Congratulations! You passed!

Grade received 100% **To pass** 80% or higher

Go to next item

Graded Quiz: Test your Project Understanding

Latest Submission Grade 100%

1. Can the TensorFlow Model Server with Docker simultaneously listen to gRPC and REST API endpoints?

1 / 1 point

[✓ Yes]

No

Correct
Correct! You can have both gRPC and REST API ports open at the same time, or choose to only open one or the other. We opened both ports in the hands-on component of this project.

2. What is exported when you call the tf.saved_model.save(model, path) function?

1 / 1 point

Model structure

Model weights

[✓ Model structure, weights, as well as signatures as Protobuf]

Correct
Correct!

3. You have trained and exported a TensorFlow model used to classify the sentiment of Amazon product reviews. The model is named **amazon_review** and has been exported to **/path/to/amazon_review/**

1 / 1 point

How would you start a TensorFlow Serving container and open the gRPC API port to serve your model? The container expects models to be in **/models/amazon_review**

```
docker run -p 8501:8501 \
    --mount type=bind,\
    source=/path/to/amazon_review/,target=/models/amazon_review \
    -e MODEL_NAME=my_model \
    -t tensorflow/serving
```

[✓]
```
docker run -p 8500:8500 \
    --mount type=bind,\
    source=/path/to/amazon_review/,target=/models/amazon_review \
    -e MODEL_NAME=my_model \
    -t tensorflow/serving
```

```
docker run -p 8500:8500 \
    --mount type=bind,\
    source=/models/amazon_review,target=/path/to/amazon_review/ \
    -e MODEL_NAME=my_model \
    -t tensorflow/serving
```

Correct
Good job! In this case, we have started a Docker container, published the gRPC port 8500 to our host's port 8500, and taken a model we named **amazon_review** and bound it to the default model base path. We also pass the name of the model as an environment variable, which will be important when we query the model .

4. Suppose you are building a text classification model with tf.keras. The input data consists of sentences. The labels to predict are either 0 or 1.

1 / 1 point

One way to represent the text is to convert sentences into embedding vectors. As we did in the hands-on project, you use a pre-trained text embedding model from [TensorFlow Hub]() called [google/tf2-preview/nnlm-en-dim50/1]() as the first layer. The model build looks like the following:

```
URL = "https://tfhub.dev/google/tf2-preview/nnlm-en-dim50/1"
hub_layer = hub.KerasLayer(URL, output_shape=[50], input_shape=[],
                           dtype=tf.string, trainable=False)
model = keras.Sequential()
model.add(hub_layer)
model.add(keras.layers.Dense(16, activation='relu'))
model.add(keras.layers.Dense(1, activation='sigmoid'))

model.summary()
```

What can be said about this model and its layers? Select all that apply.

- [ ] The model parameters or weights associated with the hub_layer are updated via backpropagation during training.
- [x] The model parameters or weights associated with the hub_layer are not updated via backpropagation during training.

✓ **Correct**
Correct! By setting **trainable=False** we ensure that the paramaters/weights associated with the hub_layer are not updated (frozen) during training. Only the paramaters of the Dense layers are trainable and learned during training.

- [x] We don't have to worry about text preprocessing.

✓ **Correct**
Correct! The text embedding module takes a batch of sentences in a 1D tensor of strings as input and outputs the embedding vectors of shape (batch_size, embedding_dim) corresponding to the sentences. It preprocesses the input by splitting on spaces. It splits the sentence into tokens, embeds each token and then combines the embedding.

- [x] We can benefit from transfer learning.

✓ **Correct**
Correct!