tf.sparse_concat

```
sparse_concat(
    axis,
    sp_inputs,
    name=None,
    expand_nonconcat_dim=False,
    concat_dim=None
)
```

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

See the guide: Sparse Tensors > Manipulation

Concatenates a list of `SparseTensor` along the specified dimension.

Concatenation is with respect to the dense versions of each sparse input. It is assumed that each inputs is a `SparseTensor` whose elements are ordered along increasing dimension number.

If expand_nonconcat_dim is False, all inputs' shapes must match, except for the concat dimension. If expand_nonconcat_dim is True, then inputs' shapes are allowed to vary among all inputs.

The `indices`, `values`, and `shapes` lists must have the same length.

If expand_nonconcat_dim is False, then the output shape is identical to the inputs', except along the concat dimension, where it is the sum of the inputs' sizes along that dimension.

If expand_nonconcat_dim is True, then the output shape along the non-concat dimensions will be expand to be the largest among all inputs, and it is the sum of the inputs sizes along the concat dimension.

The output elements will be resorted to preserve the sort order along increasing dimension number.

This op runs in `O(M log M)` time, where `M` is the total number of non-empty values across all inputs. This is due to the need for an internal sort in order to concatenate efficiently across an arbitrary dimension.

For example, if `axis = 1` and the inputs are

```
sp_inputs[0]: shape = [2, 3]
[0, 2]: "a"
[1, 0]: "b"
[1, 1]: "c"

sp_inputs[1]: shape = [2, 4]
[0, 1]: "d"
[0, 2]: "e"
```

then the output will be

```
shape = [2, 7]
[0, 2]: "a"
[0, 4]: "d"
[0, 5]: "e"
[1, 0]: "b"
[1, 1]: "c"
```

Graphically this is equivalent to doing

```
[    a] concat [  d e  ] = [    a   d e  ]
[b c  ]        [      ]   [b c          ]
```

Another example, if 'axis = 1' and the inputs are

```
sp_inputs[0]: shape = [3, 3]
[0, 2]: "a"
[1, 0]: "b"
[2, 1]: "c"

sp_inputs[1]: shape = [2, 4]
[0, 1]: "d"
[0, 2]: "e"
```

if expand_nonconcat_dim = False, this will result in an error. But if expand_nonconcat_dim = True, this will result in:

```
shape = [3, 7]
[0, 2]: "a"
[0, 4]: "d"
[0, 5]: "e"
[1, 0]: "b"
[2, 1]: "c"
```

Graphically this is equivalent to doing

```
[    a] concat [  d e  ] = [    a   d e  ]
[b    ]        [      ]   [b           ]
[  c  ]                    [  c         ]
```

Args:

- `axis`: Dimension to concatenate along. Must be in range [-rank, rank), where rank is the number of dimensions in each input `SparseTensor`.
- `sp_inputs`: List of `SparseTensor` to concatenate.
- `name`: A name prefix for the returned tensors (optional).
- `expand_nonconcat_dim`: Whether to allow the expansion in the non-concat dimensions. Defaulted to False.
- `concat_dim`: The old (deprecated) name for axis.

Returns:

A `SparseTensor` with the concatenated output.

Raises:

- `TypeError`: If `sp_inputs` is not a list of `SparseTensor`.

Last updated November 2, 2017.

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | **Privacy**