Assignment 3

What is TensorFlow Extended (TFX)?

TensorFlow Extended (TFX) is an end-to-end platform for deploying production ML pipelines. A TFX pipeline is a sequence of components that includes modeling, training, serving inference, and managing deployments to online, native mobile, and JavaScript targets.

The paper [1] explains the various components of the TFX system and the corresponding basic functions. It can be said that the engineering quality of the TFX system itself is quite high, and it also meets many basic machine learning business needs. However, TFX itself is deeply bound with TensorFlow, and has very limited support for other machine learning frameworks such as Scikit-learn, PyTorch, etc. Then in the actual machine learning system, a common problem is the mixed use of multiple frameworks.

For example, a typical ML pipeline might be:

- Data analysis done by Pandas
- Data conversion done by NLTK/SpaCy
- TensorFlow, PyTorch, Scikit-learn trained model
- Deploy the model on the company's infra

This often results in the separation of various parts, making it difficult for the entire system to be effectively unified. Since TFX is completely based on TensorFlow, it cleverly avoids this problem.

However, a company with a lot of experience in machine learning systems needs to pay special attention to this issue when adopting TFX. TFX is a large and unified solution, but it is not necessarily a panacea for the company's existing complex machine learning system.

How can we use TFX to build a TensorFlow based pipeline?

The following steps should be followed using AirFlow with TFX

1. Setup your environment
2. Bring up initial pipeline skeleton
3. Dive into your data
4. Feature engineering
5. Training
6. Analyzing model performance
7. Ready for production

What are the limitations of Tensorflow (2.0)

It is a complete new design, APIs are not compatiable with Tensorflow 1.x. Thus the ecosystem of TF2.0 is almost zero compare to TF1.x.

Please elaborate on TensorFlow Performance Optimization

1. Use `tf.function`, it is a decorator function provided by Tensorflow 2.0 that converts regular python code to a callable Tensorflow graph function, which is usually more performant and python independent. It is used to create portable Tensorflow models.
2. Use `tf.data` optimize the data process pipline. For example, prefetching, parallelizing data extraction/transformation, caching, vectorizing mapping, reducing memory footprint

Reference

[1] A. N. Modi et al., “TFX: A TensorFlow-Based Production-Scale Machine Learning Platform,” 2017.