



TensorFlow

Курс “Практическое применение по TensorFlow”

Шигапова Фирюза Зинатуллаевна

1-й семестр, 2019 г.



<https://github.com/Firyuza/TensorFlowPractice>

Custom model

- Custom model contains custom Layers
- Custom Layers use `self.add_weight` to create trainable variables

!!!Then have to save weights in specific way

Otherwise, TensorFlow cannot save (serialize) model via all existing ways

Save weights

```
file = h5py.File(file_path, 'w')  
weight = network.get_weights()  
for i in range(len(weight)):  
    file.create_dataset('weight' + str(i), data=weight[i])  
file.close()
```

1. Create h5 file
2. Get weights from model
3. Save them via h5 file

Save optimizer params

```
file = h5py.File(file_path, 'w')
weight = optimizer.get_weights()
for i in range(len(weight)):
    file.create_dataset('weight' + str(i), data=weight[i])
file.close()
```

1. Create h5 file
2. Get params from optimizer
3. Save them via h5 file

@tf.function

Function with `@tf.function` annotation:

- no `numpy()` calls
- no `self.` variables that are created within wrapped method

tf.data.map

- Try implement **map** function using TF ops
- Complicated Python code may not work properly

tensorflow_addons

Use **tensorflow_addons** from contributors for new functionality

```
pip install tensorflow_addons
```

```
import tensorflow_addons as tfa
```

- tfa
- tfa.activations
- tfa.callbacks
- tfa.image
- tfa.layers
- tfa.losses
- tfa.metrics
- tfa.optimizers
- tfa.rnn
- tfa.seq2seq
- tfa.text