Lightweight Neural Networks (Pruning Algorithm)

Abd Elfattah Fahym, Youness Iziria, Aymane Cherki

May 26, 2023

The application of pruning algorithms in computer vision is explored, focusing on image stitching and photo-based 3D modeling. Pruning involves selectively removing connections and neurons from neural networks to reduce storage requirements, energy consumption, and address network complexity and overfitting. Limitations associated with pruning algorithms include potential loss of precision, sensitivity to initialization, and the need for retraining. Mobile applications can benefit from pruning algorithms, as exemplified by the MobileNets framework, which incorporates depthwise separable convolutions and pruning techniques to reduce model size and computational complexity while preserving accuracy. The complementary nature of pruning algorithms in optimizing computer vision tasks is emphasized.

Introduction

Computer vision is a discipline that involves developing algorithms to enable computers to understand and interpret images and videos. However, these algorithms are often resource-intensive, particularly in terms of computational power and energy. This poses a problem for mobile devices such as smartphones and tablets, which have significant energy limitations.

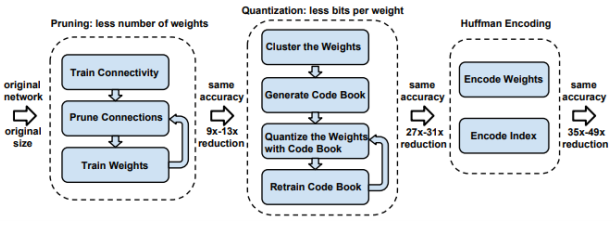
Indeed, these devices have a limited battery that needs to power not only the main processor but also the camera and other sensors required for computer vision. Additionally, these devices need to be compact enough for easy portability, which limits the size of the battery and consequently the battery life.

That is why energy efficiency is a major concern in computer vision for mobile devices. Researchers are therefore working on developing techniques to reduce energy consumption while maintaining sufficient processing quality for the intended applications. These techniques can include algorithm optimization, reducing the number of steps in the pipeline, or utilizing specialized hardware to accelerate certain tasks.

Pruning algorithm

Computer vision techniques were increasingly being used in computer graphics to create image-based models of real-world objects, to create visual effects, and to merge realworld imagery using computational photography techniques. Our decision to focus on the applications of computer vision to fun problems such as image stitching and photo-based 3D modeling from personal photos. [4]

The example of such neural network , Pruning algorithm, which is the process of removing connections (and neurons) from a neural network while preserving its functionality, the goal is to reduce the storage and energy required to run inference on such large networks so they can be deployed on mobile devices, it can also can reduce network complexity and over-fitting, As shown on the left side of Figure 1, we start by learning the connectivity via normal network training. Next, we prune the small-weight connections: all connections with weights below a threshold are removed from the network [1]



Also, Pruning typically proceeds by training the original network, removing connections, and further

fine-tuning. [2]

Experience of Pruning algorithm

We test our pruning scheme on the large-scale ImageNet classification task. In the first experiment, we begin with a trained CaffeNet implementation of AlexNet with 79.2% top-5 validation accuracy. Between pruning iterations, we fine-tune with learning rate 10^{-4} , momentum 0.9, weight decay 10^{-4}, batch size 32, and drop-out 50%. Using a subset of 5000 training images, we compute oracle-abs and Spearman's rank correlation with the criteria, as shown in Table 1. Pruning traces are illustrated in Fig.1 .[3]

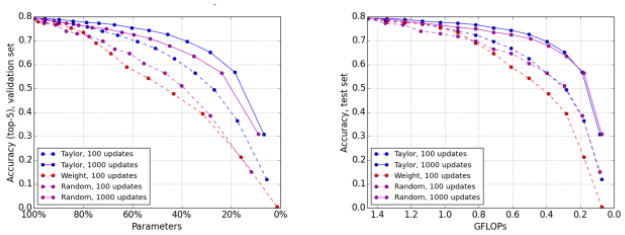


Figure 1: *Pruning of AlexNet on Imagenet with varying number of updates between pruning iterations.*

The limitations of the pruning algorithm

The pruning algorithm has certain limitations, including loss of precision, sensitivity to initialization, the need for retraining the model, reduced interpretability, dependence on the model and data, complexity of pruning techniques, and its impact on online learning. It is important to consider these limitations when using the pruning algorithm and to choose appropriate parameters and techniques based on the specific use case.

Application of Pruning algorithm in Mobile application

One example of pruning algorithm in mobile application is the work done by Sze et al. (2017) where they proposed a framework called "MobileNets" which utilizes depthwise separable convolutions for mobile vision applications. Depthwise separable convolutions separate the spatial and channelwise convolutions into two separate layers, which greatly reduces the number of parameters and computational complexity of the neural network.

Additionally, the MobileNets framework also utilizes pruning to remove unnecessary channels and parameters, further reducing the size and computational cost of the model. The authors showed that their framework achieved state-of-the-art accuracy on several mobile vision benchmarks while being up to 32 times smaller than other state-of-the-art models. [5]

Conclusion

The Configurable and Reversible Imaging Pipeline (CRIP) that proposed in the article "Reconfiguring the Imaging Pipeline for Computer Vision" and the pruning algorithm are distinct yet complementary concepts in the field of imaging and data processing. The CRIP enables the configuration and optimization of the image processing pipeline, while the pruning algorithm reduces the complexity of the models used in the pipeline. They can be used together to customize the pipeline and optimize performance while reducing model size and accelerating computations.

References

[1] S. Han. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2016.

[2] M. Carbin J. Frankle. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2019.

[3] S. Karras T. Aila J. Kautz P. Molchanov, P. Tyree. Pruning convolutional neural networks for resource efficient inference. *arXiv preprint arXiv:1611.06440*, 2019.

[4] Szeliski. Computer vision:algorithms and applications. *Springer Science and Business Media*, 2010.

[5] H. Yang J. Emer V. Sze, Y. Chen. Efficient processing of deep neural networks: A tutorial and survey. *Proceedings of the IEEE, 105(12), 2295-2329*, 2017.