

# Emotion and identity detection from face images

## Convolutional neural network

In [1]:

```
from scipy import misc
import numpy as np
import matplotlib.cm as cm
import tensorflow as tf
import os, sys, inspect
from datetime import datetime
from matplotlib import pyplot as plt
import matplotlib.image as mpimg
from modulos import utils
from modulos.utils import testResult

from tensorflow.python.framework import ops
from sklearn.metrics.classification import accuracy_score
from sklearn.metrics import precision_recall_fscore_support
import warnings
warnings.filterwarnings("ignore")
%matplotlib inline
```

```
C:\Users\Ricardo\Anaconda3\lib\site-packages\h5py\__init__.py:36: Future
Warning: Conversion of the second argument of issubdtype from `float` to
`np.floating` is deprecated. In future, it will be treated as `np.float6
4 == np.dtype(float).type`.
    from ._conv import register_converters as _register_converters
```

In [2]:

```
def rgb2gray(rgb):
    return np.dot(rgb[...,:3], [0.299, 0.587, 0.114])
```

In [3]:

```
emotion = {0:'anger',
           1:'disgust',
           2:'fear',
           3:'happy',
           4:'sad',
           5:'surprise',
           6:'neutral'}
```

In [4]:

```
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '3'
ops.reset_default_graph()
```

In [5]:

```
sess = tf.InteractiveSession()
```

In [6]:

```
new_saver = tf.train.import_meta_graph('modelo/model.ckpt-900.meta')
new_saver.restore(sess, 'modelo/model.ckpt-900')
tf.get_default_graph().as_graph_def()
```

INFO:tensorflow:Restoring parameters from modelo/model.ckpt-900

Out[6]:

```
node {
    name: "truncated_normal/shape"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_INT32
                tensor_shape {
                    dim {
                        size: 4
                    }
                }
                tensor_content: "\005\000\000\000\005\000\000\000\001\000\000\00
0 \000\000\000"
            }
        }
    }
}
node {
    name: "truncated_normal/mean"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_FLOAT
                tensor_shape {
                }
                float_val: 0.0
            }
        }
    }
}
node {
    name: "truncated_normal/stddev"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "value"
        value {
            tensor {
```

```
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.019999999552965164
    }
}
}
}
node {
    name: "truncated_normal/TruncatedNormal"
    op: "TruncatedNormal"
    input: "truncated_normal/shape"
    attr {
        key: "T"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "seed"
        value {
            i: 0
        }
    }
    attr {
        key: "seed2"
        value {
            i: 0
        }
    }
}
node {
    name: "truncated_normal/mul"
    op: "Mul"
    input: "truncated_normal/TruncatedNormal"
    input: "truncated_normal/stddev"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "truncated_normal"
    op: "Add"
    input: "truncated_normal/mul"
    input: "truncated_normal/mean"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
```

```
node {
  name: "W_conv1"
  op: "VariableV2"
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 5
        }
        dim {
          size: 5
        }
        dim {
          size: 1
        }
        dim {
          size: 32
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
  name: "W_conv1/Assign"
  op: "Assign"
  input: "W_conv1"
  input: "truncated_normal"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
```

```
key: "use_locking"
value {
  b: true
}
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
}
node {
  name: "W_conv1/read"
  op: "Identity"
  input: "W_conv1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
}
node {
  name: "truncated_normal_1/shape"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
            size: 4
          }
        }
        tensor_content: "\003\000\000\000\003\000\000\000 \000\000\000@\000\000\000"
      }
    }
  }
}
node {
  name: "truncated_normal_1/mean"
  op: "Const"
  attr {
    key: "dtype"
```

```
    value {
      type: DT_FLOAT
    }
  }
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_FLOAT
      tensor_shape {
      }
      float_val: 0.0
    }
  }
}
}
node {
  name: "truncated_normal_1/stddev"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.019999999552965164
      }
    }
  }
}
node {
  name: "truncated_normal_1/TruncatedNormal"
  op: "TruncatedNormal"
  input: "truncated_normal_1/shape"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "seed"
    value {
      i: 0
    }
  }
  attr {
    key: "seed2"
```

```
    value {
      i: 0
    }
  }
node {
  name: "truncated_normal_1/mul"
  op: "Mul"
  input: "truncated_normal_1/TruncatedNormal"
  input: "truncated_normal_1/stddev"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "truncated_normal_1"
  op: "Add"
  input: "truncated_normal_1/mul"
  input: "truncated_normal_1/mean"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "W_conv2"
  op: "VariableV2"
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 3
        }
        dim {
          size: 3
        }
        dim {
          size: 32
        }
        dim {
          size: 64
        }
      }
    }
  }
}
```

```
    }
}
attr {
  key: "shared_name"
  value {
    s: ""
  }
}
node {
  name: "W_conv2/Assign"
  op: "Assign"
  input: "W_conv2"
  input: "truncated_normal_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "W_conv2/read"
  op: "Identity"
  input: "W_conv2"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
}
node {
```

```
name: "truncated_normal_2/shape"
op: "Const"
attr {
  key: "dtype"
  value {
    type: DT_INT32
  }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_INT32
      tensor_shape {
        dim {
          size: 2
        }
      }
      tensor_content: "\000$ \000\000\000\001\000\000"
    }
  }
}
node {
  name: "truncated_normal_2/mean"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.0
      }
    }
  }
}
node {
  name: "truncated_normal_2/stddev"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.01999999552965164
      }
    }
  }
}
```

```
        }
    }
}
node {
    name: "truncated_normal_2/TruncatedNormal"
    op: "TruncatedNormal"
    input: "truncated_normal_2/shape"
    attr {
        key: "T"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "seed"
        value {
            i: 0
        }
    }
    attr {
        key: "seed2"
        value {
            i: 0
        }
    }
}
node {
    name: "truncated_normal_2/mul"
    op: "Mul"
    input: "truncated_normal_2/TruncatedNormal"
    input: "truncated_normal_2/stddev"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "truncated_normal_2"
    op: "Add"
    input: "truncated_normal_2/mul"
    input: "truncated_normal_2/mean"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "W_fc1"
    op: "VariableV2"
    attr {
```

```
key: "container"
value {
  s: ""
}
}
attr {
  key: "dtype"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "shape"
  value {
    shape {
      dim {
        size: 9216
      }
      dim {
        size: 256
      }
    }
  }
}
attr {
  key: "shared_name"
  value {
    s: ""
  }
}
}
node {
  name: "W_fc1/Assign"
  op: "Assign"
  input: "W_fc1"
  input: "truncated_normal_2"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
```

```
        }
    }
node {
    name: "W_fc1/read"
    op: "Identity"
    input: "W_fc1"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@W_fc1"
            }
        }
    }
}
node {
    name: "truncated_normal_3/shape"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_INT32
                tensor_shape {
                    dim {
                        size: 2
                    }
                }
                tensor_content: "\000\001\000\000\007\000\000\000"
            }
        }
    }
}
node {
    name: "truncated_normal_3/mean"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_FLOAT
                tensor_shape {
                }
            }
        }
    }
}
```

```
        float_val: 0.0
    }
}
}
node {
    name: "truncated_normal_3/stddev"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_FLOAT
                tensor_shape {
                }
                float_val: 0.019999999552965164
            }
        }
    }
}
node {
    name: "truncated_normal_3/TruncatedNormal"
    op: "TruncatedNormal"
    input: "truncated_normal_3/shape"
    attr {
        key: "T"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "seed"
        value {
            i: 0
        }
    }
    attr {
        key: "seed2"
        value {
            i: 0
        }
    }
}
node {
    name: "truncated_normal_3/mul"
    op: "Mul"
    input: "truncated_normal_3/TruncatedNormal"
    input: "truncated_normal_3/stddev"
    attr {
```

```
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
node {
  name: "truncated_normal_3"
  op: "Add"
  input: "truncated_normal_3/mul"
  input: "truncated_normal_3/mean"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "W_fc2"
  op: "VariableV2"
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 256
        }
        dim {
          size: 7
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
  name: "W_fc2/Assign"
  op: "Assign"
  input: "W_fc2"
  input: "truncated_normal_3"
  attr {
    key: "T"
    value {
```

```
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_fc2"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "W_fc2/read"
  op: "Identity"
  input: "W_fc2"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
}
node {
  name: "Const"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
          dim {
            size: 32
          }
        }
      }
    }
  }
}
```

```
        }
        float_val: 0.0
    }
}
node {
    name: "b_conv1"
    op: "VariableV2"
    attr {
        key: "container"
        value {
            s: ""
        }
    }
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "shape"
        value {
            shape {
                dim {
                    size: 32
                }
            }
        }
    }
    attr {
        key: "shared_name"
        value {
            s: ""
        }
    }
}
node {
    name: "b_conv1/Assign"
    op: "Assign"
    input: "b_conv1"
    input: "Const"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@b_conv1"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
}
```

```
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "b_conv1/read"
    op: "Identity"
    input: "b_conv1"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@b_conv1"
            }
        }
    }
}
node {
    name: "Const_1"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_FLOAT
                tensor_shape {
                    dim {
                        size: 64
                    }
                }
                float_val: 0.0
            }
        }
    }
}
node {
    name: "b_conv2"
    op: "VariableV2"
    attr {
        key: "container"
        value {
            s: ""
        }
    }
}
```

```
attr {
  key: "dtype"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "shape"
  value {
    shape {
      dim {
        size: 64
      }
    }
  }
}
attr {
  key: "shared_name"
  value {
    s: ""
  }
}
node {
  name: "b_conv2/Assign"
  op: "Assign"
  input: "b_conv2"
  input: "Const_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "b_conv2/read"
  op: "Identity"
  input: "b_conv2"
  attr {
    key: "T"
  }
}
```

```
    value {
      type: DT_FLOAT
    }
  }
attr {
  key: "_class"
  value {
    list {
      s: "loc:@b_conv2"
    }
  }
}
}
node {
  name: "Const_2"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
          dim {
            size: 256
          }
        }
        float_val: 0.0
      }
    }
  }
}
node {
  name: "b_fc1"
  op: "VariableV2"
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 256
        }
      }
    }
  }
}
```

```
        }
        attr {
          key: "shared_name"
          value {
            s: ""
          }
        }
      }
    node {
      name: "b_fc1/Assign"
      op: "Assign"
      input: "b_fc1"
      input: "Const_2"
      attr {
        key: "T"
        value {
          type: DT_FLOAT
        }
      }
      attr {
        key: "_class"
        value {
          list {
            s: "loc:@b_fc1"
          }
        }
      }
      attr {
        key: "use_locking"
        value {
          b: true
        }
      }
      attr {
        key: "validate_shape"
        value {
          b: true
        }
      }
    }
    node {
      name: "b_fc1/read"
      op: "Identity"
      input: "b_fc1"
      attr {
        key: "T"
        value {
          type: DT_FLOAT
        }
      }
      attr {
        key: "_class"
        value {
          list {
            s: "loc:@b_fc1"
          }
        }
      }
    }
    node {
      name: "Const_3"
```

```
op: "Const"
attr {
  key: "dtype"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_FLOAT
      tensor_shape {
        dim {
          size: 7
        }
      }
      float_val: 0.0
    }
  }
}
node {
  name: "b_fc2"
  op: "VariableV2"
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 7
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
  name: "b_fc2/Assign"
  op: "Assign"
  input: "b_fc2"
  input: "Const_3"
  attr {
    key: "T"
  }
}
```

```
    value {
      type: DT_FLOAT
    }
  }
attr {
  key: "_class"
  value {
    list {
      s: "loc:@b_fc2"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
}
node {
  name: "b_fc2/read"
  op: "Identity"
  input: "b_fc2"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc2"
      }
    }
  }
}
node {
  name: "Variable/initial_value"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
        }
        int_val: 0
      }
    }
  }
}
```

```
        }
    }
}
node {
    name: "Variable"
    op: "VariableV2"
    attr {
        key: "container"
        value {
            s: ""
        }
    }
    attr {
        key: "dtype"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "shape"
        value {
            shape {}
        }
    }
    attr {
        key: "shared_name"
        value {
            s: ""
        }
    }
}
node {
    name: "Variable/Assign"
    op: "Assign"
    input: "Variable"
    input: "Variable/initial_value"
    attr {
        key: "T"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "__class"
        value {
            list {
                s: "loc:@Variable"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
        value {

```

```
        b: true
    }
}
node {
  name: "Variable/read"
  op: "Identity"
  input: "Variable"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@Variable"
      }
    }
  }
}
node {
  name: "Placeholder"
  op: "Placeholder"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        unknown_rank: true
      }
    }
  }
}
node {
  name: "input"
  op: "Placeholder"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: -1
        }
        dim {
          size: 48
        }
        dim {

```

```
        size: 48
    }
    dim {
        size: 1
    }
}
}
node {
    name: "Placeholder_1"
    op: "Placeholder"
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "shape"
        value {
            shape {
                dim {
                    size: -1
                }
                dim {
                    size: 7
                }
            }
        }
    }
}
node {
    name: "conv1/W_conv1/tag"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_STRING
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_STRING
                tensor_shape {
                }
                string_val: "conv1/W_conv1"
            }
        }
    }
}
node {
    name: "conv1/W_conv1"
    op: "HistogramSummary"
    input: "conv1/W_conv1/tag"
    input: "W_conv1/read"
    attr {
        key: "T"
        value {
```

```
        type: DT_FLOAT
    }
}
}
node {
    name: "conv1/b_conv1/tag"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_STRING
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_STRING
                tensor_shape {
                }
                string_val: "conv1/b_conv1"
            }
        }
    }
}
node {
    name: "conv1/b_conv1"
    op: "HistogramSummary"
    input: "conv1/b_conv1/tag"
    input: "b_conv1/read"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "conv1/Conv2D"
    op: "Conv2D"
    input: "input"
    input: "W_conv1/read"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "data_format"
        value {
            s: "NHWC"
        }
    }
    attr {
        key: "dilations"
        value {
            list {
                i: 1
                i: 1
                i: 1
            }
        }
    }
}
```

```
        i: 1
    }
}
attr {
    key: "padding"
    value {
        s: "SAME"
    }
}
attr {
    key: "strides"
    value {
        list {
            i: 1
            i: 1
            i: 1
            i: 1
        }
    }
}
attr {
    key: "use_cudnn_on_gpu"
    value {
        b: true
    }
}
}
node {
    name: "conv1/BiasAdd"
    op: "BiasAdd"
    input: "conv1/Conv2D"
    input: "b_conv1/read"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "data_format"
        value {
            s: "NHWC"
        }
    }
}
node {
    name: "conv1/Relu"
    op: "Relu"
    input: "conv1/BiasAdd"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "conv1/MaxPool"
    op: "MaxPool"
    input: "conv1/Relu"
```

```
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "data_format"
  value {
    s: "NHWC"
  }
}
attr {
  key: "ksize"
  value {
    list {
      i: 1
      i: 2
      i: 2
      i: 1
    }
  }
}
attr {
  key: "padding"
  value {
    s: "SAME"
  }
}
attr {
  key: "strides"
  value {
    list {
      i: 1
      i: 2
      i: 2
      i: 1
    }
  }
}
node {
  name: "conv1/L2Loss"
  op: "L2Loss"
  input: "W_conv1/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "conv1/L2Loss_1"
  op: "L2Loss"
  input: "b_conv1/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
        }
    }
node {
    name: "conv2/W_conv2/tag"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_STRING
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_STRING
                tensor_shape {
                }
                string_val: "conv2/W_conv2"
            }
        }
    }
}
node {
    name: "conv2/W_conv2"
    op: "HistogramSummary"
    input: "conv2/W_conv2/tag"
    input: "W_conv2/read"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "conv2/b_conv2/tag"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_STRING
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_STRING
                tensor_shape {
                }
                string_val: "conv2/b_conv2"
            }
        }
    }
}
node {
    name: "conv2/b_conv2"
    op: "HistogramSummary"
    input: "conv2/b_conv2/tag"
    input: "b_conv2/read"
```

```
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
}
node {
  name: "conv2/Conv2D"
  op: "Conv2D"
  input: "conv1/MaxPool"
  input: "W_conv2/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "data_format"
    value {
      s: "NHWC"
    }
  }
  attr {
    key: "dilations"
    value {
      list {
        i: 1
        i: 1
        i: 1
        i: 1
      }
    }
  }
  attr {
    key: "padding"
    value {
      s: "SAME"
    }
  }
  attr {
    key: "strides"
    value {
      list {
        i: 1
        i: 1
        i: 1
        i: 1
      }
    }
  }
  attr {
    key: "use_cudnn_on_gpu"
    value {
      b: true
    }
  }
}
node {
  name: "conv2/BiasAdd"
```

```
op: "BiasAdd"
input: "conv2/Conv2D"
input: "b_conv2/read"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "data_format"
  value {
    s: "NHWC"
  }
}
}
node {
  name: "conv2/Relu"
  op: "Relu"
  input: "conv2/BiasAdd"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "conv2/MaxPool"
  op: "MaxPool"
  input: "conv2/Relu"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "data_format"
    value {
      s: "NHWC"
    }
  }
  attr {
    key: "ksize"
    value {
      list {
        i: 1
        i: 2
        i: 2
        i: 1
      }
    }
  }
  attr {
    key: "padding"
    value {
      s: "SAME"
    }
  }
}
attr {
```

```
key: "strides"
value {
  list {
    i: 1
    i: 2
    i: 2
    i: 1
  }
}
}
node {
  name: "conv2/L2Loss"
  op: "L2Loss"
  input: "W_conv2/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "conv2/L2Loss_1"
  op: "L2Loss"
  input: "b_conv2/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "fc_1/Reshape/shape"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
            size: 2
          }
        }
        tensor_content: "\377\377\377\377\000$\\000\\000"
      }
    }
  }
}
node {
  name: "fc_1/Reshape"
  op: "Reshape"
  input: "conv2/MaxPool"
```

```
input: "fc_1/Reshape/shape"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "Tshape"
  value {
    type: DT_INT32
  }
}
}
node {
  name: "fc_1/W_fc1/tag"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_STRING
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_STRING
        tensor_shape {
        }
        string_val: "fc_1/W_fc1"
      }
    }
  }
}
node {
  name: "fc_1/W_fc1"
  op: "HistogramSummary"
  input: "fc_1/W_fc1/tag"
  input: "W_fc1/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "fc_1/b_fc1/tag"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_STRING
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_STRING
      }
    }
  }
}
```

```
        tensor_shape {
        }
        string_val: "fc_1/b_fc1"
    }
}
node {
    name: "fc_1/b_fc1"
    op: "HistogramSummary"
    input: "fc_1/b_fc1/tag"
    input: "b_fc1/read"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "fc_1/MatMul"
    op: "MatMul"
    input: "fc_1/Reshape"
    input: "W_fc1/read"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "transpose_a"
        value {
            b: false
        }
    }
    attr {
        key: "transpose_b"
        value {
            b: false
        }
    }
}
node {
    name: "fc_1/add"
    op: "Add"
    input: "fc_1/MatMul"
    input: "b_fc1/read"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "fc_1/Relu"
    op: "Relu"
    input: "fc_1/add"
    attr {
        key: "T"
    }
}
```

```
    value {
      type: DT_FLOAT
    }
  }
node {
  name: "fc_1/dropout/keep_prob"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.5
      }
    }
  }
}
node {
  name: "fc_1/dropout/Shape"
  op: "Shape"
  input: "fc_1/Relu"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "fc_1/dropout/random_uniform/min"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.0
      }
    }
  }
}
```

```
        }
    }
}
node {
  name: "fc_1/dropout/random_uniform/max"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 1.0
      }
    }
  }
}
node {
  name: "fc_1/dropout/random_uniform/RandomUniform"
  op: "RandomUniform"
  input: "fc_1/dropout/Shape"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "seed"
    value {
      i: 0
    }
  }
  attr {
    key: "seed2"
    value {
      i: 0
    }
  }
}
node {
  name: "fc_1/dropout/random_uniform/sub"
  op: "Sub"
  input: "fc_1/dropout/random_uniform/max"
  input: "fc_1/dropout/random_uniform/min"
  attr {
    key: "T"
    value {
```

```
    type: DT_FLOAT
  }
}
}
node {
  name: "fc_1/dropout/random_uniform/mul"
  op: "Mul"
  input: "fc_1/dropout/random_uniform/RandomUniform"
  input: "fc_1/dropout/random_uniform/sub"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "fc_1/dropout/random_uniform"
  op: "Add"
  input: "fc_1/dropout/random_uniform/mul"
  input: "fc_1/dropout/random_uniform/min"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "fc_1/dropout/add"
  op: "Add"
  input: "fc_1/dropout/keep_prob"
  input: "fc_1/dropout/random_uniform"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "fc_1/dropout/Floor"
  op: "Floor"
  input: "fc_1/dropout/add"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "fc_1/dropout/div"
  op: "RealDiv"
  input: "fc_1/Relu"
  input: "fc_1/dropout/keep_prob"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
    }
}
node {
  name: "fc_1/dropout/mul"
  op: "Mul"
  input: "fc_1/dropout/div"
  input: "fc_1/dropout/Floor"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "fc_2/W_fc2/tag"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_STRING
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_STRING
        tensor_shape {
        }
        string_val: "fc_2/W_fc2"
      }
    }
  }
}
node {
  name: "fc_2/W_fc2"
  op: "HistogramSummary"
  input: "fc_2/W_fc2/tag"
  input: "W_fc2/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "fc_2/b_fc2/tag"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_STRING
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_STRING
      }
    }
  }
}
```

```
    tensor_shape {
    }
    string_val: "fc_2/b_fc2"
  }
}
node {
  name: "fc_2/b_fc2"
  op: "HistogramSummary"
  input: "fc_2/b_fc2/tag"
  input: "b_fc2/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "fc_2/MatMul"
  op: "MatMul"
  input: "fc_1/dropout/mul"
  input: "W_fc2/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "transpose_a"
    value {
      b: false
    }
  }
  attr {
    key: "transpose_b"
    value {
      b: false
    }
  }
}
node {
  name: "fc_2/add"
  op: "Add"
  input: "fc_2/MatMul"
  input: "b_fc2/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "output"
  op: "Softmax"
  input: "fc_2/add"
  attr {
    key: "T"
  }
}
```

```
    value {
      type: DT_FLOAT
    }
  }
node {
  name: "softmax_cross_entropy_with_logits_sg/labels_stop_gradient"
  op: "StopGradient"
  input: "Placeholder_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Rank"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
        }
        int_val: 2
      }
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Shape"
  op: "Shape"
  input: "fc_2/add"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Rank_1"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
```

```
    }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_INT32
      tensor_shape {
      }
      int_val: 2
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Shape_1"
  op: "Shape"
  input: "fc_2/add"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Sub/y"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
        }
        int_val: 1
      }
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Sub"
  op: "Sub"
  input: "softmax_cross_entropy_with_logits_sg/Rank_1"
  input: "softmax_cross_entropy_with_logits_sg/Sub/y"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
```

```
        }
    }
}
node {
    name: "softmax_cross_entropy_with_logits_sg/Slice/begin"
    op: "Pack"
    input: "softmax_cross_entropy_with_logits_sg/Sub"
    attr {
        key: "N"
        value {
            i: 1
        }
    }
    attr {
        key: "T"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "axis"
        value {
            i: 0
        }
    }
}
node {
    name: "softmax_cross_entropy_with_logits_sg/Slice/size"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_INT32
                tensor_shape {
                    dim {
                        size: 1
                    }
                }
                int_val: 1
            }
        }
    }
}
node {
    name: "softmax_cross_entropy_with_logits_sg/Slice"
    op: "Slice"
    input: "softmax_cross_entropy_with_logits_sg/Shape_1"
    input: "softmax_cross_entropy_with_logits_sg/Slice/begin"
    input: "softmax_cross_entropy_with_logits_sg/Slice/size"
    attr {
        key: "Index"
        value {
            type: DT_INT32
        }
    }
}
```

```
        }
        attr {
          key: "T"
          value {
            type: DT_INT32
          }
        }
      }
    node {
      name: "softmax_cross_entropy_with_logits_sg/concat/values_0"
      op: "Const"
      attr {
        key: "dtype"
        value {
          type: DT_INT32
        }
      }
      attr {
        key: "value"
        value {
          tensor {
            dtype: DT_INT32
            tensor_shape {
              dim {
                size: 1
              }
            }
            int_val: -1
          }
        }
      }
    }
    node {
      name: "softmax_cross_entropy_with_logits_sg/concat/axis"
      op: "Const"
      attr {
        key: "dtype"
        value {
          type: DT_INT32
        }
      }
      attr {
        key: "value"
        value {
          tensor {
            dtype: DT_INT32
            tensor_shape {
            }
            int_val: 0
          }
        }
      }
    }
    node {
      name: "softmax_cross_entropy_with_logits_sg/concat"
      op: "ConcatV2"
      input: "softmax_cross_entropy_with_logits_sg/concat/values_0"
      input: "softmax_cross_entropy_with_logits_sg/Slice"
      input: "softmax_cross_entropy_with_logits_sg/concat/axis"
      attr {
        key: "N"
      }
    }
```

```
    value {
      i: 2
    }
  }
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Reshape"
  op: "Reshape"
  input: "fc_2/add"
  input: "softmax_cross_entropy_with_logits_sg(concat"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Rank_2"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
        }
        int_val: 2
      }
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Shape_2"
  op: "Shape"
  input: "softmax_cross_entropy_with_logits_sg(labels_stop_gradient"
  attr {
```

```
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Sub_1/y"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
        }
        int_val: 1
      }
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Sub_1"
  op: "Sub"
  input: "softmax_cross_entropy_with_logits_sg/Rank_2"
  input: "softmax_cross_entropy_with_logits_sg/Sub_1/y"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Slice_1/begin"
  op: "Pack"
  input: "softmax_cross_entropy_with_logits_sg/Sub_1"
  attr {
    key: "N"
    value {
      i: 1
    }
  }
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
```

```
attr {
  key: "axis"
  value {
    i: 0
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Slice_1/size"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
            size: 1
          }
        }
        int_val: 1
      }
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Slice_1"
  op: "Slice"
  input: "softmax_cross_entropy_with_logits_sg/Shape_2"
  input: "softmax_cross_entropy_with_logits_sg/Slice_1/begin"
  input: "softmax_cross_entropy_with_logits_sg/Slice_1/size"
  attr {
    key: "Index"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/concat_1/values_0"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
  }
}
```

```
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
            size: 1
          }
        }
        int_val: -1
      }
    }
  }
node {
  name: "softmax_cross_entropy_with_logits_sg/concat_1/axis"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
        }
        int_val: 0
      }
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/concat_1"
  op: "ConcatV2"
  input: "softmax_cross_entropy_with_logits_sg/concat_1/values_0"
  input: "softmax_cross_entropy_with_logits_sg/Slice_1"
  input: "softmax_cross_entropy_with_logits_sg/concat_1/axis"
  attr {
    key: "N"
    value {
      i: 2
    }
  }
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Reshape_1"
```

```
op: "Reshape"
input: "softmax_cross_entropy_with_logits_sg/labels_stop_gradient"
input: "softmax_cross_entropy_with_logits_sg(concat_1"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "Tshape"
  value {
    type: DT_INT32
  }
}
}
node {
  name: "softmax_cross_entropy_with_logits_sg"
  op: "SoftmaxCrossEntropyWithLogits"
  input: "softmax_cross_entropy_with_logits_sg/Reshape"
  input: "softmax_cross_entropy_with_logits_sg/Reshape_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Sub_2/y"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
        }
        int_val: 1
      }
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Sub_2"
  op: "Sub"
  input: "softmax_cross_entropy_with_logits_sg/Rank"
  input: "softmax_cross_entropy_with_logits_sg/Sub_2/y"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
```

```
node {
  name: "softmax_cross_entropy_with_logits_sg/Slice_2/begin"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
            size: 1
          }
        }
        int_val: 0
      }
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Slice_2/size"
  op: "Pack"
  input: "softmax_cross_entropy_with_logits_sg/Sub_2"
  attr {
    key: "N"
    value {
      i: 1
    }
  }
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "axis"
    value {
      i: 0
    }
  }
}
node {
  name: "softmax_cross_entropy_with_logits_sg/Slice_2"
  op: "Slice"
  input: "softmax_cross_entropy_with_logits_sg/Shape"
  input: "softmax_cross_entropy_with_logits_sg/Slice_2/begin"
  input: "softmax_cross_entropy_with_logits_sg/Slice_2/size"
  attr {
    key: "Index"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "T"
  }
}
```

```
    value {
      type: DT_INT32
    }
  }
node {
  name: "softmax_cross_entropy_with_logits_sg/Reshape_2"
  op: "Reshape"
  input: "softmax_cross_entropy_with_logits_sg"
  input: "softmax_cross_entropy_with_logits_sg/Slice_2"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "Const_4"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
            size: 1
          }
        }
        int_val: 0
      }
    }
  }
}
node {
  name: "Mean"
  op: "Mean"
  input: "softmax_cross_entropy_with_logits_sg/Reshape_2"
  input: "Const_4"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tidx"
    value {
```

```
    type: DT_INT32
  }
}
attr {
  key: "keep_dims"
  value {
    b: false
  }
}
node {
  name: "Entropy/tags"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_STRING
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_STRING
        tensor_shape {
        }
        string_val: "Entropy"
      }
    }
  }
}
node {
  name: "Entropy"
  op: "ScalarSummary"
  input: "Entropy/tags"
  input: "Mean"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "AddN"
  op: "AddN"
  input: "conv1/L2Loss"
  input: "conv1/L2Loss_1"
  input: "conv2/L2Loss"
  input: "conv2/L2Loss_1"
  attr {
    key: "N"
    value {
      i: 4
    }
  }
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
    }
}
node {
  name: "Reg_loss/tags"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_STRING
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_STRING
        tensor_shape {
        }
        string_val: "Reg_loss"
      }
    }
  }
}
node {
  name: "Reg_loss"
  op: "ScalarSummary"
  input: "Reg_loss/tags"
  input: "AddN"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "mul/x"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.001000000474974513
      }
    }
  }
}
node {
  name: "mul"
  op: "Mul"
  input: "mul/x"
  input: "AddN"
```

```
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
node {
  name: "add"
  op: "Add"
  input: "Mean"
  input: "mul"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/Shape"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
            }
          }
        }
      }
    }
}
node {
  name: "gradients/grad_ys_0"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
          }
        float_val: 1.0
      }
    }
}
```

```
}

node {
  name: "gradients/Fill"
  op: "Fill"
  input: "gradients/Shape"
  input: "gradients/grad_ys_0"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "index_type"
    value {
      type: DT_INT32
    }
  }
}

node {
  name: "gradients/add_grad/tuple/group_deps"
  op: "NoOp"
  input: "^gradients/Fill"
}

node {
  name: "gradients/add_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/Fill"
  input: "^gradients/add_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/Fill"
      }
    }
  }
}

node {
  name: "gradients/add_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/Fill"
  input: "^gradients/add_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/Fill"
      }
    }
  }
}
```

```
        }
    }
}
node {
  name: "gradients/Mean_grad/Reshape/shape"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
            size: 1
          }
        }
        int_val: 1
      }
    }
  }
}
node {
  name: "gradients/Mean_grad/Reshape"
  op: "Reshape"
  input: "gradients/add_grad/tuple/control_dependency"
  input: "gradients/Mean_grad/Reshape/shape"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/Mean_grad/Shape"
  op: "Shape"
  input: "softmax_cross_entropy_with_logits_sg/Reshape_2"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
```

```
}

node {
  name: "gradients/Mean_grad/Tile"
  op: "Tile"
  input: "gradients/Mean_grad/Reshape"
  input: "gradients/Mean_grad/Shape"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tmultiples"
    value {
      type: DT_INT32
    }
  }
}

node {
  name: "gradients/Mean_grad/Shape_1"
  op: "Shape"
  input: "softmax_cross_entropy_with_logits_sg/Reshape_2"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}

node {
  name: "gradients/Mean_grad/Shape_2"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
            {}
          }
        }
      }
    }
  }
}

node {
  name: "gradients/Mean_grad/Const"
  op: "Const"
```

```
attr {
  key: "dtype"
  value {
    type: DT_INT32
  }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_INT32
      tensor_shape {
        dim {
          size: 1
        }
      }
      int_val: 0
    }
  }
}
node {
  name: "gradients/Mean_grad/Prod"
  op: "Prod"
  input: "gradients/Mean_grad/Shape_1"
  input: "gradients/Mean_grad/Const"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "keep_dims"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/Mean_grad/Const_1"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
          dim {
```

```
        size: 1
    }
}
int_val: 0
}
}
}
}
node {
name: "gradients/Mean_grad/Prod_1"
op: "Prod"
input: "gradients/Mean_grad/Shape_2"
input: "gradients/Mean_grad/Const_1"
attr {
key: "T"
value {
type: DT_INT32
}
}
attr {
key: "Tidx"
value {
type: DT_INT32
}
}
attr {
key: "keep_dims"
value {
b: false
}
}
}
node {
name: "gradients/Mean_grad/Maximum/y"
op: "Const"
attr {
key: "dtype"
value {
type: DT_INT32
}
}
attr {
key: "value"
value {
tensor {
dtype: DT_INT32
tensor_shape {
}
int_val: 1
}
}
}
}
node {
name: "gradients/Mean_grad/Maximum"
op: "Maximum"
input: "gradients/Mean_grad/Prod_1"
input: "gradients/Mean_grad/Maximum/y"
attr {
key: "T"
value {
```

```
    type: DT_INT32
  }
}
}
node {
  name: "gradients/Mean_grad/floordiv"
  op: "FloorDiv"
  input: "gradients/Mean_grad/Prod"
  input: "gradients/Mean_grad/Maximum"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/Mean_grad/Cast"
  op: "Cast"
  input: "gradients/Mean_grad/floordiv"
  attr {
    key: "DstT"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "SrcT"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "Truncate"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/Mean_grad/truediv"
  op: "RealDiv"
  input: "gradients/Mean_grad/Tile"
  input: "gradients/Mean_grad/Cast"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/mul_grad/Mul"
  op: "Mul"
  input: "gradients/add_grad/tuple/control_dependency_1"
  input: "AddN"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
    }
}
node {
  name: "gradients/mul_grad/Mul_1"
  op: "Mul"
  input: "gradients/add_grad/tuple/control_dependency_1"
  input: "mul/x"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/mul_grad/tuple/group_deps"
  op: "NoOp"
  input: "^gradients/mul_grad/Mul"
  input: "^gradients/mul_grad/Mul_1"
}
node {
  name: "gradients/mul_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/mul_grad/Mul"
  input: "^gradients/mul_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/mul_grad/Mul"
      }
    }
  }
}
node {
  name: "gradients/mul_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/mul_grad/Mul_1"
  input: "^gradients/mul_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/mul_grad/Mul_1"
      }
    }
  }
}
node {
```

```
    name: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_2_grad/S
hape"
    op: "Shape"
    input: "softmax_cross_entropy_with_logits_sg"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "out_type"
        value {
            type: DT_INT32
        }
    }
}
node {
    name: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_2_grad/R
eshape"
    op: "Reshape"
    input: "gradients/Mean_grad/truediv"
    input: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_2_grad/
Shape"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "Tshape"
        value {
            type: DT_INT32
        }
    }
}
node {
    name: "gradients/AddN_grad/tuple/group_deps"
    op: "NoOp"
    input: "^gradients/mul_grad/tuple/control_dependency_1"
}
node {
    name: "gradients/AddN_grad/tuple/control_dependency"
    op: "Identity"
    input: "gradients/mul_grad/tuple/control_dependency_1"
    input: "^gradients/AddN_grad/tuple/group_deps"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "__class"
        value {
            list {
                s: "loc:@gradients/mul_grad/Mul_1"
            }
        }
    }
}
```

```
}

node {
  name: "gradients/AddN_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/mul_grad/tuple/control_dependency_1"
  input: "^gradients/AddN_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/mul_grad/Mul_1"
      }
    }
  }
}

node {
  name: "gradients/AddN_grad/tuple/control_dependency_2"
  op: "Identity"
  input: "gradients/mul_grad/tuple/control_dependency_1"
  input: "^gradients/AddN_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/mul_grad/Mul_1"
      }
    }
  }
}

node {
  name: "gradients/AddN_grad/tuple/control_dependency_3"
  op: "Identity"
  input: "gradients/mul_grad/tuple/control_dependency_1"
  input: "^gradients/AddN_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/mul_grad/Mul_1"
      }
    }
  }
}
```

```
node {
  name: "gradients/zeros_like"
  op: "ZerosLike"
  input: "softmax_cross_entropy_with_logits_sg:1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/softmax_cross_entropy_with_logits_sg_grad/ExpandDims/dim"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
        }
        int_val: -1
      }
    }
  }
}
node {
  name: "gradients/softmax_cross_entropy_with_logits_sg_grad/ExpandDims"
  op: "ExpandDims"
  input: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_2_grad/Reshape"
  input: "gradients/softmax_cross_entropy_with_logits_sg_grad/ExpandDim/s/dim"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tdim"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/softmax_cross_entropy_with_logits_sg_grad/mul"
  op: "Mul"
  input: "gradients/softmax_cross_entropy_with_logits_sg_grad/ExpandDim/s"
  input: "softmax_cross_entropy_with_logits_sg:1"
  attr {
    key: "T"
  }
}
```

```
    value {
      type: DT_FLOAT
    }
  }
node {
  name: "gradients/softmax_cross_entropy_with_logits_sg_grad/LogSoftmax"
  op: "LogSoftmax"
  input: "softmax_cross_entropy_with_logits_sg/Reshape"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/softmax_cross_entropy_with_logits_sg_grad/Neg"
  op: "Neg"
  input: "gradients/softmax_cross_entropy_with_logits_sg_grad/LogSoftma
x"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/softmax_cross_entropy_with_logits_sg_grad/ExpandDims_
1/dim"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_INT32
        tensor_shape {
        }
        int_val: -1
      }
    }
  }
}
node {
  name: "gradients/softmax_cross_entropy_with_logits_sg_grad/ExpandDims_
1"
  op: "ExpandDims"
  input: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_2_grad/
Reshape"
  input: "gradients/softmax_cross_entropy_with_logits_sg_grad/ExpandDims
_1/dim"
  attr {
    key: "T"
    value {
```

```

        type: DT_FLOAT
    }
}
attr {
    key: "Tdim"
    value {
        type: DT_INT32
    }
}
node {
    name: "gradients/softmax_cross_entropy_with_logits_sg_grad/mul_1"
    op: "Mul"
    input: "gradients/softmax_cross_entropy_with_logits_sg_grad/ExpandDims
_-1"
    input: "gradients/softmax_cross_entropy_with_logits_sg_grad/Neg"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "gradients/softmax_cross_entropy_with_logits_sg_grad/tuple/group
_deps"
    op: "NoOp"
    input: "^gradients/softmax_cross_entropy_with_logits_sg_grad/mul"
    input: "^gradients/softmax_cross_entropy_with_logits_sg_grad/mul_1"
}
node {
    name: "gradients/softmax_cross_entropy_with_logits_sg_grad/tuple/contr
ol_dependency"
    op: "Identity"
    input: "gradients/softmax_cross_entropy_with_logits_sg_grad/mul"
    input: "gradients/softmax_cross_entropy_with_logits_sg_grad/tuple/gro
up_deps"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@gradients/softmax_cross_entropy_with_logits_sg_grad/mu
1"
            }
        }
    }
}
node {
    name: "gradients/softmax_cross_entropy_with_logits_sg_grad/tuple/contr
ol_dependency_1"
    op: "Identity"
    input: "gradients/softmax_cross_entropy_with_logits_sg_grad/mul_1"
    input: "gradients/softmax_cross_entropy_with_logits_sg_grad/tuple/gro
up_deps"
    attr {

```

```
key: "T"
value {
  type: DT_FLOAT
}
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@gradients/softmax_cross_entropy_with_logits_sg_grad/mul
_1"
    }
  }
}
}
node {
  name: "gradients/conv1/L2Loss_grad/mul"
  op: "Mul"
  input: "W_conv1/read"
  input: "gradients/AddN_grad/tuple/control_dependency"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/conv1/L2Loss_1_grad/mul"
  op: "Mul"
  input: "b_conv1/read"
  input: "gradients/AddN_grad/tuple/control_dependency_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/conv2/L2Loss_grad/mul"
  op: "Mul"
  input: "W_conv2/read"
  input: "gradients/AddN_grad/tuple/control_dependency_2"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/conv2/L2Loss_1_grad/mul"
  op: "Mul"
  input: "b_conv2/read"
  input: "gradients/AddN_grad/tuple/control_dependency_3"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
    }
}
node {
  name: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_grad/Shape"
  op: "Shape"
  input: "fc_2/add"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_grad/Reshape"
  op: "Reshape"
  input: "gradients/softmax_cross_entropy_with_logits_sg_grad/tuple/control_dependency"
  input: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_grad/Shape"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_2/add_grad/Shape"
  op: "Shape"
  input: "fc_2/MatMul"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_2/add_grad/Shape_1"
  op: "Const"
```

```
attr {
  key: "dtype"
  value {
    type: DT_INT32
  }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_INT32
      tensor_shape {
        dim {
          size: 1
        }
      }
      int_val: 7
    }
  }
}
node {
  name: "gradients/fc_2/add_grad/BroadcastGradientArgs"
  op: "BroadcastGradientArgs"
  input: "gradients/fc_2/add_grad/Shape"
  input: "gradients/fc_2/add_grad/Shape_1"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_2/add_grad/Sum"
  op: "Sum"
  input: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_grad/Reshape"
  input: "gradients/fc_2/add_grad/BroadcastGradientArgs"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "keep_dims"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_2/add_grad/Reshape"
  op: "Reshape"
```

```
input: "gradients/fc_2/add_grad/Sum"
input: "gradients/fc_2/add_grad/Shape"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "Tshape"
  value {
    type: DT_INT32
  }
}
}
node {
  name: "gradients/fc_2/add_grad/Sum_1"
  op: "Sum"
  input: "gradients/softmax_cross_entropy_with_logits_sg/Reshape_grad/Reshape"
  input: "gradients/fc_2/add_grad/BroadcastGradientArgs:1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "keep_dims"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_2/add_grad/Reshape_1"
  op: "Reshape"
  input: "gradients/fc_2/add_grad/Sum_1"
  input: "gradients/fc_2/add_grad/Shape_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_2/add_grad/tuple/group_deps"
  op: "NoOp"
```

```
    input: "^gradients/fc_2/add_grad/Reshape"
    input: "^gradients/fc_2/add_grad/Reshape_1"
}
node {
  name: "gradients/fc_2/add_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/fc_2/add_grad/Reshape"
  input: "^gradients/fc_2/add_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/fc_2/add_grad/Reshape"
      }
    }
  }
}
node {
  name: "gradients/fc_2/add_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/fc_2/add_grad/Reshape_1"
  input: "^gradients/fc_2/add_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/fc_2/add_grad/Reshape_1"
      }
    }
  }
}
node {
  name: "gradients/fc_2/MatMul_grad/MatMul"
  op: "MatMul"
  input: "gradients/fc_2/add_grad/tuple/control_dependency"
  input: "W_fc2/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "transpose_a"
    value {
      b: false
    }
  }
}
attr {
```

```
    key: "transpose_b"
    value {
      b: true
    }
  }
node {
  name: "gradients/fc_2/MatMul_grad/MatMul_1"
  op: "MatMul"
  input: "fc_1/dropout/mul"
  input: "gradients/fc_2/add_grad/tuple/control_dependency"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "transpose_a"
    value {
      b: true
    }
  }
  attr {
    key: "transpose_b"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_2/MatMul_grad/tuple/group_deps"
  op: "NoOp"
  input: "^gradients/fc_2/MatMul_grad/MatMul"
  input: "^gradients/fc_2/MatMul_grad/MatMul_1"
}
node {
  name: "gradients/fc_2/MatMul_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/fc_2/MatMul_grad/MatMul"
  input: "gradients/fc_2/MatMul_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/fc_2/MatMul_grad/MatMul"
      }
    }
  }
}
node {
  name: "gradients/fc_2/MatMul_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/fc_2/MatMul_grad/MatMul_1"
  input: "gradients/fc_2/MatMul_grad/tuple/group_deps"
```

```
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@gradients/fc_2/MatMul_grad/MatMul_1"
    }
  }
}
}
node {
  name: "gradients/fc_1/dropout/mul_grad/Shape"
  op: "Shape"
  input: "fc_1/dropout/div"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/dropout/mul_grad/Shape_1"
  op: "Shape"
  input: "fc_1/dropout/Floor"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/dropout/mul_grad/BroadcastGradientArgs"
  op: "BroadcastGradientArgs"
  input: "gradients/fc_1/dropout/mul_grad/Shape"
  input: "gradients/fc_1/dropout/mul_grad/Shape_1"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
```

```
node {
  name: "gradients/fc_1/dropout/mul_grad/Mul"
  op: "Mul"
  input: "gradients/fc_2/MatMul_grad/tuple/control_dependency"
  input: "fc_1/dropout/Floor"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/fc_1/dropout/mul_grad/Sum"
  op: "Sum"
  input: "gradients/fc_1/dropout/mul_grad/Mul"
  input: "gradients/fc_1/dropout/mul_grad/BroadcastGradientArgs"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "keep_dims"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_1/dropout/mul_grad/Reshape"
  op: "Reshape"
  input: "gradients/fc_1/dropout/mul_grad/Sum"
  input: "gradients/fc_1/dropout/mul_grad/Shape"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/dropout/mul_grad/Mul_1"
  op: "Mul"
  input: "fc_1/dropout/div"
  input: "gradients/fc_2/MatMul_grad/tuple/control_dependency"
  attr {
    key: "T"
  }
}
```

```
    value {
      type: DT_FLOAT
    }
  }
node {
  name: "gradients/fc_1/dropout/mul_grad/Sum_1"
  op: "Sum"
  input: "gradients/fc_1/dropout/mul_grad/Mul_1"
  input: "gradients/fc_1/dropout/mul_grad/BroadcastGradientArgs:1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "keep_dims"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_1/dropout/mul_grad/Reshape_1"
  op: "Reshape"
  input: "gradients/fc_1/dropout/mul_grad/Sum_1"
  input: "gradients/fc_1/dropout/mul_grad/Shape_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/dropout/mul_grad/tuple/group_deps"
  op: "NoOp"
  input: "^gradients/fc_1/dropout/mul_grad/Reshape"
  input: "^gradients/fc_1/dropout/mul_grad/Reshape_1"
}
node {
  name: "gradients/fc_1/dropout/mul_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/fc_1/dropout/mul_grad/Reshape"
  input: "^gradients/fc_1/dropout/mul_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
```

```
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@gradients/fc_1/dropout/mul_grad/Reshape"
    }
  }
}
node {
  name: "gradients/fc_1/dropout/mul_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/fc_1/dropout/mul_grad/Reshape_1"
  input: "^gradients/fc_1/dropout/mul_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/fc_1/dropout/mul_grad/Reshape_1"
      }
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/Shape"
  op: "Shape"
  input: "fc_1/Relu"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/Shape_1"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "value"
    value {

```

```
    tensor {
      dtype: DT_INT32
      tensor_shape {
        dim {
        }
      }
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/BroadcastGradientArgs"
  op: "BroadcastGradientArgs"
  input: "gradients/fc_1/dropout/div_grad/Shape"
  input: "gradients/fc_1/dropout/div_grad/Shape_1"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/RealDiv"
  op: "RealDiv"
  input: "gradients/fc_1/dropout/mul_grad/tuple/control_dependency"
  input: "fc_1/dropout/keep_prob"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/Sum"
  op: "Sum"
  input: "gradients/fc_1/dropout/div_grad/RealDiv"
  input: "gradients/fc_1/dropout/div_grad/BroadcastGradientArgs"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "keep_dims"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/Reshape"
  op: "Reshape"
```

```
input: "gradients/fc_1/dropout/div_grad/Sum"
input: "gradients/fc_1/dropout/div_grad/Shape"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "Tshape"
  value {
    type: DT_INT32
  }
}
}
node {
  name: "gradients/fc_1/dropout/div_grad/Neg"
  op: "Neg"
  input: "fc_1/Relu"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/RealDiv_1"
  op: "RealDiv"
  input: "gradients/fc_1/dropout/div_grad/Neg"
  input: "fc_1/dropout/keep_prob"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/RealDiv_2"
  op: "RealDiv"
  input: "gradients/fc_1/dropout/div_grad/RealDiv_1"
  input: "fc_1/dropout/keep_prob"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/mul"
  op: "Mul"
  input: "gradients/fc_1/dropout/mul_grad/tuple/control_dependency"
  input: "gradients/fc_1/dropout/div_grad/RealDiv_2"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
}

node {
  name: "gradients/fc_1/dropout/div_grad/Sum_1"
  op: "Sum"
  input: "gradients/fc_1/dropout/div_grad/mul"
  input: "gradients/fc_1/dropout/div_grad/BroadcastGradientArgs:1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "keep_dims"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/Reshape_1"
  op: "Reshape"
  input: "gradients/fc_1/dropout/div_grad/Sum_1"
  input: "gradients/fc_1/dropout/div_grad/Shape_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/dropout/div_grad/tuple/group_deps"
  op: "NoOp"
  input: "^gradients/fc_1/dropout/div_grad/Reshape"
  input: "^gradients/fc_1/dropout/div_grad/Reshape_1"
}
node {
  name: "gradients/fc_1/dropout/div_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/fc_1/dropout/div_grad/Reshape"
  input: "^gradients/fc_1/dropout/div_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
```

```
key: "_class"
value {
  list {
    s: "loc:@gradients/fc_1/dropout/div_grad/Reshape"
  }
}
}
node {
  name: "gradients/fc_1/dropout/div_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/fc_1/dropout/div_grad/Reshape_1"
  input: "^gradients/fc_1/dropout/div_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/fc_1/dropout/div_grad/Reshape_1"
      }
    }
  }
}
node {
  name: "gradients/fc_1/Relu_grad/ReluGrad"
  op: "ReluGrad"
  input: "gradients/fc_1/dropout/div_grad/tuple/control_dependency"
  input: "fc_1/Relu"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/fc_1/add_grad/Shape"
  op: "Shape"
  input: "fc_1/MatMul"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/add_grad/Shape_1"
  op: "Const"
  attr {
```

```
key: "dtype"
value {
  type: DT_INT32
}
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_INT32
      tensor_shape {
        dim {
          size: 1
        }
      }
      int_val: 256
    }
  }
}
}
node {
  name: "gradients/fc_1/add_grad/BroadcastGradientArgs"
  op: "BroadcastGradientArgs"
  input: "gradients/fc_1/add_grad/Shape"
  input: "gradients/fc_1/add_grad/Shape_1"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/add_grad/Sum"
  op: "Sum"
  input: "gradients/fc_1/Relu_grad/ReluGrad"
  input: "gradients/fc_1/add_grad/BroadcastGradientArgs"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "keep_dims"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_1/add_grad/Reshape"
  op: "Reshape"
  input: "gradients/fc_1/add_grad/Sum"
  input: "gradients/fc_1/add_grad/Shape"
```

```
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "Tshape"
  value {
    type: DT_INT32
  }
}
}
node {
  name: "gradients/fc_1/add_grad/Sum_1"
  op: "Sum"
  input: "gradients/fc_1/Relu_grad/ReluGrad"
  input: "gradients/fc_1/add_grad/BroadcastGradientArgs:1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tidx"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "keep_dims"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_1/add_grad/Reshape_1"
  op: "Reshape"
  input: "gradients/fc_1/add_grad/Sum_1"
  input: "gradients/fc_1/add_grad/Shape_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/add_grad/tuple/group_deps"
  op: "NoOp"
  input: "^gradients/fc_1/add_grad/Reshape"
  input: "^gradients/fc_1/add_grad/Reshape_1"
}
```

```
node {
  name: "gradients/fc_1/add_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/fc_1/add_grad/Reshape"
  input: "^gradients/fc_1/add_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/fc_1/add_grad/Reshape"
      }
    }
  }
}
node {
  name: "gradients/fc_1/add_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/fc_1/add_grad/Reshape_1"
  input: "^gradients/fc_1/add_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/fc_1/add_grad/Reshape_1"
      }
    }
  }
}
node {
  name: "gradients/fc_1/MatMul_grad/MatMul"
  op: "MatMul"
  input: "gradients/fc_1/add_grad/tuple/control_dependency"
  input: "W_fc1/read"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "transpose_a"
    value {
      b: false
    }
  }
  attr {
    key: "transpose_b"
    value {
      b: true
    }
  }
}
```

```
        }
    }
}
node {
  name: "gradients/fc_1/MatMul_grad/MatMul_1"
  op: "MatMul"
  input: "fc_1/Reshape"
  input: "gradients/fc_1/add_grad/tuple/control_dependency"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "transpose_a"
    value {
      b: true
    }
  }
  attr {
    key: "transpose_b"
    value {
      b: false
    }
  }
}
node {
  name: "gradients/fc_1/MatMul_grad/tuple/group_deps"
  op: "NoOp"
  input: "^gradients/fc_1/MatMul_grad/MatMul"
  input: "^gradients/fc_1/MatMul_grad/MatMul_1"
}
node {
  name: "gradients/fc_1/MatMul_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/fc_1/MatMul_grad/MatMul"
  input: "^gradients/fc_1/MatMul_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/fc_1/MatMul_grad/MatMul"
      }
    }
  }
}
node {
  name: "gradients/fc_1/MatMul_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/fc_1/MatMul_grad/MatMul_1"
  input: "^gradients/fc_1/MatMul_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
```

```
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@gradients/fc_1/MatMul_grad/MatMul_1"
    }
  }
}
node {
  name: "gradients/fc_1/Reshape_grad/Shape"
  op: "Shape"
  input: "conv2/MaxPool"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "out_type"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/fc_1/Reshape_grad/Reshape"
  op: "Reshape"
  input: "gradients/fc_1/MatMul_grad/tuple/control_dependency"
  input: "gradients/fc_1/Reshape_grad/Shape"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "Tshape"
    value {
      type: DT_INT32
    }
  }
}
node {
  name: "gradients/conv2/MaxPool_grad/MaxPoolGrad"
  op: "MaxPoolGrad"
  input: "conv2/Relu"
  input: "conv2/MaxPool"
  input: "gradients/fc_1/Reshape_grad/Reshape"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "data_format"
```

```
    value {
      s: "NHW"
    }
  }
attr {
  key: "ksize"
  value {
    list {
      i: 1
      i: 2
      i: 2
      i: 1
    }
  }
}
attr {
  key: "padding"
  value {
    s: "SAME"
  }
}
attr {
  key: "strides"
  value {
    list {
      i: 1
      i: 2
      i: 2
      i: 1
    }
  }
}
}
node {
  name: "gradients/conv2/Relu_grad/ReluGrad"
  op: "ReluGrad"
  input: "gradients/conv2/MaxPool_grad/MaxPoolGrad"
  input: "conv2/Relu"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
node {
  name: "gradients/conv2/BiasAdd_grad/BiasAddGrad"
  op: "BiasAddGrad"
  input: "gradients/conv2/Relu_grad/ReluGrad"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "data_format"
    value {
      s: "NHW"
    }
  }
}
```

```
}

node {
  name: "gradients/conv2/BiasAdd_grad/tuple/group_deps"
  op: "NoOp"
  input: "^gradients/conv2/BiasAdd_grad/BiasAddGrad"
  input: "^gradients/conv2/Relu_grad/ReluGrad"
}
node {
  name: "gradients/conv2/BiasAdd_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/conv2/Relu_grad/ReluGrad"
  input: "^gradients/conv2/BiasAdd_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/conv2/Relu_grad/ReluGrad"
      }
    }
  }
}
node {
  name: "gradients/conv2/BiasAdd_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/conv2/BiasAdd_grad/BiasAddGrad"
  input: "^gradients/conv2/BiasAdd_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/conv2/BiasAdd_grad/BiasAddGrad"
      }
    }
  }
}
node {
  name: "gradients/conv2/Conv2D_grad/ShapeN"
  op: "ShapeN"
  input: "conv1/MaxPool"
  input: "W_conv2/read"
  attr {
    key: "N"
    value {
      i: 2
    }
  }
  attr {
    key: "T"
    value {

```

```
    type: DT_FLOAT
  }
}
attr {
  key: "out_type"
  value {
    type: DT_INT32
  }
}
node {
  name: "gradients/conv2/Conv2D_grad/Conv2DBackpropInput"
  op: "Conv2DBackpropInput"
  input: "gradients/conv2/Conv2D_grad/ShapeN"
  input: "W_conv2/read"
  input: "gradients/conv2/BiasAdd_grad/tuple/control_dependency"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "data_format"
    value {
      s: "NHWC"
    }
  }
  attr {
    key: "dilations"
    value {
      list {
        i: 1
        i: 1
        i: 1
        i: 1
      }
    }
  }
  attr {
    key: "padding"
    value {
      s: "SAME"
    }
  }
  attr {
    key: "strides"
    value {
      list {
        i: 1
        i: 1
        i: 1
        i: 1
      }
    }
  }
  attr {
    key: "use_cudnn_on_gpu"
    value {
      b: true
    }
  }
}
```

```
        }
    }
node {
    name: "gradients/conv2/Conv2D_grad/Conv2DBackpropFilter"
    op: "Conv2DBackpropFilter"
    input: "conv1/MaxPool"
    input: "gradients/conv2/Conv2D_grad/ShapeN:1"
    input: "gradients/conv2/BiasAdd_grad/tuple/control_dependency"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "data_format"
        value {
            s: "NHWC"
        }
    }
    attr {
        key: "dilations"
        value {
            list {
                i: 1
                i: 1
                i: 1
                i: 1
            }
        }
    }
    attr {
        key: "padding"
        value {
            s: "SAME"
        }
    }
    attr {
        key: "strides"
        value {
            list {
                i: 1
                i: 1
                i: 1
                i: 1
            }
        }
    }
    attr {
        key: "use_cudnn_on_gpu"
        value {
            b: true
        }
    }
}
node {
    name: "gradients/conv2/Conv2D_grad/tuple/group_deps"
    op: "NoOp"
    input: "^gradients/conv2/Conv2D_grad/Conv2DBackpropFilter"
    input: "^gradients/conv2/Conv2D_grad/Conv2DBackpropInput"
}
```

```
node {
  name: "gradients/conv2/Conv2D_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/conv2/Conv2D_grad/Conv2DBackpropInput"
  input: "^gradients/conv2/Conv2D_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/conv2/Conv2D_grad/Conv2DBackpropInput"
      }
    }
  }
}
node {
  name: "gradients/conv2/Conv2D_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/conv2/Conv2D_grad/Conv2DBackpropFilter"
  input: "^gradients/conv2/Conv2D_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/conv2/Conv2D_grad/Conv2DBackpropFilter"
      }
    }
  }
}
node {
  name: "gradients/AddN"
  op: "AddN"
  input: "gradients/conv2/L2Loss_1_grad/mul"
  input: "gradients/conv2/BiasAdd_grad/tuple/control_dependency_1"
  attr {
    key: "N"
    value {
      i: 2
    }
  }
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
```

```
        s: "loc:@gradients/conv2/L2Loss_1_grad/mul"
    }
}
}
node {
  name: "gradients/conv1/MaxPool_grad/MaxPoolGrad"
  op: "MaxPoolGrad"
  input: "conv1/Relu"
  input: "conv1/MaxPool"
  input: "gradients/conv2/Conv2D_grad/tuple/control_dependency"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "data_format"
    value {
      s: "NHWC"
    }
  }
  attr {
    key: "ksize"
    value {
      list {
        i: 1
        i: 2
        i: 2
        i: 1
      }
    }
  }
  attr {
    key: "padding"
    value {
      s: "SAME"
    }
  }
  attr {
    key: "strides"
    value {
      list {
        i: 1
        i: 2
        i: 2
        i: 1
      }
    }
  }
}
node {
  name: "gradients/AddN_1"
  op: "AddN"
  input: "gradients/conv2/L2Loss_grad/mul"
  input: "gradients/conv2/Conv2D_grad/tuple/control_dependency_1"
  attr {
    key: "N"
    value {
      i: 2
    }
  }
}
```

```
        }
    }
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@gradients/conv2/L2Loss_grad/mul"
            }
        }
    }
}
node {
    name: "gradients/conv1/Relu_grad/ReluGrad"
    op: "ReluGrad"
    input: "gradients/conv1/MaxPool_grad/MaxPoolGrad"
    input: "conv1/Relu"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
node {
    name: "gradients/conv1/BiasAdd_grad/BiasAddGrad"
    op: "BiasAddGrad"
    input: "gradients/conv1/Relu_grad/ReluGrad"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "data_format"
        value {
            s: "NHWC"
        }
    }
}
node {
    name: "gradients/conv1/BiasAdd_grad/tuple/group_deps"
    op: "NoOp"
    input: "^gradients/conv1/BiasAdd_grad/BiasAddGrad"
    input: "^gradients/conv1/Relu_grad/ReluGrad"
}
node {
    name: "gradients/conv1/BiasAdd_grad/tuple/control_dependency"
    op: "Identity"
    input: "gradients/conv1/Relu_grad/ReluGrad"
    input: "^gradients/conv1/BiasAdd_grad/tuple/group_deps"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
}
```

```
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@gradients/conv1/Relu_grad/ReluGrad"
            }
        }
    }
}
node {
    name: "gradients/conv1/BiasAdd_grad/tuple/control_dependency_1"
    op: "Identity"
    input: "gradients/conv1/BiasAdd_grad/BiasAddGrad"
    input: "^gradients/conv1/BiasAdd_grad/tuple/group_deps"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@gradients/conv1/BiasAdd_grad/BiasAddGrad"
            }
        }
    }
}
node {
    name: "gradients/conv1/Conv2D_grad/ShapeN"
    op: "ShapeN"
    input: "input"
    input: "W_conv1/read"
    attr {
        key: "N"
        value {
            i: 2
        }
    }
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "out_type"
        value {
            type: DT_INT32
        }
    }
}
node {
    name: "gradients/conv1/Conv2D_grad/Conv2DBackpropInput"
    op: "Conv2DBackpropInput"
    input: "gradients/conv1/Conv2D_grad/ShapeN"
    input: "W_conv1/read"
    input: "gradients/conv1/BiasAdd_grad/tuple/control_dependency"
```

```
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "data_format"
  value {
    s: "NHWC"
  }
}
attr {
  key: "dilations"
  value {
    list {
      i: 1
      i: 1
      i: 1
      i: 1
    }
  }
}
attr {
  key: "padding"
  value {
    s: "SAME"
  }
}
attr {
  key: "strides"
  value {
    list {
      i: 1
      i: 1
      i: 1
      i: 1
    }
  }
}
attr {
  key: "use_cudnn_on_gpu"
  value {
    b: true
  }
}
node {
  name: "gradients/conv1/Conv2D_grad/Conv2DBackpropFilter"
  op: "Conv2DBackpropFilter"
  input: "input"
  input: "gradients/conv1/Conv2D_grad/ShapeN:1"
  input: "gradients/conv1/BiasAdd_grad/tuple/control_dependency"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "data_format"
```

```
    value {
      s: "NHWC"
    }
  }
attr {
  key: "dilations"
  value {
    list {
      i: 1
      i: 1
      i: 1
      i: 1
    }
  }
}
attr {
  key: "padding"
  value {
    s: "SAME"
  }
}
attr {
  key: "strides"
  value {
    list {
      i: 1
      i: 1
      i: 1
      i: 1
    }
  }
}
attr {
  key: "use_cudnn_on_gpu"
  value {
    b: true
  }
}
}
node {
  name: "gradients/conv1/Conv2D_grad/tuple/group_deps"
  op: "NoOp"
  input: "^gradients/conv1/Conv2D_grad/Conv2DBackpropFilter"
  input: "^gradients/conv1/Conv2D_grad/Conv2DBackpropInput"
}
node {
  name: "gradients/conv1/Conv2D_grad/tuple/control_dependency"
  op: "Identity"
  input: "gradients/conv1/Conv2D_grad/Conv2DBackpropInput"
  input: "^gradients/conv1/Conv2D_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/conv1/Conv2D_grad/Conv2DBackpropInput"
      }
    }
  }
}
```

```
        }
    }
}
node {
  name: "gradients/conv1/Conv2D_grad/tuple/control_dependency_1"
  op: "Identity"
  input: "gradients/conv1/Conv2D_grad/Conv2DBackpropFilter"
  input: "^gradients/conv1/Conv2D_grad/tuple/group_deps"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/conv1/Conv2D_grad/Conv2DBackpropFilter"
      }
    }
  }
}
node {
  name: "gradients/AddN_2"
  op: "AddN"
  input: "gradients/conv1/L2Loss_1_grad/mul"
  input: "gradients/conv1/BiasAdd_grad/tuple/control_dependency_1"
  attr {
    key: "N"
    value {
      i: 2
    }
  }
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@gradients/conv1/L2Loss_1_grad/mul"
      }
    }
  }
}
node {
  name: "gradients/AddN_3"
  op: "AddN"
  input: "gradients/conv1/L2Loss_grad/mul"
  input: "gradients/conv1/Conv2D_grad/tuple/control_dependency_1"
  attr {
    key: "N"
    value {
      i: 2
    }
  }
}
```

```
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@gradients/conv1/L2Loss_grad/mul"
    }
  }
}
node {
  name: "beta1_power/initial_value"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.8999999761581421
      }
    }
  }
}
node {
  name: "beta1_power"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
}
```

```
attr {
  key: "dtype"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "shape"
  value {
    shape {}
  }
}
attr {
  key: "shared_name"
  value {
    s: ""
  }
}
node {
  name: "beta1_power/Assign"
  op: "Assign"
  input: "beta1_power"
  input: "beta1_power/initial_value"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "beta1_power/read"
  op: "Identity"
  input: "beta1_power"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
        }
        attr {
          key: "_class"
          value {
            list {
              s: "loc:@W_conv1"
            }
          }
        }
      }
    }
  }
node {
  name: "beta2_power/initial_value"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.9990000128746033
      }
    }
  }
}
node {
  name: "beta2_power"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
        }
      attr {
        key: "shape"
        value {
          shape {}
        }
      }
    attr {
      key: "shared_name"
      value {
        s: ""
      }
    }
  }
node {
  name: "beta2_power/Assign"
  op: "Assign"
  input: "beta2_power"
  input: "beta2_power/initial_value"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "beta2_power/read"
  op: "Identity"
  input: "beta2_power"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {

```

```
        s: "loc:@W_conv1"
    }
}
}
node {
  name: "W_conv1/Adam/Initializer/zeros"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
          dim {
            size: 5
          }
          dim {
            size: 5
          }
          dim {
            size: 1
          }
          dim {
            size: 32
          }
        }
        float_val: 0.0
      }
    }
  }
}
node {
  name: "W_conv1/Adam"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
}
```

```
        }
    }
attr {
    key: "dtype"
    value {
        type: DT_FLOAT
    }
}
attr {
    key: "shape"
    value {
        shape {
            dim {
                size: 5
            }
            dim {
                size: 5
            }
            dim {
                size: 1
            }
            dim {
                size: 32
            }
        }
    }
}
attr {
    key: "shared_name"
    value {
        s: ""
    }
}
node {
    name: "W_conv1/Adam/Assign"
    op: "Assign"
    input: "W_conv1/Adam"
    input: "W_conv1/Adam/Initializer/zeros"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@W_conv1"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
```

```
    value {
      b: true
    }
  }
node {
  name: "W_conv1/Adam/read"
  op: "Identity"
  input: "W_conv1/Adam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
}
node {
  name: "W_conv1/Adam_1/Initializer/zeros"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
          dim {
            size: 5
          }
          dim {
            size: 5
          }
          dim {
            size: 1
          }
          dim {
            size: 32
          }
        }
        float_val: 0.0
      }
    }
  }
}
```

```
        }
    }
}
node {
  name: "W_conv1/Adam_1"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 5
        }
        dim {
          size: 5
        }
        dim {
          size: 1
        }
        dim {
          size: 32
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
  name: "W_conv1/Adam_1/Assign"
  op: "Assign"
  input: "W_conv1/Adam_1"
  input: "W_conv1/Adam_1/Initializer/zeros"
  attr {
    key: "T"
    value {
```

```
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_conv1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
}
node {
  name: "W_conv1/Adam_1/read"
  op: "Identity"
  input: "W_conv1/Adam_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
}
node {
  name: "W_conv2/Adam/Initializer/zeros/shape_as_tensor"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
}
attr {
```

```
key: "value"
value {
  tensor {
    dtype: DT_INT32
    tensor_shape {
      dim {
        size: 4
      }
    }
    tensor_content: "\003\000\000\000\003\000\000\000 \000\000\000@\000\000\000"
  }
}
node {
  name: "W_conv2/Adam/Initializer/zeros/Const"
  op: "Const"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.0
      }
    }
  }
}
node {
  name: "W_conv2/Adam/Initializer/zeros"
  op: "Fill"
  input: "W_conv2/Adam/Initializer/zeros/shape_as_tensor"
  input: "W_conv2/Adam/Initializer/zeros/Const"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
}
```

```
    }
}
attr {
  key: "index_type"
  value {
    type: DT_INT32
  }
}
node {
  name: "W_conv2/Adam"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 3
        }
        dim {
          size: 3
        }
        dim {
          size: 32
        }
        dim {
          size: 64
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
  name: "W_conv2/Adam/Assign"
  op: "Assign"
```

```
input: "W_conv2/Adam"
input: "W_conv2/Adam/Initializer/zeros"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "__class"
  value {
    list {
      s: "loc:@W_conv2"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "W_conv2/Adam/read"
  op: "Identity"
  input: "W_conv2/Adam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
}
node {
  name: "W_conv2/Adam_1/Initializer/zeros/shape_as_tensor"
  op: "Const"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
  attr {
    key: "dtype"
```

```
    value {
      type: DT_INT32
    }
  }
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_INT32
      tensor_shape {
        dim {
          size: 4
        }
      }
      tensor_content: "\003\000\000\000\003\000\000\000 \000\000\000@\000\000\000"
    }
  }
}
node {
  name: "W_conv2/Adam_1/Initializer/zeros/Const"
  op: "Const"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.0
      }
    }
  }
}
node {
  name: "W_conv2/Adam_1/Initializer/zeros"
  op: "Fill"
  input: "W_conv2/Adam_1/Initializer/zeros/shape_as_tensor"
  input: "W_conv2/Adam_1/Initializer/zeros/Const"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
```

```
key: "_class"
value {
  list {
    s: "loc:@W_conv2"
  }
}
attr {
  key: "index_type"
  value {
    type: DT_INT32
  }
}
}
node {
  name: "W_conv2/Adam_1"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 3
        }
        dim {
          size: 3
        }
        dim {
          size: 32
        }
        dim {
          size: 64
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
```

```
    }
}
node {
  name: "W_conv2/Adam_1/Assign"
  op: "Assign"
  input: "W_conv2/Adam_1"
  input: "W_conv2/Adam_1/Initializer/zeros"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "W_conv2/Adam_1/read"
  op: "Identity"
  input: "W_conv2/Adam_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
}
node {
  name: "W_fc1/Adam/Initializer/zeros/shape_as_tensor"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
}
```

```
        }
    }
attr {
    key: "dtype"
    value {
        type: DT_INT32
    }
}
attr {
    key: "value"
    value {
        tensor {
            dtype: DT_INT32
            tensor_shape {
                dim {
                    size: 2
                }
            }
            tensor_content: "\000$\000\000\000\001\000\000"
        }
    }
}
node {
    name: "W_fc1/Adam/Initializer/zeros/Const"
    op: "Const"
    attr {
        key: "__class"
        value {
            list {
                s: "loc:@W_fc1"
            }
        }
    }
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_FLOAT
                tensor_shape {
                }
                float_val: 0.0
            }
        }
    }
}
node {
    name: "W_fc1/Adam/Initializer/zeros"
    op: "Fill"
    input: "W_fc1/Adam/Initializer/zeros/shape_as_tensor"
    input: "W_fc1/Adam/Initializer/zeros/Const"
    attr {
        key: "T"
        value {
```

```
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_fc1"
    }
  }
}
attr {
  key: "index_type"
  value {
    type: DT_INT32
  }
}
}
node {
  name: "W_fc1/Adam"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 9216
        }
        dim {
          size: 256
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
```

```
node {
  name: "W_fc1/Adam/Assign"
  op: "Assign"
  input: "W_fc1/Adam"
  input: "W_fc1/Adam/Initializer/zeros"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "W_fc1/Adam/read"
  op: "Identity"
  input: "W_fc1/Adam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
}
node {
  name: "W_fc1/Adam_1/Initializer/zeros/shape_as_tensor"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
}
```

```
        }
      attr {
        key: "dtype"
        value {
          type: DT_INT32
        }
      }
    attr {
      key: "value"
      value {
        tensor {
          dtype: DT_INT32
          tensor_shape {
            dim {
              size: 2
            }
          }
          tensor_content: "\000$\000\000\000\001\000\000"
        }
      }
    }
  }
node {
  name: "W_fc1/Adam_1/Initializer/zeros/Const"
  op: "Const"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.0
      }
    }
  }
}
node {
  name: "W_fc1/Adam_1/Initializer/zeros"
  op: "Fill"
  input: "W_fc1/Adam_1/Initializer/zeros/shape_as_tensor"
  input: "W_fc1/Adam_1/Initializer/zeros/Const"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
        }
        attr {
            key: "_class"
            value {
                list {
                    s: "loc:@W_fc1"
                }
            }
        }
        attr {
            key: "index_type"
            value {
                type: DT_INT32
            }
        }
    }
node {
    name: "W_fc1/Adam_1"
    op: "VariableV2"
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@W_fc1"
            }
        }
    }
    attr {
        key: "container"
        value {
            s: ""
        }
    }
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "shape"
        value {
            shape {
                dim {
                    size: 9216
                }
                dim {
                    size: 256
                }
            }
        }
    }
    attr {
        key: "shared_name"
        value {
            s: ""
        }
    }
}
node {
    name: "W_fc1/Adam_1/Assign"
```

```
op: "Assign"
input: "W_fc1/Adam_1"
input: "W_fc1/Adam_1/Initializer/zeros"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_fc1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "W_fc1/Adam_1/read"
  op: "Identity"
  input: "W_fc1/Adam_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
}
node {
  name: "W_fc2/Adam/Initializer/zeros/shape_as_tensor"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
```

```
key: "dtype"
value {
  type: DT_INT32
}
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_INT32
      tensor_shape {
        dim {
          size: 2
        }
      }
      tensor_content: "\000\001\000\000\007\000\000\000"
    }
  }
}
}
node {
  name: "W_fc2/Adam/Initializer/zeros/Const"
  op: "Const"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.0
      }
    }
  }
}
node {
  name: "W_fc2/Adam/Initializer/zeros"
  op: "Fill"
  input: "W_fc2/Adam/Initializer/zeros/shape_as_tensor"
  input: "W_fc2/Adam/Initializer/zeros/Const"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
```

```
key: "_class"
value {
  list {
    s: "loc:@W_fc2"
  }
}
attr {
  key: "index_type"
  value {
    type: DT_INT32
  }
}
}
node {
  name: "W_fc2/Adam"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 256
        }
        dim {
          size: 7
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
  name: "W_fc2/Adam/Assign"
  op: "Assign"
  input: "W_fc2/Adam"
```

```
input: "W_fc2/Adam/Initializer/zeros"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_fc2"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "W_fc2/Adam/read"
  op: "Identity"
  input: "W_fc2/Adam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
}
node {
  name: "W_fc2/Adam_1/Initializer/zeros/shape_as_tensor"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
    key: "dtype"
    value {
```

```
    type: DT_INT32
  }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_INT32
      tensor_shape {
        dim {
          size: 2
        }
      }
      tensor_content: "\000\001\000\000\007\000\000\000"
    }
  }
}
node {
  name: "W_fc2/Adam_1/Initializer/zeros/Const"
  op: "Const"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.0
      }
    }
  }
}
node {
  name: "W_fc2/Adam_1/Initializer/zeros"
  op: "Fill"
  input: "W_fc2/Adam_1/Initializer/zeros/shape_as_tensor"
  input: "W_fc2/Adam_1/Initializer/zeros/Const"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "__class"
    value {
```

```
    list {
      s: "loc:@W_fc2"
    }
  }
attr {
  key: "index_type"
  value {
    type: DT_INT32
  }
}
}
node {
  name: "W_fc2/Adam_1"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 256
        }
        dim {
          size: 7
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
  name: "W_fc2/Adam_1/Assign"
  op: "Assign"
  input: "W_fc2/Adam_1"
  input: "W_fc2/Adam_1/Initializer/zeros"
  attr {
```

```
key: "T"
value {
  type: DT_FLOAT
}
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_fc2"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
}
node {
  name: "W_fc2/Adam_1/read"
  op: "Identity"
  input: "W_fc2/Adam_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
}
node {
  name: "b_conv1/Adam/Initializer/zeros"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv1"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
        }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_FLOAT
                tensor_shape {
                    dim {
                        size: 32
                    }
                }
                float_val: 0.0
            }
        }
    }
}
node {
    name: "b_conv1/Adam"
    op: "VariableV2"
    attr {
        key: "__class"
        value {
            list {
                s: "loc:@b_conv1"
            }
        }
    }
    attr {
        key: "container"
        value {
            s: ""
        }
    }
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "shape"
        value {
            shape {
                dim {
                    size: 32
                }
            }
        }
    }
    attr {
        key: "shared_name"
        value {
            s: ""
        }
    }
}
node {
    name: "b_conv1/Adam/Assign"
    op: "Assign"
    input: "b_conv1/Adam"
    input: "b_conv1/Adam/Initializer/zeros"
```

```
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@b_conv1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "b_conv1/Adam/read"
  op: "Identity"
  input: "b_conv1/Adam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv1"
      }
    }
  }
}
node {
  name: "b_conv1/Adam_1/Initializer/zeros"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv1"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
        }
    }
attr {
    key: "value"
    value {
        tensor {
            dtype: DT_FLOAT
            tensor_shape {
                dim {
                    size: 32
                }
            }
            float_val: 0.0
        }
    }
}
node {
    name: "b_conv1/Adam_1"
    op: "VariableV2"
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@b_conv1"
            }
        }
    }
    attr {
        key: "container"
        value {
            s: ""
        }
    }
    attr {
        key: "dtype"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "shape"
        value {
            shape {
                dim {
                    size: 32
                }
            }
        }
    }
    attr {
        key: "shared_name"
        value {
            s: ""
        }
    }
}
node {
    name: "b_conv1/Adam_1/Assign"
    op: "Assign"
    input: "b_conv1/Adam_1"
```

```
input: "b_conv1/Adam_1/Initializer/zeros"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@b_conv1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "b_conv1/Adam_1/read"
  op: "Identity"
  input: "b_conv1/Adam_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv1"
      }
    }
  }
}
node {
  name: "b_conv2/Adam/Initializer/zeros"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
  attr {
    key: "dtype"
    value {
```

```
    type: DT_FLOAT
  }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_FLOAT
      tensor_shape {
        dim {
          size: 64
        }
      }
      float_val: 0.0
    }
  }
}
node {
  name: "b_conv2/Adam"
  op: "VariableV2"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 64
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
  name: "b_conv2/Adam/Assign"
  op: "Assign"
```

```
  input: "b_conv2/Adam"
  input: "b_conv2/Adam/Initializer/zeros"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "b_conv2/Adam/read"
  op: "Identity"
  input: "b_conv2/Adam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
}
node {
  name: "b_conv2/Adam_1/Initializer/zeros"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
  attr {
    key: "dtype"
```

```
    value {
      type: DT_FLOAT
    }
  }
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_FLOAT
      tensor_shape {
        dim {
          size: 64
        }
      }
      float_val: 0.0
    }
  }
}
node {
  name: "b_conv2/Adam_1"
  op: "VariableV2"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 64
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
  name: "b_conv2/Adam_1/Assign"
```

```
op: "Assign"
input: "b_conv2/Adam_1"
input: "b_conv2/Adam_1/Initializer/zeros"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@b_conv2"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "b_conv2/Adam_1/read"
  op: "Identity"
  input: "b_conv2/Adam_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
}
node {
  name: "b_fc1/Adam/Initializer/zeros"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
  attr {
```

```
key: "dtype"
value {
  type: DT_FLOAT
}
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_FLOAT
      tensor_shape {
        dim {
          size: 256
        }
      }
      float_val: 0.0
    }
  }
}
}
node {
  name: "b_fc1/Adam"
  op: "VariableV2"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 256
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
node {
```

```
name: "b_fc1/Adam/Assign"
op: "Assign"
input: "b_fc1/Adam"
input: "b_fc1/Adam/Initializer/zeros"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "__class"
  value {
    list {
      s: "loc:@b_fc1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
}
node {
  name: "b_fc1/Adam/read"
  op: "Identity"
  input: "b_fc1/Adam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
}
node {
  name: "b_fc1/Adam_1/Initializer/zeros"
  op: "Const"
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
}
```

```
attr {
  key: "dtype"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_FLOAT
      tensor_shape {
        dim {
          size: 256
        }
      }
      float_val: 0.0
    }
  }
}
node {
  name: "b_fc1/Adam_1"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 256
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
```

```
node {
  name: "b_fc1/Adam_1/Assign"
  op: "Assign"
  input: "b_fc1/Adam_1"
  input: "b_fc1/Adam_1/Initializer/zeros"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "b_fc1/Adam_1/read"
  op: "Identity"
  input: "b_fc1/Adam_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
}
node {
  name: "b_fc2/Adam/Initializer/zeros"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc2"
      }
    }
  }
}
```

```
        }
        attr {
          key: "dtype"
          value {
            type: DT_FLOAT
          }
        }
      attr {
        key: "value"
        value {
          tensor {
            dtype: DT_FLOAT
            tensor_shape {
              dim {
                size: 7
              }
            }
            float_val: 0.0
          }
        }
      }
    }
  node {
    name: "b_fc2/Adam"
    op: "VariableV2"
    attr {
      key: "__class"
      value {
        list {
          s: "loc:@b_fc2"
        }
      }
    }
    attr {
      key: "container"
      value {
        s: ""
      }
    }
    attr {
      key: "dtype"
      value {
        type: DT_FLOAT
      }
    }
    attr {
      key: "shape"
      value {
        shape {
          dim {
            size: 7
          }
        }
      }
    }
    attr {
      key: "shared_name"
      value {
        s: ""
      }
    }
  }
```

```
}

node {
  name: "b_fc2/Adam/Assign"
  op: "Assign"
  input: "b_fc2/Adam"
  input: "b_fc2/Adam/Initializer/zeros"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "b_fc2/Adam/read"
  op: "Identity"
  input: "b_fc2/Adam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc2"
      }
    }
  }
}
node {
  name: "b_fc2/Adam_1/Initializer/zeros"
  op: "Const"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc2"
      }
    }
  }
}
```

```
    }
}
attr {
  key: "dtype"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_FLOAT
      tensor_shape {
        dim {
          size: 7
        }
      }
      float_val: 0.0
    }
  }
}
node {
  name: "b_fc2/Adam_1"
  op: "VariableV2"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc2"
      }
    }
  }
  attr {
    key: "container"
    value {
      s: ""
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "shape"
    value {
      shape {
        dim {
          size: 7
        }
      }
    }
  }
  attr {
    key: "shared_name"
    value {
      s: ""
    }
  }
}
```

```
    }
}
node {
  name: "b_fc2/Adam_1/Assign"
  op: "Assign"
  input: "b_fc2/Adam_1"
  input: "b_fc2/Adam_1/Initializer/zeros"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "b_fc2/Adam_1/read"
  op: "Identity"
  input: "b_fc2/Adam_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc2"
      }
    }
  }
}
node {
  name: "Adam/learning_rate"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
        }
      attr {
        key: "value"
        value {
          tensor {
            dtype: DT_FLOAT
            tensor_shape {
            }
            float_val: 0.001000000474974513
          }
        }
      }
    }
  }
node {
  name: "Adam/beta1"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.8999999761581421
      }
    }
  }
}
node {
  name: "Adam/beta2"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_FLOAT
        tensor_shape {
        }
        float_val: 0.9990000128746033
      }
    }
  }
}
node {
  name: "Adam/epsilon"
  op: "Const"
  attr {
    key: "dtype"
    value {
```

```
    type: DT_FLOAT
  }
}
attr {
  key: "value"
  value {
    tensor {
      dtype: DT_FLOAT
      tensor_shape {
      }
      float_val: 9.9999993922529e-09
    }
  }
}
node {
  name: "Adam/update_W_conv1/ApplyAdam"
  op: "ApplyAdam"
  input: "W_conv1"
  input: "W_conv1/Adam"
  input: "W_conv1/Adam_1"
  input: "beta1_power/read"
  input: "beta2_power/read"
  input: "Adam/learning_rate"
  input: "Adam/beta1"
  input: "Adam/beta2"
  input: "Adam/epsilon"
  input: "gradients/AddN_3"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: false
    }
  }
  attr {
    key: "use_nesterov"
    value {
      b: false
    }
  }
}
node {
  name: "Adam/update_W_conv2/ApplyAdam"
  op: "ApplyAdam"
  input: "W_conv2"
  input: "W_conv2/Adam"
  input: "W_conv2/Adam_1"
```

```
input: "beta1_power/read"
input: "beta2_power/read"
input: "Adam/learning_rate"
input: "Adam/beta1"
input: "Adam/beta2"
input: "Adam/epsilon"
input: "gradients/AddN_1"
attr {
    key: "T"
    value {
        type: DT_FLOAT
    }
}
attr {
    key: "_class"
    value {
        list {
            s: "loc:@W_conv2"
        }
    }
}
attr {
    key: "use_locking"
    value {
        b: false
    }
}
attr {
    key: "use_nesterov"
    value {
        b: false
    }
}
node {
    name: "Adam/update_W_fc1/ApplyAdam"
    op: "ApplyAdam"
    input: "W_fc1"
    input: "W_fc1/Adam"
    input: "W_fc1/Adam_1"
    input: "beta1_power/read"
    input: "beta2_power/read"
    input: "Adam/learning_rate"
    input: "Adam/beta1"
    input: "Adam/beta2"
    input: "Adam/epsilon"
    input: "gradients/fc_1/MatMul_grad/tuple/control_dependency_1"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@W_fc1"
            }
        }
    }
}
```

```
attr {
  key: "use_locking"
  value {
    b: false
  }
}
attr {
  key: "use_nesterov"
  value {
    b: false
  }
}
}
node {
  name: "Adam/update_W_fc2/ApplyAdam"
  op: "ApplyAdam"
  input: "W_fc2"
  input: "W_fc2/Adam"
  input: "W_fc2/Adam_1"
  input: "beta1_power/read"
  input: "beta2_power/read"
  input: "Adam/learning_rate"
  input: "Adam/beta1"
  input: "Adam/beta2"
  input: "Adam/epsilon"
  input: "gradients/fc_2/MatMul_grad/tuple/control_dependency_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: false
    }
  }
  attr {
    key: "use_nesterov"
    value {
      b: false
    }
  }
}
node {
  name: "Adam/update_b_conv1/ApplyAdam"
  op: "ApplyAdam"
  input: "b_conv1"
  input: "b_conv1/Adam"
  input: "b_conv1/Adam_1"
  input: "beta1_power/read"
  input: "beta2_power/read"
```

```
input: "Adam/learning_rate"
input: "Adam/beta1"
input: "Adam/beta2"
input: "Adam/epsilon"
input: "gradients/AddN_2"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@b_conv1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: false
  }
}
attr {
  key: "use_nesterov"
  value {
    b: false
  }
}
}
node {
  name: "Adam/update_b_conv2/ApplyAdam"
  op: "ApplyAdam"
  input: "b_conv2"
  input: "b_conv2/Adam"
  input: "b_conv2/Adam_1"
  input: "beta1_power/read"
  input: "beta2_power/read"
  input: "Adam/learning_rate"
  input: "Adam/beta1"
  input: "Adam/beta2"
  input: "Adam/epsilon"
  input: "gradients/AddN"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
  attr {
    key: "use_locking"
```

```
    value {
      b: false
    }
  }
  attr {
    key: "use_nesterov"
    value {
      b: false
    }
  }
}
node {
  name: "Adam/update_b_fc1/ApplyAdam"
  op: "ApplyAdam"
  input: "b_fc1"
  input: "b_fc1/Adam"
  input: "b_fc1/Adam_1"
  input: "beta1_power/read"
  input: "beta2_power/read"
  input: "Adam/learning_rate"
  input: "Adam/beta1"
  input: "Adam/beta2"
  input: "Adam/epsilon"
  input: "gradients/fc_1/add_grad/tuple/control_dependency_1"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: false
    }
  }
  attr {
    key: "use_nesterov"
    value {
      b: false
    }
  }
}
node {
  name: "Adam/update_b_fc2/ApplyAdam"
  op: "ApplyAdam"
  input: "b_fc2"
  input: "b_fc2/Adam"
  input: "b_fc2/Adam_1"
  input: "beta1_power/read"
  input: "beta2_power/read"
  input: "Adam/learning_rate"
  input: "Adam/beta1"
```

```
input: "Adam/beta2"
input: "Adam/epsilon"
input: "gradients/fc_2/add_grad/tuple/control_dependency_1"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@b_fc2"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: false
  }
}
attr {
  key: "use_nesterov"
  value {
    b: false
  }
}
node {
  name: "Adam/mul"
  op: "Mul"
  input: "beta1_power/read"
  input: "Adam/beta1"
  input: "^Adam/update_W_conv1/ApplyAdam"
  input: "^Adam/update_W_conv2/ApplyAdam"
  input: "^Adam/update_W_fc1/ApplyAdam"
  input: "^Adam/update_W_fc2/ApplyAdam"
  input: "^Adam/update_b_conv1/ApplyAdam"
  input: "^Adam/update_b_conv2/ApplyAdam"
  input: "^Adam/update_b_fc1/ApplyAdam"
  input: "^Adam/update_b_fc2/ApplyAdam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
}
node {
  name: "Adam/Assign"
  op: "Assign"
```

```
input: "beta1_power"
input: "Adam/mul"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_conv1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: false
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "Adam/mul_1"
  op: "Mul"
  input: "beta2_power/read"
  input: "Adam/beta2"
  input: "^Adam/update_W_conv1/ApplyAdam"
  input: "^Adam/update_W_conv2/ApplyAdam"
  input: "^Adam/update_W_fc1/ApplyAdam"
  input: "^Adam/update_W_fc2/ApplyAdam"
  input: "^Adam/update_b_conv1/ApplyAdam"
  input: "^Adam/update_b_conv2/ApplyAdam"
  input: "^Adam/update_b_fc1/ApplyAdam"
  input: "^Adam/update_b_fc2/ApplyAdam"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
}
node {
  name: "Adam/Assign_1"
  op: "Assign"
  input: "beta2_power"
```

```
input: "Adam/mul_1"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_conv1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: false
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "Adam/update"
  op: "NoOp"
  input: "^Adam/Assign"
  input: "^Adam/Assign_1"
  input: "^Adam/update_W_conv1/ApplyAdam"
  input: "^Adam/update_W_conv2/ApplyAdam"
  input: "^Adam/update_W_fc1/ApplyAdam"
  input: "^Adam/update_W_fc2/ApplyAdam"
  input: "^Adam/update_b_conv1/ApplyAdam"
  input: "^Adam/update_b_conv2/ApplyAdam"
  input: "^Adam/update_b_fc1/ApplyAdam"
  input: "^Adam/update_b_fc2/ApplyAdam"
}
node {
  name: "Adam/value"
  op: "Const"
  input: "^Adam/update"
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@Variable"
      }
    }
  }
  attr {
    key: "dtype"
    value {
      type: DT_INT32
    }
  }
  attr {
```

```
key: "value"
value {
  tensor {
    dtype: DT_INT32
    tensor_shape {
    }
    int_val: 1
  }
}
}
node {
  name: "Adam"
  op: "AssignAdd"
  input: "Variable"
  input: "Adam/value"
  attr {
    key: "T"
    value {
      type: DT_INT32
    }
  }
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@Variable"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: false
    }
  }
}
node {
  name: "Merge/MergeSummary"
  op: "MergeSummary"
  input: "conv1/W_conv1"
  input: "conv1/b_conv1"
  input: "conv2/W_conv2"
  input: "conv2/b_conv2"
  input: "fc_1/W_fc1"
  input: "fc_1/b_fc1"
  input: "fc_2/W_fc2"
  input: "fc_2/b_fc2"
  input: "Entropy"
  input: "Reg_loss"
  attr {
    key: "N"
    value {
      i: 10
    }
  }
}
node {
  name: "init"
  op: "NoOp"
  input: "^Variable/Assign"
```

```
input: "^W_conv1/Adam/Assign"
input: "^W_conv1/Adam_1/Assign"
input: "^W_conv1/Assign"
input: "^W_conv2/Adam/Assign"
input: "^W_conv2/Adam_1/Assign"
input: "^W_conv2/Assign"
input: "^W_fc1/Adam/Assign"
input: "^W_fc1/Adam_1/Assign"
input: "^W_fc1/Assign"
input: "^W_fc2/Adam/Assign"
input: "^W_fc2/Adam_1/Assign"
input: "^W_fc2/Assign"
input: "^b_conv1/Adam/Assign"
input: "^b_conv1/Adam_1/Assign"
input: "^b_conv1/Assign"
input: "^b_conv2/Adam/Assign"
input: "^b_conv2/Adam_1/Assign"
input: "^b_conv2/Assign"
input: "^b_fc1/Adam/Assign"
input: "^b_fc1/Adam_1/Assign"
input: "^b_fc1/Assign"
input: "^b_fc2/Adam/Assign"
input: "^b_fc2/Adam_1/Assign"
input: "^b_fc2/Assign"
input: "^beta1_power/Assign"
input: "^beta2_power/Assign"
}
node {
  name: "save/Const"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_STRING
    }
  }
  attr {
    key: "value"
    value {
      tensor {
        dtype: DT_STRING
        tensor_shape {
        }
        string_val: "model"
      }
    }
  }
}
node {
  name: "save/SaveV2/tensor_names"
  op: "Const"
  attr {
    key: "dtype"
    value {
      type: DT_STRING
    }
  }
  attr {
    key: "value"
    value {
      tensor {
```

```
    dtype: DT_STRING
    tensor_shape {
        dim {
            size: 27
        }
    }
    string_val: "Variable"
    string_val: "W_conv1"
    string_val: "W_conv1/Adam"
    string_val: "W_conv1/Adam_1"
    string_val: "W_conv2"
    string_val: "W_conv2/Adam"
    string_val: "W_conv2/Adam_1"
    string_val: "W_fc1"
    string_val: "W_fc1/Adam"
    string_val: "W_fc1/Adam_1"
    string_val: "W_fc2"
    string_val: "W_fc2/Adam"
    string_val: "W_fc2/Adam_1"
    string_val: "b_conv1"
    string_val: "b_conv1/Adam"
    string_val: "b_conv1/Adam_1"
    string_val: "b_conv2"
    string_val: "b_conv2/Adam"
    string_val: "b_conv2/Adam_1"
    string_val: "b_fc1"
    string_val: "b_fc1/Adam"
    string_val: "b_fc1/Adam_1"
    string_val: "b_fc2"
    string_val: "b_fc2/Adam"
    string_val: "b_fc2/Adam_1"
    string_val: "beta1_power"
    string_val: "beta2_power"
}
}
}
}
}
node {
    name: "save/SaveV2/shape_and_slices"
    op: "Const"
    attr {
        key: "dtype"
        value {
            type: DT_STRING
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_STRING
                tensor_shape {
                    dim {
                        size: 27
                    }
                }
                string_val: ""
                string_val: ""
                string_val: ""
                string_val: ""
                string_val: ""
            }
        }
    }
}
```

```
        string_val: ""
        string_val: ""
    }
}
}
}
}
node {
    name: "save/SaveV2"
    op: "SaveV2"
    input: "save/Const"
    input: "save/SaveV2/tensor_names"
    input: "save/SaveV2/shape_and_slices"
    input: "Variable"
    input: "W_conv1"
    input: "W_conv1/Adam"
    input: "W_conv1/Adam_1"
    input: "W_conv2"
    input: "W_conv2/Adam"
    input: "W_conv2/Adam_1"
    input: "W_fc1"
    input: "W_fc1/Adam"
    input: "W_fc1/Adam_1"
    input: "W_fc2"
    input: "W_fc2/Adam"
    input: "W_fc2/Adam_1"
    input: "b_conv1"
    input: "b_conv1/Adam"
    input: "b_conv1/Adam_1"
    input: "b_conv2"
    input: "b_conv2/Adam"
    input: "b_conv2/Adam_1"
    input: "b_fc1"
    input: "b_fc1/Adam"
    input: "b_fc1/Adam_1"
    input: "b_fc2"
    input: "b_fc2/Adam"
    input: "b_fc2/Adam_1"
    input: "beta1_power"
    input: "beta2_power"
    attr {
        key: "dtypes"
    }
}
```

```
    value {
      list {
        type: DT_INT32
        type: DT_FLOAT
        type: DT_FLOAT
      }
    }
  }
}

node {
  name: "save/control_dependency"
  op: "Identity"
  input: "save/Const"
  input: "^save/SaveV2"
  attr {
    key: "T"
    value {
      type: DT_STRING
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@save/Const"
      }
    }
  }
}
node {
  name: "save/RestoreV2/tensor_names"
  op: "Const"
  device: "/device:CPU:0"
  attr {
    key: "dtype"
    value {
      type: DT_STRING
```

```
        }
    }
attr {
    key: "value"
    value {
        tensor {
            dtype: DT_STRING
            tensor_shape {
                dim {
                    size: 27
                }
            }
        string_val: "Variable"
        string_val: "W_conv1"
        string_val: "W_conv1/Adam"
        string_val: "W_conv1/Adam_1"
        string_val: "W_conv2"
        string_val: "W_conv2/Adam"
        string_val: "W_conv2/Adam_1"
        string_val: "W_fc1"
        string_val: "W_fc1/Adam"
        string_val: "W_fc1/Adam_1"
        string_val: "W_fc2"
        string_val: "W_fc2/Adam"
        string_val: "W_fc2/Adam_1"
        string_val: "b_conv1"
        string_val: "b_conv1/Adam"
        string_val: "b_conv1/Adam_1"
        string_val: "b_conv2"
        string_val: "b_conv2/Adam"
        string_val: "b_conv2/Adam_1"
        string_val: "b_fc1"
        string_val: "b_fc1/Adam"
        string_val: "b_fc1/Adam_1"
        string_val: "b_fc2"
        string_val: "b_fc2/Adam"
        string_val: "b_fc2/Adam_1"
        string_val: "beta1_power"
        string_val: "beta2_power"
    }
}
}
}
node {
    name: "save/RestoreV2/shape_and_slices"
    op: "Const"
    device: "/device:CPU:0"
    attr {
        key: "dtype"
        value {
            type: DT_STRING
        }
    }
    attr {
        key: "value"
        value {
            tensor {
                dtype: DT_STRING
                tensor_shape {
                    dim {
                        size: 27
                    }
                }
            }
        }
    }
}
```



```
        type: DT_FLOAT
        type: DT_FLOAT
    }
}
}
}
node {
    name: "save/Assign"
    op: "Assign"
    input: "Variable"
    input: "save/RestoreV2"
    attr {
        key: "T"
        value {
            type: DT_INT32
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@Variable"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "save/Assign_1"
    op: "Assign"
    input: "W_conv1"
    input: "save/RestoreV2:1"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
```

```
        s: "loc:@W_conv1"
    }
}
attr {
    key: "use_locking"
    value {
        b: true
    }
}
attr {
    key: "validate_shape"
    value {
        b: true
    }
}
}
node {
    name: "save/Assign_2"
    op: "Assign"
    input: "W_conv1/Adam"
    input: "save/RestoreV2:2"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@W_conv1"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "save/Assign_3"
    op: "Assign"
    input: "W_conv1/Adam_1"
    input: "save/RestoreV2:3"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
```

```
key: "_class"
value {
  list {
    s: "loc:@W_conv1"
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
}
node {
  name: "save/Assign_4"
  op: "Assign"
  input: "W_conv2"
  input: "save/RestoreV2:4"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "save/Assign_5"
  op: "Assign"
  input: "W_conv2/Adam"
  input: "save/RestoreV2:5"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@W_conv2"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "save/Assign_6"
    op: "Assign"
    input: "W_conv2/Adam_1"
    input: "save/RestoreV2:6"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@W_conv2"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "save/Assign_7"
    op: "Assign"
    input: "W_fc1"
    input: "save/RestoreV2:7"
    attr {
```

```
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "save/Assign_8"
  op: "Assign"
  input: "W_fc1/Adam"
  input: "save/RestoreV2:8"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "save/Assign_9"
  op: "Assign"
```

```
input: "W_fc1/Adam_1"
input: "save/RestoreV2:9"
attr {
  key: "T"
  value {
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_fc1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "save/Assign_10"
  op: "Assign"
  input: "W_fc2"
  input: "save/RestoreV2:10"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
```

```
node {
  name: "save/Assign_11"
  op: "Assign"
  input: "W_fc2/Adam"
  input: "save/RestoreV2:11"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "save/Assign_12"
  op: "Assign"
  input: "W_fc2/Adam_1"
  input: "save/RestoreV2:12"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_fc2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
```

```
        }
    }
}
node {
    name: "save/Assign_13"
    op: "Assign"
    input: "b_conv1"
    input: "save/RestoreV2:13"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@b_conv1"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "save/Assign_14"
    op: "Assign"
    input: "b_conv1/Adam"
    input: "save/RestoreV2:14"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@b_conv1"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
}
attr {
```

```
    key: "validate_shape"
    value {
      b: true
    }
  }
node {
  name: "save/Assign_15"
  op: "Assign"
  input: "b_conv1/Adam_1"
  input: "save/RestoreV2:15"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@b_conv1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "save/Assign_16"
  op: "Assign"
  input: "b_conv2"
  input: "save/RestoreV2:16"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "__class"
    value {
      list {
        s: "loc:@b_conv2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
}
```

```
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "save/Assign_17"
    op: "Assign"
    input: "b_conv2/Adam"
    input: "save/RestoreV2:17"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@b_conv2"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "save/Assign_18"
    op: "Assign"
    input: "b_conv2/Adam_1"
    input: "save/RestoreV2:18"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@b_conv2"
            }
        }
    }
}
attr {
```

```
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "save/Assign_19"
  op: "Assign"
  input: "b_fc1"
  input: "save/RestoreV2:19"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "save/Assign_20"
  op: "Assign"
  input: "b_fc1/Adam"
  input: "save/RestoreV2:20"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc1"
      }
    }
  }
}
```

```
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "save/Assign_21"
    op: "Assign"
    input: "b_fc1/Adam_1"
    input: "save/RestoreV2:21"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
            list {
                s: "loc:@b_fc1"
            }
        }
    }
    attr {
        key: "use_locking"
        value {
            b: true
        }
    }
    attr {
        key: "validate_shape"
        value {
            b: true
        }
    }
}
node {
    name: "save/Assign_22"
    op: "Assign"
    input: "b_fc2"
    input: "save/RestoreV2:22"
    attr {
        key: "T"
        value {
            type: DT_FLOAT
        }
    }
    attr {
        key: "_class"
        value {
```

```
    list {
      s: "loc:@b_fc2"
    }
  }
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
}
node {
  name: "save/Assign_23"
  op: "Assign"
  input: "b_fc2/Adam"
  input: "save/RestoreV2:23"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@b_fc2"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "save/Assign_24"
  op: "Assign"
  input: "b_fc2/Adam_1"
  input: "save/RestoreV2:24"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
}
```

```
attr {
  key: "_class"
  value {
    list {
      s: "loc:@b_fc2"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
}
node {
  name: "save/Assign_25"
  op: "Assign"
  input: "beta1_power"
  input: "save/RestoreV2:25"
  attr {
    key: "T"
    value {
      type: DT_FLOAT
    }
  }
  attr {
    key: "_class"
    value {
      list {
        s: "loc:@W_conv1"
      }
    }
  }
  attr {
    key: "use_locking"
    value {
      b: true
    }
  }
  attr {
    key: "validate_shape"
    value {
      b: true
    }
  }
}
node {
  name: "save/Assign_26"
  op: "Assign"
  input: "beta2_power"
  input: "save/RestoreV2:26"
  attr {
    key: "T"
    value {
```

