

tf.contrib.rnn.stack_bidirectional_rnn

```
stack_bidirectional_rnn(  
    cells_fw,  
    cells_bw,  
    inputs,  
    initial_states_fw=None,  
    initial_states_bw=None,  
    dtype=None,  
    sequence_length=None,  
    scope=None  
)
```

Defined in [tensorflow/contrib/rnn/python/ops/rnn.py](https://github.com/tensorflow/tensorflow/blob/master/tensorflow/contrib/rnn/python/ops/rnn.py).

Creates a bidirectional recurrent neural network.

Stacks several bidirectional rnn layers. The combined forward and backward layer outputs are used as input of the next layer. `tf.bidirectional_rnn` does not allow to share forward and backward information between layers. The `input_size` of the first forward and backward cells must match. The initial state for both directions is zero and no intermediate states are returned.

As described in <https://arxiv.org/abs/1303.5778>

Args:

- `cells_fw`: List of instances of `RNNCell`, one per layer, to be used for forward direction.
- `cells_bw`: List of instances of `RNNCell`, one per layer, to be used for backward direction.
- `inputs`: A length `T` list of inputs, each a tensor of shape `[batch_size, input_size]`, or a nested tuple of such elements.
- `initial_states_fw`: (optional) A list of the initial states (one per layer) for the forward RNN. Each tensor must have an appropriate type and shape `[batch_size, cell_fw.state_size]`.
- `initial_states_bw`: (optional) Same as for `initial_states_fw`, but using the corresponding properties of `cells_bw`.
- `dtype`: (optional) The data type for the initial state. Required if either of the initial states are not provided.
- `sequence_length`: (optional) An `int32/int64` vector, size `[batch_size]`, containing the actual lengths for each of the sequences.
- `scope`: `VariableScope` for the created subgraph; defaults to `None`.

Returns:

A tuple `(outputs, output_state_fw, output_state_bw)` where: `outputs` is a length `T` list of outputs (one for each input), which are depth-concatenated forward and backward outputs. `output_states_fw` is the final states, one tensor per layer, of the forward rnn. `output_states_bw` is the final states, one tensor per layer, of the backward rnn.

Raises:

- `TypeError`: If `cell_fw` or `cell_bw` is not an instance of `RNNCell`.

- `ValueError` : If inputs is None, not a list or an empty list.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](#), and code samples are licensed under the [Apache 2.0 License](#). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated November 2, 2017.

Stay Connected

[Blog](#)

[GitHub](#)

[Twitter](#)

Support

[Issue Tracker](#)

[Release Notes](#)

[Stack Overflow](#)

English

[Terms](#) | [Privacy](#)