tf.nn.static_state_saving_rnn

Aliases:

- `tf.contrib.rnn.static_state_saving_rnn`
- `tf.nn.static_state_saving_rnn`

```
static_state_saving_rnn(
    cell,
    inputs,
    state_saver,
    state_name,
    sequence_length=None,
    scope=None
)
```

Defined in `tensorflow/python/ops/rnn.py`.

See the guide: RNN and Cells (contrib) > Recurrent Neural Networks

RNN that accepts a state saver for time-truncated RNN calculation.

Args:

- `cell`: An instance of `RNNCell`.
- `inputs`: A length T list of inputs, each a `Tensor` of shape `[batch_size, input_size]`.
- `state_saver`: A state saver object with methods `state` and `save_state`.
- `state_name`: Python string or tuple of strings. The name to use with the state_saver. If the cell returns tuples of states (i.e., `cell.state_size` is a tuple) then `state_name` should be a tuple of strings having the same length as `cell.state_size`. Otherwise it should be a single string.
- `sequence_length`: (optional) An int32/int64 vector size [batch_size]. See the documentation for rnn() for more details about sequence_length.
- `scope`: VariableScope for the created subgraph; defaults to "rnn".

Returns:

A pair (outputs, state) where: outputs is a length T list of outputs (one for each input) states is the final state

Raises:

- `TypeError`: If `cell` is not an instance of RNNCell.
- `ValueError`: If `inputs` is `None` or an empty list, or if the arity and type of `state_name` does not match that of `cell.state_size`.

Last updated November 2, 2017.