tf.contrib.layers.sparse_column_with_hash_bucket

```
sparse_column_with_hash_bucket(
    column_name,
    hash_bucket_size,
    combiner='sum',
    dtype=tf.string
)
```

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

See the guide: Layers (contrib) > Feature columns

Creates a _SparseColumn with hashed bucket configuration.

Use this when your sparse features are in string or integer format, but you don't have a vocab file that maps each value to an integer ID. output_id = Hash(input_feature_string) % bucket_size

Args:

- `column_name`: A string defining sparse column name.
- `hash_bucket_size`: An int that is > 1. The number of buckets.
- `combiner`: A string specifying how to reduce if the sparse column is multivalent. Currently "mean", "sqrtn" and "sum" are supported, with "sum" the default. "sqrtn" often achieves good accuracy, in particular with bag-of-words columns.
  - "sum": do not normalize features in the column
  - "mean": do l1 normalization on features in the column
  - "sqrtn": do l2 normalization on features in the column For more information: `tf.embedding_lookup_sparse`.
- `dtype`: The type of features. Only string and integer types are supported.

Returns:

A _SparseColumn with hashed bucket configuration

Raises:

- `ValueError`: hash_bucket_size is not greater than 2.
- `ValueError`: dtype is neither string nor integer.

Last updated November 2, 2017.