What does a SavedModel contain? How do you inspect its content?

Ans. A SavedModel contains a complete TensorFlow program, including trained parameters (i.e, tf. Variable s) and computation. It does not require the original model building code to run, which makes it useful for sharing or deploying with TFLite, TensorFlow. js, TensorFlow Serving, or TensorFlow Hub.

When should you use TF Serving? What are its main features? What are some tools you can use to deploy it?

Ans. TensorFlow Serving allows us to select which version of a model, or "servable" we want to use when we make inference requests. Each version will be exported to a different sub-directory under the given path. TensorFlow Serving makes it easy to deploy new algorithms and experiments, while keeping the same server architecture and APIs. TensorFlow Serving provides out-of-the-box integration with TensorFlow models, but can be easily extended to serve other types of models and data. In TensorFlow. js there are two ways to train a machine learning model: using the Layers API with LayersModel. The code above is the standard recipe when training a model with the Core API:

Loop over the number of epochs.

Inside each epoch, loop over your batches of data. ...

For each batch, call optimizer.

How do you deploy a model across multiple TF Serving instances?

Ans. For Windows 10, we will use a TensorFlow serving image.

Step 1: Install the Docker App.

Step 2: Pull the TensorFlow Serving Image. docker pull tensorflow/serving. ...

Step 3: Create and Train the Model. ...

Step 4: Save the Model. ...

Step 5: Serving the model using Tensorflow Serving. ...

Step 6: Make a REST request the model to predict.

When should you use the gRPC API rather than the REST API to query a model served by TF Serving?

Ans. gRPC is a robust open-source RPC (Remote Procedure Call) framework used to build scalable and fast APIs. It allows the client and server applications to communicate transparently and develop connected systems. Many leading tech firms have adopted gRPC, such as Google, Netflix, Square, IBM, Cisco, & Dropbox.

What are the different ways TFLite reduces a model’s size to make it run on a mobile or embedded device?

Ans. We can reduce the size of a Tensorflow Model using the below mentioned methods: Freezing: Convert the variables stored in a checkpoint file of the SavedModel into constants stored directly in the model graph. This reduces the overall size of the model.

What is quantization-aware training, and why would you need it?

Ans. Quantization-aware training helps you train DNNs for lower precision INT8 deployment, without compromising on accuracy. This is achieved by modeling quantization errors during training which helps in maintaining accuracy as compared to FP16 or FP32.

What are model parallelism and data parallelism? Why is the latter generally recommended?

Ans. Data parallelism is when you use the same model for every thread, but feed it with different parts of the data; model parallelism is when you use the same data for every thread, but split the model among threads. Model parallelism is the process of splitting a model up between multiple devices or nodes (such as GPU-equipped instances) and creating an efficient pipeline to train the model across these devices to maximize GPU utilization.

When training a model across multiple servers, what distribution strategies can you use? How do you choose which one to use?

Ans. In distributed training the workload to train a model is split up and shared among multiple mini processors, called worker nodes. These worker nodes work in parallel to speed up model training.