tf.contrib.legacy_seq2seq.rnn_decoder

```
rnn_decoder(
    decoder_inputs,
    initial_state,
    cell,
    loop_function=None,
    scope=None
)
```

Defined in tensorflow/contrib/legacy_seq2seq/python/ops/seq2seq.py .

RNN decoder for the sequence-to-sequence model.

Args:

- `decoder_inputs` : A list of 2D Tensors [batch_size x input_size].
- `initial_state` : 2D Tensor with shape [batch_size x cell.state_size].
- `cell` : rnn_cell.RNNCell defining the cell function and size.
- `loop_function` : If not None, this function will be applied to the i-th output in order to generate the i+1-st input, and decoder_inputs will be ignored, except for the first element ("GO" symbol). This can be used for decoding, but also for training to emulate http://arxiv.org/abs/1506.03099. Signature – loop_function(prev, i) = next
  - prev is a 2D Tensor of shape [batch_size x output_size],
  - i is an integer, the step number (when advanced control is needed),
  - next is a 2D Tensor of shape [batch_size x input_size].
- `scope` : VariableScope for the created subgraph; defaults to "rnn_decoder".

Returns:

A tuple of the form (outputs, state), where: `outputs` : *A list of the same length as decoder_inputs of 2D Tensors with shape [batch_size x output_size] containing generated outputs.* `state` : The state of each cell at the final time-step. It is a 2D Tensor of shape [batch_size x cell.state_size]. (Note that in some cases, like basic RNN cell or GRU cell, outputs and states can be the same. They are different for LSTM cells though.)

Last updated November 2, 2017.

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | **Privacy**