**TensorFlow Lite**

**Date:** 2024.10.04

**Points:**

**Processes:**

1. Collecting data
2. Labelling data
3. ML model
4. Cloud computing: deploy on cloud. Problem is the network latency (3-10 lb), it takes a lot of time. Storage resource and computing resource.
5. Inference on android using the new model.

**Edge devices:**

1. Cell phones
2. Microcontrollers
3. Wearable devices: smart watches

**Edge ML explosion:** low latency. Reduce the computing inference 30 ms to 20 ms, network latency can go up to second. Real-time. Network connectivity. No connectivity, new model will be better. User privacy. ML on cloud, send user data over network, can be intercepted.

Trying cosmetics using AR on YouTube, on device itself.

Google translates. Capture text and translate into the phone. Using edge computing.

**Keras -> Edge device Challenge:**

1. Limited compute power
2. Limited memory
3. Battery consumption
4. App size

**Solution:** TensorFlow Lite runs on every device.

**Workflow:**

1. Train a TF model. (Clean data, Preprocess data, create model, Iterate over multiple models, having best possible accuracy)
2. Convert to TensorFlow Lite format.
3. Deploy and run on edge device. Run inferences on the edge device.

A diagram of data flow

Description automatically generated

Converters translate to FlatBuffer files.

FlatBuffer is a cross-platform library.

Why not Protocol Buffers? Keras model or tensorflow model are protocal buffer format, but FB doesnot need unpacking steps.