

Define and Run

MokkeMeguru¹

2020-02-12 Wed

¹meguru.mokke@gmail.com

Outline

1 Define and Run

2 Define and Run を深く見る

3 Feed Dict

Presentaion agenda

1 Define and Run

2 Define and Run を深く見る

3 Feed Dict

Define and Run

Define and Run vs Define by Run

- 計算グラフ: 演算、モデル = $f(x)$

Define and Run

計算グラフの定義と実行

Pros:

- 高速化のための実装が容易
- 初期化などについて特殊な実装ができる

Cons:

- デバッグが困難

ref:

- Tensorflow 1.x, Tensorflow 2.x
- MxNet, theano, Caffe, etc.

Define by Run

実行時に計算グラフを定義

Pros:

- デバッグが容易
- Python のインタプリタとの親和性が高い

Cons:

- 高速化は難しい

ref:

- PyTorch
- Tensorflow 2.x

Define and Run

計算グラフの定義と実行 (Tensorflow 1.12)

```
1 left = tf.constant(1.0)
2 right = tf.constant(2.0)
3 # ここで計算グラフ (演算) を定義
4 add_op = tf.add(left, right)
5
6 # Session (実行領域) を立ち上げる
7 with tf.Session() as sess:
8     # 実行を行う
9     result = sess.run([add_op])
10    print(result)
11    # => [3.0]
```

Define by Run

実行時に計算グラフを定義 (Tensorflow 2.1.0)

```
1 left = tf.constant(1.0)
2 right = tf.constant(2.0)
3 # 計算グラフの実行と定義
4 result = tf.add(left, right)
5 print(result)
6 # => [3.0]
```

Presentaion agenda

1 Define and Run

2 Define and Run を深く見る

3 Feed Dict

クラス TensorflowHello

TensorflowHello クラス

- 初期化子に string 型の引数 name を引き受ける
- インスタンスの実行時に session を引き受け、operation #greeting を実行する

```
1 class TensorflowHello():
2     def __init__(self, name: str = 'Jack'):
3         self.name = 'Jack'
4         self.greeting = tf.print(
5             'Hello, Tensorflow! --{}'.format(self.name))
6
7     def __call__(self, sess: tf.Session):
8         sess.run(self.greeting)
```

課題 1-1

- 次のコードの実行で得られる標準出力を予測せよ。

```
1 print('[Info] initialize')
2 tf_hello = TensorflowHello()
3 print('[Info] Call')
4 with tf.Session() as sess:
5     tf_hello(sess)
```

課題 1-2

- 実際に ~ のコードを実行し予測と比較せよ。

1 `pipenv run python define_and_run/define_and_run.py`

読み方: `pipenv` の環境下で、`define_and_run.py` を実行する

- また、`define_by_run.py` を実行しその差を確認せよ。
* コードを眺めることで `define_and_run` と `define_by_run` の違いを確認できる

Presentaion agenda

1 Define and Run

2 Define and Run を深く見る

3 Feed Dict

Placeholder と Feed Dict

Define and Run の計算グラフ ⇒ 鋳型のようなもの

- 1 どこからデータを流し込む？
→ **placeholder**
- 2 何を流し込む？
→ **feed_dict**

クラス TensorflowHelloFeedDict

```
1 class TensorflowHelloFeedDict():
2     def __init__(self):
3         # placeholder ここがデータの入り口
4         self.name = tf.placeholder(tf.string)
5         self.greeting = tf.print(tf.strings.format(
6             'Hello, Tensorflow! --{ }', self.name))
7
8     def __call__(self, sess: tf.Session,
9                 name: str = 'Jack'):
10        # feed_dict
11        # self.name という入り口に name を流し込む
12        sess.run(self.greeting,
13                 feed_dict={self.name: name})
```

課題 2

- `feed_dict.py` を実行せよ。