

COSC4337_CNN-2

```
[1]: # Convolutional Neural Network.

from __future__ import division, print_function, absolute_import

# Import MNIST data
from tensorflow.examples.tutorials.mnist import input_data
mnist = input_data.read_data_sets("/tmp/data/", one_hot=False)

import tensorflow as tf

# Training Parameters
learning_rate = 0.001
num_steps = 2000
batch_size = 128

# Network Parameters
num_input = 784 # MNIST data input (img shape: 28*28)
num_classes = 10 # MNIST total classes (0-9 digits)
dropout = 0.25 # Dropout, probability to drop a unit

# Create the neural network
def conv_net(x_dict, n_classes, dropout, reuse, is_training):
    # Define a scope for reusing the variables
    with tf.variable_scope('ConvNet', reuse=reuse):
        # TF Estimator input is a dict, in case of multiple inputs
        x = x_dict['images']

        # MNIST data input is a 1-D vector of 784 features (28*28 pixels)
        # Reshape to match picture format [Height x Width x Channel]
        # Tensor input become 4-D: [Batch Size, Height, Width, Channel]
        x = tf.reshape(x, shape=[-1, 28, 28, 1])

        # Convolution Layer with 32 filters and a kernel size of 5
        conv1 = tf.layers.conv2d(x, 32, 5, activation=tf.nn.relu)
        # Max Pooling (down-sampling) with strides of 2 and kernel size of 2
```

```

conv1 = tf.layers.max_pooling2d(conv1, 2, 2)

# Convolution Layer with 64 filters and a kernel size of 3
conv2 = tf.layers.conv2d(conv1, 64, 3, activation=tf.nn.relu)
# Max Pooling (down-sampling) with strides of 2 and kernel size of 2
conv2 = tf.layers.max_pooling2d(conv2, 2, 2)

# Flatten the data to a 1-D vector for the fully connected layer
fc1 = tf.contrib.layers.flatten(conv2)

# Fully connected layer (in tf contrib folder for now)
fc1 = tf.layers.dense(fc1, 1024)
# Apply Dropout (if is_training is False, dropout is not applied)
fc1 = tf.layers.dropout(fc1, rate=dropout, training=is_training)

# Output layer, class prediction
out = tf.layers.dense(fc1, n_classes)

return out

# Define the model function (following TF Estimator Template)
def model_fn(features, labels, mode):
    # Build the neural network
    # Because Dropout have different behavior at training and prediction time, ↵
    ↪we
    # need to create 2 distinct computation graphs that still share the same ↵
    ↪weights.
    logits_train = conv_net(features, num_classes, dropout, reuse=False,
                             is_training=True)
    logits_test = conv_net(features, num_classes, dropout, reuse=True,
                             is_training=False)

    # Predictions
    pred_classes = tf.argmax(logits_test, axis=1)
    pred_probas = tf.nn.softmax(logits_test)

    # If prediction mode, early return
    if mode == tf.estimator.ModeKeys.PREDICT:
        return tf.estimator.EstimatorSpec(mode, predictions=pred_classes)

    # Define loss and optimizer
    loss_op = tf.reduce_mean(tf.nn.sparse_softmax_cross_entropy_with_logits(
        logits=logits_train, labels=tf.cast(labels, dtype=tf.int32)))
    optimizer = tf.train.AdamOptimizer(learning_rate=learning_rate)
    train_op = optimizer.minimize(loss_op,
                                  global_step=tf.train.get_global_step())

