TensorFlow   API r1.4

Module: tf.contrib.rnn

> **Contents**
> Classes
> Functions

Defined in `tensorflow/contrib/rnn/__init__.py` .

RNN Cells and additional RNN operations.

See RNN and Cells (contrib) guide.

Classes

`class AttentionCellWrapper` : Basic attention cell wrapper.

`class BasicLSTMCell` : Basic LSTM recurrent network cell.

`class BasicRNNCell` : The most basic RNN cell.

`class BidirectionalGridLSTMCell` : Bidirectional GridLstm cell.

`class CompiledWrapper` : Wraps step execution in an XLA JIT scope.

`class Conv1DLSTMCell` : 1D Convolutional LSTM recurrent network cell.

`class Conv2DLSTMCell` : 2D Convolutional LSTM recurrent network cell.

`class Conv3DLSTMCell` : 3D Convolutional LSTM recurrent network cell.

`class ConvLSTMCell` : Convolutional LSTM recurrent network cell.

`class CoupledInputForgetGateLSTMCell` : Long short-term memory unit (LSTM) recurrent network cell.

`class DeviceWrapper` : Operator that ensures an RNNCell runs on a particular device.

`class DropoutWrapper` : Operator adding dropout to inputs and outputs of the given cell.

`class EmbeddingWrapper` : Operator adding input embedding to the given cell.

`class FusedRNNCell` : Abstract object representing a fused RNN cell.

`class FusedRNNCellAdaptor` : This is an adaptor for RNNCell classes to be used with `FusedRNNCell` .

`class GLSTMCell` : Group LSTM cell (G-LSTM).

`class GRUBlockCell` : Block GRU cell implementation.

`class GRUBlockCellV2` : Temporary GRUBlockCell impl with a different variable naming scheme.

`class GRUCell` : Gated Recurrent Unit cell (cf. http://arxiv.org/abs/1406.1078).

`class GridLSTMCell` : Grid Long short-term memory unit (LSTM) recurrent network cell.

`class HighwayWrapper` : RNNCell wrapper that adds highway connection on cell input and output.

`class` `InputProjectionWrapper` : Operator adding an input projection to the given cell.

`class` `IntersectionRNNCell` : Intersection Recurrent Neural Network (+RNN) cell.

`class` `LSTMBlockCell` : Basic LSTM recurrent network cell.

`class` `LSTMBlockFusedCell` : FusedRNNCell implementation of LSTM.

`class` `LSTMBlockWrapper` : This is a helper class that provides housekeeping for LSTM cells.

`class` `LSTMCell` : Long short-term memory unit (LSTM) recurrent network cell.

`class` `LSTMStateTuple` : Tuple used by LSTM Cells for `state_size` , `zero_state` , and output state.

`class` `LayerNormBasicLSTMCell` : LSTM unit with layer normalization and recurrent dropout.

`class` `MultiRNNCell` : RNN cell composed sequentially of multiple simple cells.

`class` `NASCell` : Neural Architecture Search (NAS) recurrent network cell.

`class` `OutputProjectionWrapper` : Operator adding an output projection to the given cell.

`class` `PhasedLSTMCell` : Phased LSTM recurrent network cell.

`class` `RNNCell` : Abstract object representing an RNN cell.

`class` `ResidualWrapper` : RNNCell wrapper that ensures cell inputs are added to the outputs.

`class` `TimeFreqLSTMCell` : Time-Frequency Long short-term memory unit (LSTM) recurrent network cell.

`class` `TimeReversedFusedRNN` : This is an adaptor to time-reverse a FusedRNNCell.

`class` `UGRNNCell` : Update Gate Recurrent Neural Network (UGRNN) cell.

Functions

`stack_bidirectional_dynamic_rnn(...)` : Creates a dynamic bidirectional recurrent neural network.

`stack_bidirectional_rnn(...)` : Creates a bidirectional recurrent neural network.

`static_bidirectional_rnn(...)` : Creates a bidirectional recurrent neural network.

`static_rnn(...)` : Creates a recurrent neural network specified by RNNCell `cell` .

`static_state_saving_rnn(...)` : RNN that accepts a state saver for time-truncated RNN calculation.

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | **Privacy**