tf.contrib.layers.safe_embedding_lookup_sparse

```
safe_embedding_lookup_sparse(
    embedding_weights,
    sparse_ids,
    sparse_weights=None,
    combiner=None,
    default_id=None,
    name=None,
    partition_strategy='div',
    max_norm=None
)
```

Defined in `tensorflow/contrib/layers/python/layers/embedding_ops.py`.

See the guide: Layers (contrib) > Higher level ops for building neural network layers

Lookup embedding results, accounting for invalid IDs and empty features.

The partitioned embedding in `embedding_weights` must all be the same shape except for the first dimension. The first dimension is allowed to vary as the vocabulary size is not necessarily a multiple of `P`. `embedding_weights` may be a `PartitionedVariable` as returned by using `tf.get_variable()` with a partitioner.

Invalid IDs (< 0) are pruned from input IDs and weights, as well as any IDs with non-positive weight. For an entry with no features, the embedding vector for `default_id` is returned, or the 0-vector if `default_id` is not supplied.

The ids and weights may be multi-dimensional. Embeddings are always aggregated along the last dimension.

Args:

- `embedding_weights`: A list of `P` float tensors or values representing partitioned embedding tensors. Alternatively, a `PartitionedVariable`, created by partitioning along dimension 0. The total unpartitioned shape should be `[e_0, e_1, ..., e_m]`, where `e_0` represents the vocab size and `e_1, ..., e_m` are the embedding dimensions.
- `sparse_ids`: `SparseTensor` of shape `[d_0, d_1, ..., d_n]` containing the ids. `d_0` is typically batch size.
- `sparse_weights`: `SparseTensor` of same shape as `sparse_ids`, containing float weights corresponding to `sparse_ids`, or `None` if all weights are be assumed to be 1.0.
- `combiner`: A string specifying how to combine embedding results for each entry. Currently "mean", "sqrtn" and "sum" are supported, with "mean" the default.
- `default_id`: The id to use for an entry with no features.
- `name`: A name for this operation (optional).
- `partition_strategy`: A string specifying the partitioning strategy. Currently `"div"` and `"mod"` are supported. Default is `"div"`.
- `max_norm`: If not None, all embeddings are l2-normalized to max_norm before combining.

Returns:

Dense tensor of shape `[d_0, d_1, ..., d_{n-1}, e_1, ..., e_m]`.

Raises:

- `ValueError`: if `embedding_weights` is empty.

Last updated November 2, 2017.

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | **Privacy**