Offline Compression of Convolutional Neural Networks on Edge Devices

CSE3000 Research Project by Simon Tulling | Supervision by Lydia Chen, Masoud Ghiassi & Bart Cox

24 June 2020

Background

- **Edge Devices:** Devices with limited resources.
- **Neural Networks:** Focused on accuracy over file size.
- Due to the size of the network, **inference** on an edge device can be **extremely slow.**

Question

- **How do we reduce the file size of neural networks on edge devices?**

Answer:

A **compression pipeline** that can run on an edge device and allows the user to **choose the trade-off between file size and accuracy.**

Why choose this way of compressing?

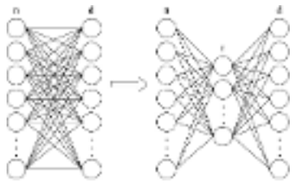
- Does not use **retraining**.
- Keeps the same **network architecture**. The resulting model uses the same architecture, but is just smaller.
- Capable of being run on a device with **limited resources**.

Compression Pipeline

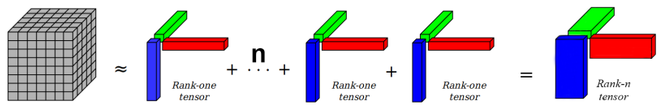
Big network goes in, small model comes out

But how does it work?

1. Provide the compression pipeline with a **model**.
2. Tell the pipeline which **trade-off** between file size and accuracy you want.
3. The model will try to **decompose all possible layers** in the model such that the specified ratio of the user is satisfied.
   a. Decompose **fully connected** layers using **singular value decomposition**, this basically splits the layer up in two smaller layers.



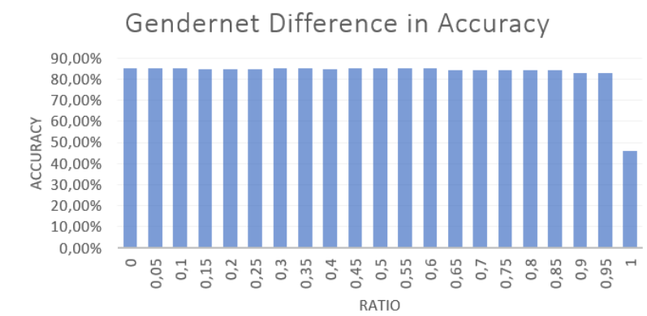
   b. Decompose **convolutional** layers using **canonical polyadic decomposition**, splitting one layer into 4 different ones.



4. Now the pipeline will **reconstruct the model** using these new layers.
5. That's it.

Results

Here are the results when the compression pipeline is used on the GenderNet network. GenderNet figures out the gender of a person given a photo of their face. We run the compression pipeline using different trade-off ratio's.



We can see that the accuracy stays the same up with a trade off of 0.95, making the model focus on decreasing the file size, rather than trying to optimize for accuracy. This ratio gives us a file size that is more than 5x smaller than the original.