```

```

# Evaluate the accuracy of the model
acc_op = tf.metrics.accuracy(labels=labels, predictions=pred_classes)

# TF Estimators requires to return a EstimatorSpec, that specify
# the different ops for training, evaluating, ...
estim_specs = tf.estimator.EstimatorSpec(
    mode=mode,
    predictions=pred_classes,
    loss=loss_op,
    train_op=train_op,
    eval_metric_ops={'accuracy': acc_op})

return estim_specs

# Build the Estimator
model = tf.estimator.Estimator(model_fn)

# Define the input function for training
input_fn = tf.estimator.inputs.numpy_input_fn(
    x={'images': mnist.train.images}, y=mnist.train.labels,
    batch_size=batch_size, num_epochs=None, shuffle=True)
# Train the Model
model.train(input_fn, steps=num_steps)

# Evaluate the Model
# Define the input function for evaluating
input_fn = tf.estimator.inputs.numpy_input_fn(
    x={'images': mnist.test.images}, y=mnist.test.labels,
    batch_size=batch_size, shuffle=False)
# Use the Estimator 'evaluate' method
e = model.evaluate(input_fn)

print("Testing Accuracy:", e['accuracy'])

```

WARNING:tensorflow:From <ipython-input-1-ce83c3549510>:8: read_data_sets (from tensorflow.contrib.learn.python.learn.datasets.mnist) is deprecated and will be removed in a future version.

Instructions for updating:

Please use alternatives such as official/mnist/dataset.py from tensorflow/models.

WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-packages\tensorflow_core\contrib\learn\python\learn\datasets\mnist.py:260: maybe_download (from tensorflow.contrib.learn.python.learn.datasets.base) is deprecated and will be removed in a future version.

Instructions for updating:

Please write your own downloading logic.

WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-

```

packages\tensorflow_core\contrib\learn\python\learn\datasets\mnist.py:262:
extract_images (from tensorflow.contrib.learn.python.learn.datasets.mnist) is
deprecated and will be removed in a future version.
Instructions for updating:
Please use tf.data to implement this functionality.
Extracting /tmp/data/train-images-idx3-ubyte.gz
WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-
packages\tensorflow_core\contrib\learn\python\learn\datasets\mnist.py:267:
extract_labels (from tensorflow.contrib.learn.python.learn.datasets.mnist) is
deprecated and will be removed in a future version.
Instructions for updating:
Please use tf.data to implement this functionality.
Extracting /tmp/data/train-labels-idx1-ubyte.gz
Extracting /tmp/data/t10k-images-idx3-ubyte.gz
Extracting /tmp/data/t10k-labels-idx1-ubyte.gz
WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-
packages\tensorflow_core\contrib\learn\python\learn\datasets\mnist.py:290:
DataSet.__init__ (from tensorflow.contrib.learn.python.learn.datasets.mnist) is
deprecated and will be removed in a future version.
Instructions for updating:
Please use alternatives such as official/mnist/dataset.py from
tensorflow/models.
INFO:tensorflow:Using default config.
WARNING:tensorflow:Using temporary folder as model directory:
C:\Users\RizkN\AppData\Local\Temp\tmpkb3n9lp1
INFO:tensorflow:Using config: {'_model_dir':
'C:\Users\RizkN\AppData\Local\Temp\tmpkb3n9lp1', '_tf_random_seed': None,
'_save_summary_steps': 100, '_save_checkpoints_steps': None,
'_save_checkpoints_secs': 600, '_session_config': allow_soft_placement: true
graph_options {
  rewrite_options {
    meta_optimizer_iterations: ONE
  }
}
, '_keep_checkpoint_max': 5, '_keep_checkpoint_every_n_hours': 10000,
'_log_step_count_steps': 100, '_train_distribute': None, '_device_fn': None,
'_protocol': None, '_eval_distribute': None, '_experimental_distribute': None,
'_experimental_max_worker_delay_secs': None, '_session_creation_timeout_secs':
7200, '_service': None, '_cluster_spec':
<tensorflow.python.training.server_lib.ClusterSpec object at
0x000001D7BA5C2E08>, '_task_type': 'worker', '_task_id': 0,
'_global_id_in_cluster': 0, '_master': '', '_evaluation_master': '',
'_is_chief': True, '_num_ps_replicas': 0, '_num_worker_replicas': 1}
WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-
packages\tensorflow_core\python\training\training_util.py:236:
Variable.initialized_value (from tensorflow.python.ops.variables) is deprecated
and will be removed in a future version.
Instructions for updating:

```

Use `Variable.read_value`. Variables in 2.X are initialized automatically both in eager and graph (inside `tf.defun`) contexts.

WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-packages\tensorflow_estimator\python\estimator\inputs\queues\feeding_queue_runner.py:62: `QueueRunner.__init__` (from tensorflow.python.training.queue_runner_impl) is deprecated and will be removed in a future version.

Instructions for updating:

To construct input pipelines, use the ``tf.data`` module.

WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-packages\tensorflow_estimator\python\estimator\inputs\queues\feeding_functions.py:500: `add_queue_runner` (from tensorflow.python.training.queue_runner_impl) is deprecated and will be removed in a future version.

Instructions for updating:

To construct input pipelines, use the ``tf.data`` module.

INFO:tensorflow:Calling model_fn.

WARNING:tensorflow:From <ipython-input-1-ce83c3549510>:36: `conv2d` (from tensorflow.python.layers.convolutional) is deprecated and will be removed in a future version.

Instructions for updating:

Use ``tf.keras.layers.Conv2D`` instead.

WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-packages\tensorflow_core\python\layers\convolutional.py:424: `Layer.apply` (from tensorflow.python.keras.engine.base_layer) is deprecated and will be removed in a future version.

Instructions for updating:

Please use ``layer.__call__`` method instead.

WARNING:tensorflow:From <ipython-input-1-ce83c3549510>:38: `max_pooling2d` (from tensorflow.python.layers.pooling) is deprecated and will be removed in a future version.

Instructions for updating:

Use `keras.layers.MaxPooling2D` instead.

WARNING:tensorflow:

The TensorFlow contrib module will not be included in TensorFlow 2.0.

For more information, please see:

- * <https://github.com/tensorflow/community/blob/master/rfcs/20180907-contrib-sunset.md>
- * <https://github.com/tensorflow/addons>
- * <https://github.com/tensorflow/io> (for I/O related ops)

If you depend on functionality not listed there, please file an issue.

WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-packages\tensorflow_core\contrib\layers\python\layers\layers.py:1634: `flatten` (from tensorflow.python.layers.core) is deprecated and will be removed in a future version.

Instructions for updating:

Use `keras.layers.flatten` instead.

WARNING:tensorflow:From <ipython-input-1-ce83c3549510>:49: `dense` (from tensorflow.python.layers.core) is deprecated and will be removed in a future

```

version.
Instructions for updating:
Use keras.layers.Dense instead.
WARNING:tensorflow:From <ipython-input-1-ce83c3549510>:51: dropout (from
tensorflow.python.layers.core) is deprecated and will be removed in a future
version.
Instructions for updating:
Use keras.layers.dropout instead.
INFO:tensorflow:Done calling model_fn.
INFO:tensorflow:Create CheckpointSaverHook.
WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-
packages\tensorflow_core\python\ops\array_ops.py:1475: where (from
tensorflow.python.ops.array_ops) is deprecated and will be removed in a future
version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
INFO:tensorflow:Graph was finalized.
INFO:tensorflow:Running local_init_op.
INFO:tensorflow:Done running local_init_op.
WARNING:tensorflow:From C:\Users\RizkN\.conda\envs\tf1\lib\site-
packages\tensorflow_core\python\training\monitored_session.py:882:
start_queue_runners (from tensorflow.python.training.queue_runner_impl) is
deprecated and will be removed in a future version.
Instructions for updating:
To construct input pipelines, use the `tf.data` module.
INFO:tensorflow:Saving checkpoints for 0 into
C:\Users\RizkN\AppData\Local\Temp\tmpkb3n9lp1\model.ckpt.
INFO:tensorflow:loss = 2.309338, step = 1
INFO:tensorflow:global_step/sec: 28.9113
INFO:tensorflow:loss = 0.16401601, step = 101 (3.466 sec)
INFO:tensorflow:global_step/sec: 28.9147
INFO:tensorflow:loss = 0.13811278, step = 201 (3.452 sec)
INFO:tensorflow:global_step/sec: 28.8025
INFO:tensorflow:loss = 0.021986876, step = 301 (3.472 sec)
INFO:tensorflow:global_step/sec: 29.286
INFO:tensorflow:loss = 0.09833016, step = 401 (3.415 sec)
INFO:tensorflow:global_step/sec: 29.4623
INFO:tensorflow:loss = 0.040909756, step = 501 (3.393 sec)
INFO:tensorflow:global_step/sec: 29.5904
INFO:tensorflow:loss = 0.057522487, step = 601 (3.380 sec)
INFO:tensorflow:global_step/sec: 29.8257
INFO:tensorflow:loss = 0.110231556, step = 701 (3.352 sec)
INFO:tensorflow:global_step/sec: 29.432
INFO:tensorflow:loss = 0.03939844, step = 801 (3.399 sec)
INFO:tensorflow:global_step/sec: 29.8613
INFO:tensorflow:loss = 0.0833751, step = 901 (3.348 sec)
INFO:tensorflow:global_step/sec: 29.8347
INFO:tensorflow:loss = 0.049353756, step = 1001 (3.353 sec)

