Prakhar Ganesh · Follow
Jun 24, 2019 · 5 min read ★

Deep Learning — Model Optimization and Compression: Simplified

Take a peek into the domain of compression, pruning and quantization of state-of-the-art Machine Learning models

What's this?

The world around us is filled with Neural Networks and Deep Learning models doing *wonders*!! But these models are both computationally expensive and energy intensive. So expensive that people have actually started holding AI/ML accountable for their carbon emission and the numbers are ***not pretty!!***

Training a single AI model can emit as much carbon as five cars in their lifetimes
The artificial-intelligence industry is often compared to the oil industry: once mined and refined, data, like oil, can...
www.technologyreview.com

Another major reason why more researchers have turned towards Model compression is the difficulty in deploying these models on systems with limited hardware resources. While these models have been successful in making headlines and achieving extraordinary performances, they require the support of expensive, high speed GPUs to get them working, which limits their applications.

Being able to compress these highly complex models, transfer them to hardware devices and end their dependence on huge computational resources is one of the major goals of this domain. These kind of advances can help us incorporate AI into every small embedded system around us.

Why not just use GPU servers?

Yes *offcourse!!* With the internet giants like Google and Amazon offering computational services online, one does wonder if doing remote computations is the way to go. While people have started using these cloud services like a crutch for heavy computations, it does come with it's own set of problems.

One of the major issues with cloud computing is network connectivity. The system needs to be online at all times to work smoothly. But that cannot be always guaranteed and thus doing the computations locally is of extreme importance to systems that cannot afford any network delays.

Another issue with using these cloud services is sacrificing the "air gap". The air gap is a technical term used to represent systems that are not connected to the internet and thus cannot be breached remotely. Getting access to the data present in these configurations needs to be done physically, *Mission Impossible style*!! :P

For systems which are extremely protective regarding their privacy and security, giving up this "air gap" is not ideal and so they prefer local computations over cloud services.

But none of that actually affects me!!

That's what the majority of ML community believes but is not true!! If you are a beginner in ML looking to develop state-of-the-art models and are not bound by processing capacities, you might think that highly complex and Deep models are always the way to go.

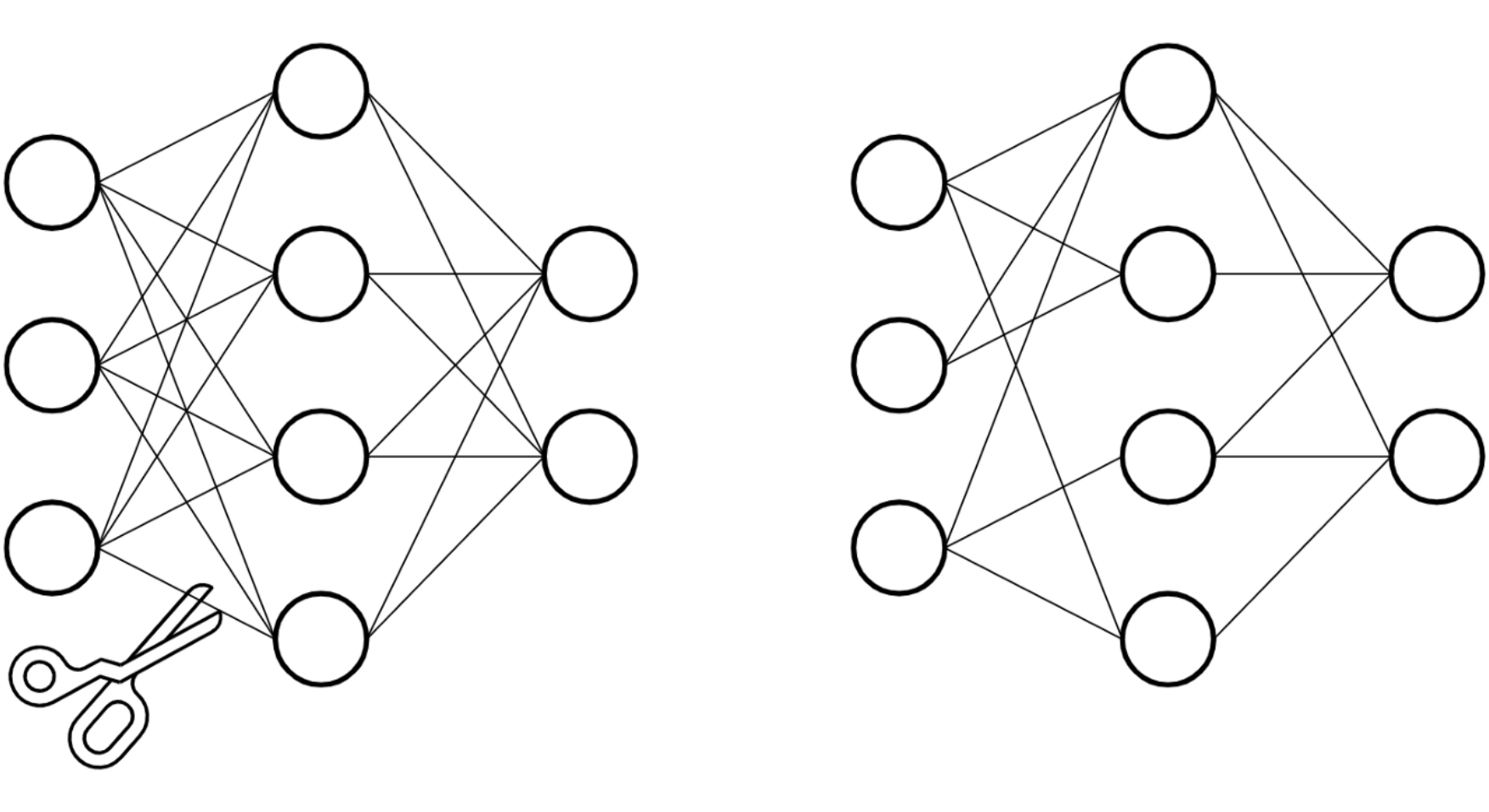
But that's a huge misconception. Highly complex and Deep models does not guarantee performance. Not to mention, these models can take hours or sometimes even days to train (even on GPUs). Research into pruning and quantization has shown that the connections that actually matter in the model are only a small percentage of the whole spider web!!