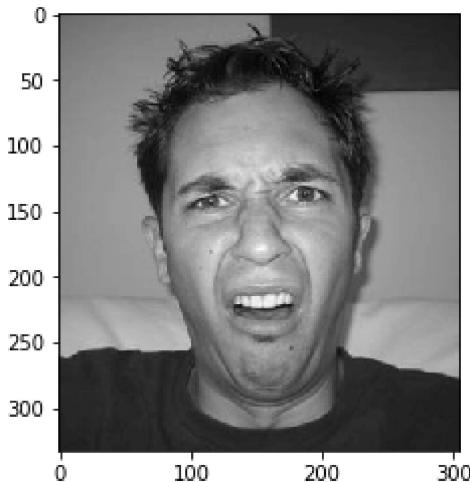
```
    type: DT_FLOAT
  }
}
attr {
  key: "_class"
  value {
    list {
      s: "loc:@W_conv1"
    }
  }
}
attr {
  key: "use_locking"
  value {
    b: true
  }
}
attr {
  key: "validate_shape"
  value {
    b: true
  }
}
node {
  name: "save/restore_all"
  op: "NoOp"
  input: "^save/Assign"
  input: "^save/Assign_1"
  input: "^save/Assign_10"
  input: "^save/Assign_11"
  input: "^save/Assign_12"
  input: "^save/Assign_13"
  input: "^save/Assign_14"
  input: "^save/Assign_15"
  input: "^save/Assign_16"
  input: "^save/Assign_17"
  input: "^save/Assign_18"
  input: "^save/Assign_19"
  input: "^save/Assign_2"
  input: "^save/Assign_20"
  input: "^save/Assign_21"
  input: "^save/Assign_22"
  input: "^save/Assign_23"
  input: "^save/Assign_24"
  input: "^save/Assign_25"
  input: "^save/Assign_26"
  input: "^save/Assign_3"
  input: "^save/Assign_4"
  input: "^save/Assign_5"
  input: "^save/Assign_6"
  input: "^save/Assign_7"
  input: "^save/Assign_8"
  input: "^save/Assign_9"
}
versions {
  producer: 27
}
```

In [7]:

```
x = sess.graph.get_tensor_by_name("input:0")
y_conv = sess.graph.get_tensor_by_name("output:0")
```

In [22]:

```
img = mpimg.imread('images_teste/image05.jpg')
gray = rgb2gray(img)
plt.imshow(gray, cmap = plt.get_cmap('gray'))
plt.show()
```



In [23]:

```
image_0 = np.resize(gray,(1,48,48,1))
tResult = testResult()
num_evaluations = 50
```

In [24]:

```
for i in range(0, num_evaluations):
    result = sess.run(y_conv, feed_dict={x:image_0})
    label = sess.run(tf.argmax(result, 1))
    label = label[0]
    label = int(label)
    tResult.evaluate(label)
tResult.display_result(num_evaluations)
```

```
anger = 0.0%
disgust = 0.0%
fear = 74.0%
happy = 26.0%
sad = 0.0%
surprise = 0.0%
neutral = 0.0%
```

In [ ]: