TensorFlow   API r1.4

tf.contrib.lookup.TextFileInitializer

Class **TextFileInitializer**

Inherits From: `TableInitializerBase`

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

Table initializers from a text file.

This initializer assigns one entry in the table for each line in the file.

The key and value type of the table to initialize is given by `key_dtype` and `value_dtype` .

The key and value content to get from each line is specified by the `key_index` and `value_index` .

- `TextFileIndex.LINE_NUMBER` means use the line number starting from zero, expects data type int64.
- `TextFileIndex.WHOLE_LINE` means use the whole line content, expects data type string.
- A value `>=0` means use the index (starting at zero) of the split line based on `delimiter` .

For example if we have a file with the following content:

```
emerson 10
lake 20
palmer 30
```

The following snippet initializes a table with the first column as keys and second column as values:

- `emerson -> 10`
- `lake -> 20`
- `palmer -> 30`

```
table = tf.contrib.lookup.HashTable(tf.contrib.lookup.TextFileInitializer(
    "test.txt", tf.string, 0, tf.int64, 1, delimiter=" "), -1)
...
table.init.run()
```

Similarly to initialize the whole line as keys and the line number as values.

- `emerson 10 -> 0`
- `lake 20 -> 1`

- **palmer 30 -> 2**

```
table = tf.contrib.lookup.HashTable(tf.contrib.lookup.TextFileInitializer(
    "test.txt", tf.string, tf.contrib.lookup.TextFileIndex.WHOLE_LINE,
    tf.int64, tf.contrib.lookup.TextFileIndex.LINE_NUMBER, delimiter=" "), -1)
...
table.init.run()
```

Properties

key_dtype

The expected table key dtype.

value_dtype

The expected table value dtype.

Methods

__init__

```
__init__(
    filename,
    key_dtype,
    key_index,
    value_dtype,
    value_index,
    vocab_size=None,
    delimiter='\t',
    name=None
)
```

Constructs a table initializer object to populate from a text file.

It generates one key-value pair per line. The type of table key and value are specified by `key_dtype` and `value_dtype`, respectively. Similarly the content of the key and value are specified by the key_index and value_index.

- TextFileIndex.LINE_NUMBER means use the line number starting from zero, expects data type int64.
- TextFileIndex.WHOLE_LINE means use the whole line content, expects data type string.
- A value >=0 means use the index (starting at zero) of the split line based on `delimiter`.

Args:

- `filename` : The filename of the text file to be used for initialization. The path must be accessible from wherever the graph is initialized (eg. trainer or eval workers). The filename may be a scalar `Tensor` .
- `key_dtype` : The `key` data type.
- `key_index` : the index that represents information of a line to get the table 'key' values from.
- `value_dtype` : The `value` data type.
- `value_index` : the index that represents information of a line to get the table 'value' values from.'
- `vocab_size` : The number of elements in the file, if known.

- `delimiter` : The delimiter to separate fields in a line.
- `name` : A name for the operation (optional).

Raises:

- `ValueError` : when the filename is empty, or when the table key and value data types do not match the expected data types.

`initialize`

```
initialize(table)
```

Initializes the table from a text file.

Args:

- `table` : The table to be initialized.

Returns:

The operation that initializes the table.

Raises:

- `TypeError` : when the keys and values data types do not match the table key and value data types.

Stay Connected

Blog

GitHub

Twitter

Support

Issue Tracker

Release Notes

Stack Overflow

English

Terms | Privacy