For example, famous ImageNet models like AlexNet and VGG-16 have been compressed to 40–50 times their original size, without any loss of (and actually a slight gain of) accuracy. This dramatically increases their inference speed and the ease with which they can adapted across various devices.

Enough with the convincing, let's talk about the techniques involved!!

Model compression can be divided into two broad categories,

Pruning : Removing redundant connections present in the architecture. Pruning involves cutting out unimportant weights (which are usually defined as weights with small absolute value).



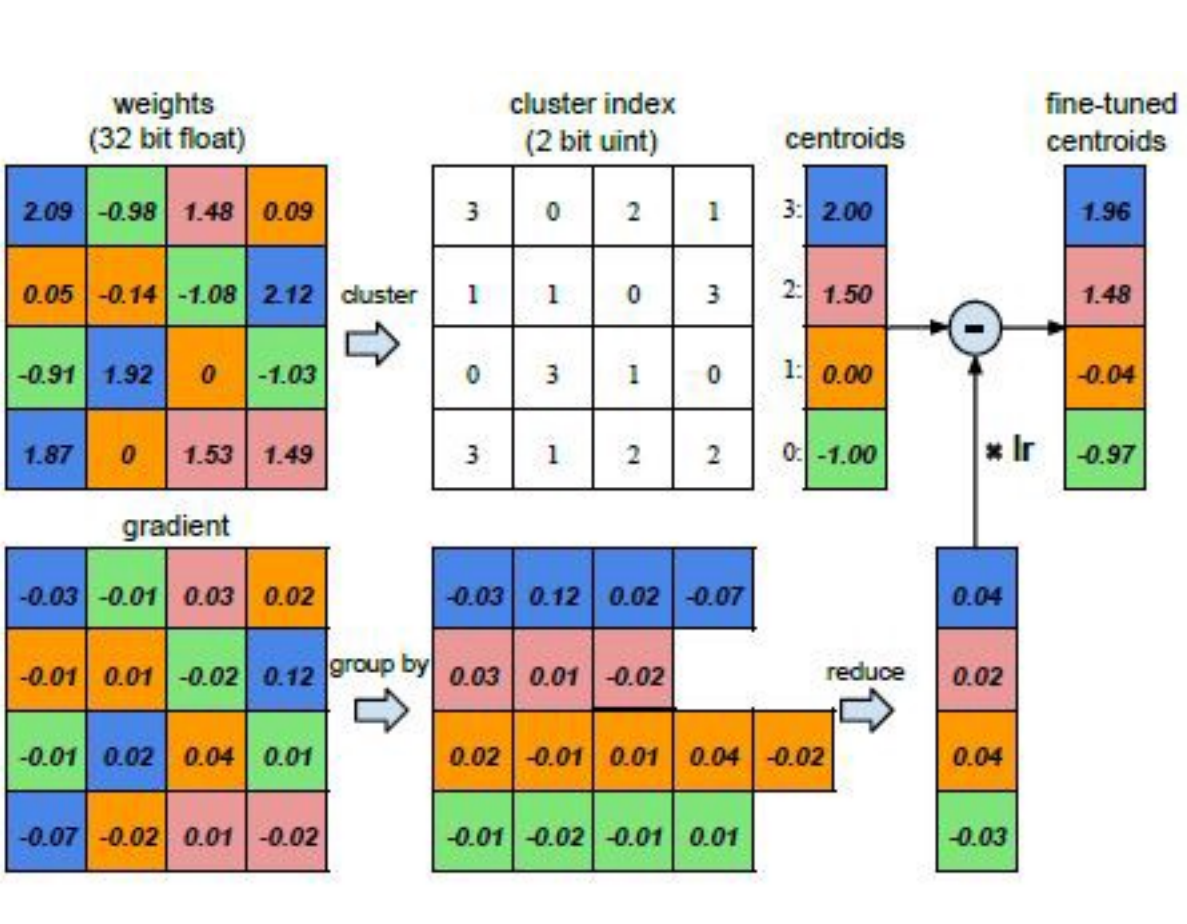
Before pruning　　　After pruning

Obviously the new model formed will have lower accuracy since the model was actually trained for the original connections. That is why the model is fine tuned after pruning to regain the accuracy. It is noted that fully connected layers and CNNs can usually go upto **90% sparsity** without losing any accuracy.

Quantization : Quantization involves bundling weights together by clustering them or rounding them off so that the same number of connections can be represented using lesser amount of memory.

Quantization by doing clustering/bundling and thus using lesser number of distinct float values to represent more number of features is one of the most common techniques. Another common technique that forms the skeleton for a lot of quantization methods is converting floating point weights to fixed point representation by rounding off.

Again, as it was with pruning, we need to fine tune the model after quantization. The important point here is that the property that was given to the weights while quantization should be maintained through the fine tuning too. That is why specific ways of fine tuning are used which are tailored to match the quantization method.



Look at the image above for an example of quantization by clustering. Weights of same color are clustered together and represented by their centroid. This decreases the amount of data required to represent these weights. Earlier it required 32bits*16 = 512 bits to represent them. Now it will only take 32bits*4 + 2bits*16 = 160 bits to represent them. During the fine tuning, the gradient for all the weights which belong to the same color are summed and then subtracted from the centroid. This makes sure that the clustering made during quantization is maintained through the fine tuning.

What's next?

Deep Learning model pruning and quantization are relatively new fields. While there have been significant success in this field, it still has a long way to go. The next focus of the domain should be creating open source and easily accessible pipelines for transferring common Deep Learning models to embedded systems like FPGAs.

· · ·

This blog is a part of an effort to create simplified introductions to the field of Machine Learning. Follow the complete series here

Machine Learning : Simplified
Know it before you dive in
towardsdatascience.com

Or simply read the next blog in the series

High Frequency Trading (HFT) with AI : Simplified
Take a peek into the ever-competing world of HFTs and how is AI becoming a part of it.
towardsdatascience.com

· · ·

References

[1] Han, Song, Huizi Mao, and William J. Dally. "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding." arXiv preprint arXiv:1510.00149 (2015).
[2] Jia, Haipeng, et al. "DropPruning for Model Compression." arXiv preprint arXiv:1812.02035 (2018).
[3] Wang, Shuo, et al. "C-lstm: Enabling efficient lstm using structured compression techniques on fpgas." Proceedings of the 2018 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays. ACM, 2018.