```

```

INFO:tensorflow:global_step/sec: 29.8258
INFO:tensorflow:loss = 0.034437608, step = 1101 (3.353 sec)
INFO:tensorflow:global_step/sec: 24.024
INFO:tensorflow:loss = 0.014138788, step = 1201 (4.163 sec)
INFO:tensorflow:global_step/sec: 22.5574
INFO:tensorflow:loss = 0.016765349, step = 1301 (4.433 sec)
INFO:tensorflow:global_step/sec: 22.5068
INFO:tensorflow:loss = 0.030810857, step = 1401 (4.443 sec)
INFO:tensorflow:global_step/sec: 21.1416
INFO:tensorflow:loss = 0.020056078, step = 1501 (4.730 sec)
INFO:tensorflow:global_step/sec: 20.0947
INFO:tensorflow:loss = 0.021679306, step = 1601 (4.976 sec)
INFO:tensorflow:global_step/sec: 20.2858
INFO:tensorflow:loss = 0.0064062476, step = 1701 (4.930 sec)
INFO:tensorflow:global_step/sec: 19.8054
INFO:tensorflow:loss = 0.02267082, step = 1801 (5.050 sec)
INFO:tensorflow:global_step/sec: 19.4483
INFO:tensorflow:loss = 0.040724084, step = 1901 (5.141 sec)
INFO:tensorflow:Saving checkpoints for 2000 into
C:\Users\RizkN\AppData\Local\Temp\tmpkb3n9lp1\model.ckpt.
INFO:tensorflow:Loss for final step: 0.005416939.
INFO:tensorflow:Calling model_fn.
INFO:tensorflow:Done calling model_fn.
INFO:tensorflow:Starting evaluation at 2021-09-17T18:23:22Z
INFO:tensorflow:Graph was finalized.
INFO:tensorflow:Restoring parameters from
C:\Users\RizkN\AppData\Local\Temp\tmpkb3n9lp1\model.ckpt-2000
INFO:tensorflow:Running local_init_op.
INFO:tensorflow:Done running local_init_op.
INFO:tensorflow:Finished evaluation at 2021-09-17-18:23:23
INFO:tensorflow:Saving dict for global step 2000: accuracy = 0.9897, global_step
= 2000, loss = 0.03167975
INFO:tensorflow:Saving 'checkpoint_path' summary for global step 2000:
C:\Users\RizkN\AppData\Local\Temp\tmpkb3n9lp1\model.ckpt-2000
Testing Accuracy: 0.9897

```

[]: