Pruning 기반 알고리즘 경량화 기법의 성능 최적화 연구 동향

김서연 , 황원준*
아주대학교 소프트웨어학과
e-mail : wjhwang@ajou.ac.kr

A Review on Performance Optimization of Pruning-based Algorithm Lightweight Technique

Seoyeon Kim and Wonjun Hwang
dept. of Software, Ajou University

Abstract

Recently, lightweighting of AI models has emerged as an essential element in mobile and embedded systems, and pruning-based algorithms are at the center of this. In this paper, we analyze performance optimization trends of CNN models through structural pruning techniques such as filter removal. We also introduce the latest research on improving performance in complex architectures, such as NAS and automated pruning, and discuss the potential for real-time applications through energy savings and inference time optimization.

I. 서론

임베디드 시스템이나 모바일 환경에서 모델을 배포하는 것은 제한된 자원으로 인해 어려움이 있으며, CNN모델의 연산량(FLOPs)이 증가하면 추론 속도이 길어져 실시간 성능이 중요한 애플리케이션에 부적합할 수 있다. 이를 해결하기 위해 Pruning과 같은 모델 압축 기법이 중요한 역할을 한다. Pruning은 크게 unstructured pruning과 structured pruning으로 나눌 수 있는데, unstructured pruning은 신경망의 가중치를 비구조적으로 제거하여 압축을 수행한다. 이 방식은 비정형적인 희소성을 유발하며, 실제 연산 속도 향상을 위해서는 특수 하드웨어나 라이브러리가 필요하다. 반면, structured pruning은 신경망의 필터를 제거하여, 표준 하드웨어에서도 효율적인 가속과 압축을 제공한다.

CNN 모델은 여러 필터와 피처 채널로 인해 redundancy가 많다. Pruning Filters for Efficient ConvNets(ICLR 2017)에서는 필터를 제거하는 structured pruning 기법을 제안하여, 성능 손실 없이 모델을 압축할 수 있는 방법을 소개했다. 필터 제거를 통해 연산량을 줄이고, one-shot pruning과 retraining을 통해 네트워크 깊이에 따른 pruning 시간을 단축할 수 있다는 사실을 실험적으로 보여주었으며, 특히 ResNet에서는 성능 저하 없이 FLOPs를 약 30% 감소시킨 결과를 보여주었다.

II. 본론

본 장에서는 pruning 및 filter pruning을 위해 기초를 마련한 연구에 대해 소개하고, Pruning Filters for Efficient ConvNets(ICLR 2017)에서 제안하는 방식에 대해 살펴본다. 또한 전체적인 내용은 (Li et al., 2017)를 토대로 하여 다른 structured pruning 방법과 비교해보며 structured pruning의 전체적인

미래 방향성에 초점을 맞추어서 기술한다.

2.1 Background on Structured Pruning

Structured Pruning은 filter, channel, layer 같은 구조적 단위를 제거하여 모델 크기를 줄이고, 불필요한 연산을 없애 추론 속도를 향상시키는 기법이다. Unstructured Pruning은 가중치만 제거하여 모델 구조를 그대로 유지하지만, 연산 속도 향상에는 한계가 있다. 반면 Structured pruning은 필터나 채널을 제거해 표준 하드웨어에서도 쉽게 가속화가 가능하다.

역사적으로는 Optimal Brain Damage(1989)와 Optimal Brain Surgeon(1993)이 가중치 pruning 기법을 제안해 모델 성능을 유지하면서 파라미터 수를 줄이는 방법을 제시하였다. 이후 연구들은 필터의 수를 변경하지 않고 가중치 행렬을 근사하거나, FFT 기반 합성곱 및 Winograd 알고리즘으로 연산 속도를 개선하는 방법을 발전시켰다. 또한, Anwar(2015)는 가중치 중요도를 여러 단계로 평가하여 particle filtering을 통해 pruning 후보를 찾았고, Polyak&Wolf(2015)는 자주 활성화되지 않는 피처맵을 제거하는 방법을 연구하였다. 이러한 접근법들은 CNN에서 필터를 제거하여 연산량을 줄이는 네트워크 차원의 솔루션을 제안하였다.

2.2 Review on ICLR 2017 paper

Filter pruning은 모델의 구조를 유지하면서 불필요한 연산을 줄여, 추론 속도와 모델 크기를 동시에 줄이는 기법이다. L1-norm을 사용해 각 필터의 중요도를 측정하고, 중요도가 낮은 필터를 제거해 연산량을 줄인다. L1-norm은 필터의 절대값 합을 측정하는데, 계산이 단순하고 효율적일 뿐만 아니라, 필터가 작을수록 그에 해당하는 출력 활성화 또한 작을 것이라는 가정이 적용된다. One-shot Pruning은 여러 layer에서 filter를 동시에 제거하여 재학습 시간을 단축할 수 있다. 예를 들어, VGG-16에서는 필터의 64%정도를 제거해도 성능 저하가 거의 없었으며, ResNet에서는 FLOPs를 30%까지 줄일 수 있었다. 하지만 layer간 상호 의존성을 고려하지 않으며, L1-norm이 작다고 필터 중요도 또한 낮은 것으로 단정지을 수 없다는 문제점이 있다. 이로 인해 여러 layer를 동시에 pruning할 경우 정확도 손실을 예측하기 어렵다. 또한 One-shot pruning은 재학습 시간을 줄일 수 있지만, 여러 레이어를 동시에 제거할 때 각 레이어의 상호작용을 충분히 고려하지 못할 수 있다.

Ⅲ. 비교분석

L1-norm을 기반으로 필터를 제거하는 structured pruning 방식 이후, 모델 성능 유지와 동시에 효율성 증가를 목표로 한 다양한 pruning 방법들이 제안되어 왔다. 그중 Soft Filter Pruning(SFP), Geometric Median based Pruning(FPGM), 그리고 Group Normalization based Pruning의 특징을 살펴보고, 이 방법들과 ICLR2017에서 제안한 filter pruning 간의 차이를 살펴본다.

4.1 Soft Filter Pruning (SFP)

전통적인 필터 제거 방법과는 달리, SFP는 필터의 가중치를 약화시킨 후 다시 복구하는 방법을 사용한다. 훈련 중 pruning과 가중치 복구를 반복적으로 적용하여 필터가 완전히 제거되지 않고 약화된 상태로 유지되었다가 다시 강화된다. 이를 통해 성능을 유지하면서도 불필요한 필터를 점진적으로 줄여 모델을 경량화 한다. 그러나 정확한 pruning 대상 필터를 선정하는 데 시간이 소요될 수 있다.

4.2 Geometric Median based Pruning (FPGM)

GMP는 필터 간의 유사성을 기반으로 필터를 제거한다. Geometric median은 다차원 공간에서 다른 점들과의 거리합을 최소화하는 점을 의미하는데. 여러 필터를 다차원 공간에 배치한 후 중심에서 가장 멀리 떨어진 필터를 제거한다. 이는 필터간 상호관계를 고려하여 중요한 필터는 유지하고, 덜 중요한 필터를 선택적으로 제거하는 방식이다. 그러나 제거할 필터를 찾기 위한 계산 과정에서 계산 비용이 증가할 수 있고, 고차원 공간에서 필터 간 유사성을 분석하는 경우 시간이 많이 소요될 수 있다.

4.3 Group Normalization based Pruning

Group Normalization은 필터나 채널을 그룹화한 후, 정규화를 적용하여 불필요한 그룹을 제거하는 방식이다. 개별 가중치가 아닌 필터나 채널을 그룹단위로 판단하고, 중요도가 낮은 그룹을 제거하여 성능 저하 없이 모델을 압축할 수 있다. 그러나, 그룹화를 하게 되는 경우 유연성이 부족해지는 문제가 생길 수 있으며, 모델에 따라 최적의 그룹화를 찾는 과정이 어려울 수 있다.

이 세 가지 방법과 비교해보았을 때, ICLR 2017의 filter pruning은 단순하고 직관적이라는 장점이 있다. L1-norm을 기반으로 한 필터 중요도 평가를 통해 계산 비용이 적고 구현이 용이하며, 복잡한 그룹화나 정규화 과정 없이 다양한 네트워크 구조에 쉽게 적용할 수 있다. 그러나 ResNet과 같은 복잡한 모델에서는 스킵 연결과 필터 간 상호작용으로 인해

성능 저하의 위험이 생길 수도 있다.

Ⅳ. 결론 및 향후 연구 방향

Structured Pruning은 하드웨어 가속 및 리소스 제한 환경에서 매우 유리한 기법이다.

5.1 연산 최적화

필터나 채널 단위로 불필요한 연산을 제거하여 GPU나 FPGA에서 효율적인 병렬 처리가 가능하고, 메모리 접근 및 연산량을 감소시킬 수 있다.

5.2 메모리 절약

모델 크기 감소로 메모리 사용량이 줄어들어, 임베디드 시스템 및 모바일 기기와 같은 자원 제한 환경에서 배포가 용이하다.

5.3 추론 시간 단축

경량화된 모델 구조로 인하여 추론 속도가 개선되어 실시간 애플리케이션에서 효과적으로 활용할 수 있다.

그러나 트랜스포머 및 NAS 기반 모델은 파라미터가 많고 복잡도가 높기에, Pruning이 성능에 미치는 영향이 더욱 크다. ICLR 2017에서 제안하는 방법은 주로 CNN에 적합하게 설계되었는데, CNN은 필터와 같은 구조적 요소가 명확히 정의되어 있으므로 제안하는 방식이 쉽게 적용된다. 반면 트랜스포머 및 NAS 기반 모델은 필터 대신 자연어 처리나 시계열 분석에 적합한 attention 메커니즘을 많이 사용하기에 이러한 구조는 단순히 필터 단위로 제거할 수 없다.

실제 응용에서 경량화된 모델의 중요성이 커지며, pruning은 모바일 및 임베디드 시스템에 적합한 AI 모델 배포의 핵심 기술이 되었다. ICLR 2017의 "Pruning Filters for Efficient ConvNets"은 필터 단위의 structured pruning을 제안하며, 가중치를 제거하는 대신 필터를 제거함으로써 연산 효율성을 높이고, 하드웨어 가속기에서 실제 성능 향상을 가능하게 했다. 이후 이 기법을 바탕으로 자동화된 프루닝과 다양한 네트워크 구조에 대한 연구가 촉진되었다. 특히, 프랜스포머와 NAS 기반 모델에서 필터 단위의 pruning 대신, attention 메커니즘 혹은 파라미터 간 상호작용을 기반으로 한 새로운 방식이 필요할 수 있으며, 향후 연구에서는 이러한 구조를 고려한 맞춤형 pruning 기법이 중요한 역할을 할 것이다.

Ⅴ. 사사의 글

본 연구는 2024년 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학사업의 연구결과로 수행되었음(2022-0-01077)

참고문헌

[1] He, Y., Zhang, X., & Sun, J. (2017). *Channel Pruning for Accelerating Very Deep Neural Networks.* Proceedings of IEEE International Conference on Computer Vision (ICCV).

[2] Li, H., Kadav, A., Durdanovic, I., Samet, H., & Graf, H. P. (2017). *Pruning Filters for Efficient ConvNets.* International Conference on Learning Representations (ICLR).

[3] Liu, Z., Sun, M., Zhou, T., Huang, G., & Darrell, T. (2019). *Rethinking the Value of Network Pruning.* International Conference on Learning Representations (ICLR).

[4] Anwar, S., Hwang, K., & Sung, W. (2015). *Structured Pruning of Deep Convolutional Neural Networks.* arXiv preprint arXiv:1512.08571.

[5] Han, S., Pool, J., Tran, J., & Dally, W. (2015). *Learning both Weights and Connections for Efficient Neural Networks.* Advances in Neural Information Processing Systems (NeurIPS).