TensorFlow   API r1.4

tf.VariableScope

Class `VariableScope`

Defined in `tensorflow/python/ops/variable_scope.py` .

See the guide: Variables > Sharing Variables

Variable scope object to carry defaults to provide to `get_variable` .

Many of the arguments we need for `get_variable` in a variable store are most easily handled with a context. This object is used for the defaults.

Attributes:

- `name` : name of the current scope, used as prefix in get_variable.

- `initializer` : default initializer passed to get_variable.

- `regularizer` : default regularizer passed to get_variable.

- `reuse` : Boolean, None, or tf.AUTO_REUSE, setting the reuse in get_variable. In Eager mode, this argument is always forced to be tf.AUTO_REUSE.

- `caching_device` : string, callable, or None: the caching device passed to get_variable.

- `partitioner` : callable or `None` : the partitioner passed to `get_variable` .

- `custom_getter` : default custom getter passed to get_variable.

- `name_scope` : The name passed to `tf.name_scope` .

- `dtype` : default type passed to get_variable (defaults to DT_FLOAT).

- `use_resource` : if False, create a normal Variable; if True create an experimental ResourceVariable with well-defined semantics. Defaults to False (will later change to True). In Eager mode, this argument is always forced to be True.

- `constraint` : An optional projection function to be applied to the variable after being updated by an `Optimizer` (e.g. used to implement norm constraints or value constraints for layer weights). The function must take as input the unprojected Tensor representing the value of the variable and return the Tensor for the projected value (which must have the same shape). Constraints are not safe to use when doing asynchronous distributed training.

Properties

`caching_device`

constraint

custom_getter

dtype

initializer

name

original_name_scope

partitioner

regularizer

reuse

use_resource

Methods

`__init__`

```
__init__(
    reuse,
    name='',
    initializer=None,
    regularizer=None,
    caching_device=None,
    partitioner=None,
    custom_getter=None,
    name_scope='',
    dtype=tf.float32,
    use_resource=None,
    constraint=None
)
```

Creates a new VariableScope with the given properties.

`get_collection`

```
get_collection(name)
```

Get this scope's variables.

`get_variable`

```
get_variable(
    var_store,
    name,
    shape=None,
    dtype=None,
    initializer=None,
    regularizer=None,
    reuse=None,
    trainable=True,
    collections=None,
    caching_device=None,
    partitioner=None,
    validate_shape=True,
    use_resource=None,
    custom_getter=None,
    constraint=None
)
```

Gets an existing variable with this name or create a new one.

global_variables

```
global_variables()
```

Get this scope's global variables.

local_variables

```
local_variables()
```

Get this scope's local variables.

reuse_variables

```
reuse_variables()
```

Reuse variables in this scope.

set_caching_device

```
set_caching_device(caching_device)
```

Set caching_device for this scope.

set_custom_getter

```
set_custom_getter(custom_getter)
```

Set custom getter for this scope.

set_dtype

```
set_dtype(dtype)
```

Set data type for this scope.

set_initializer

```
set_initializer(initializer)
```

Set initializer for this scope.

set_partitioner

```
set_partitioner(partitioner)
```

Set partitioner for this scope.

set_regularizer

```
set_regularizer(regularizer)
```

Set regularizer for this scope.

set_use_resource

```
set_use_resource(use_resource)
```

Sets whether to use ResourceVariables for this scope.

trainable_variables

```
trainable_variables()
```

Get this scope's trainable variables.

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow