assets_collection=foo_assets)
    ...

with tf.Session(graph=tf.Graph()) as sess:
    ...
    builder.add_meta_graph(["bar-tag", "baz-tag"])
    ...

builder.save()
```

Methods

`__init__`

```
__init__(export_dir)
```

add_meta_graph

```
add_meta_graph(  
    tags,  
    signature_def_map=None,  
    assets_collection=None,  
    legacy_init_op=None,  
    clear_devices=False,  
    main_op=None  
)
```

Adds the current meta graph to the SavedModel.

Creates a Saver in the current scope and uses the Saver to export the meta graph def. Invoking this API requires the `add_meta_graph_and_variables()` API to have been invoked before.

Args:

- `tags` : The set of tags to annotate the meta graph def with.
- `signature_def_map` : The map of signature defs to be added to the meta graph def.
- `assets_collection` : Assets collection to be saved with SavedModel. Note that this collection should be a subset of the assets saved as part of the first meta graph in the SavedModel.
- `legacy_init_op` : Legacy support for op or group of ops to execute after the restore op upon a load.
- `clear_devices` : Set to true if the device info on the default graph should be cleared.
- `main_op` : Op or group of ops to execute when the graph is loaded. Note that when the main_op is specified it is run after the restore op at load-time.

Raises:

- `AssertionError` : If the variables for the SavedModel have not been saved yet, or if the graph already contains one or more legacy init ops.

add_meta_graph_and_variables

```
add_meta_graph_and_variables(  
    sess,  
    tags,  
    signature_def_map=None,  
    assets_collection=None,  
    legacy_init_op=None,  
    clear_devices=False,  
    main_op=None  
)
```

Adds the current meta graph to the SavedModel and saves variables.

Creates a Saver to save the variables from the provided session. Exports the corresponding meta graph def. This function assumes that the variables to be saved have been initialized. For a given `SavedModelBuilder`, this API must be called exactly once and for the first meta graph to save. For subsequent meta graph defs to be added, the `add_meta_graph()` API must be used.

Args:

- `sess` : The TensorFlow session from which to save the meta graph and variables.
- `tags` : The set of tags with which to save the meta graph.
- `signature_def_map` : The map of signature def map to add to the meta graph def.
- `assets_collection` : Assets collection to be saved with SavedModel.
- `legacy_init_op` : Legacy support for op or group of ops to execute after the restore op upon a load.
- `clear_devices` : Set to true if the device info on the default graph should be cleared.
- `main_op` : Op or group of ops to execute when the graph is loaded. Note that when the `main_op` is specified it is run after the restore op at load-time.

save

```
save(as_text=False)
```

Writes a `SavedModel` protocol buffer to disk.

The function writes the SavedModel protocol buffer to the export directory in serialized format.

Args:

- `as_text` : Writes the SavedModel protocol buffer in text format to disk.

Returns:

The path to which the SavedModel protocol buffer was written.

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](#)