Weight Recycle을 이용한 모델 압축

Deep Neural Network Model Compression using Weight Recycle

요 약

최근 딥러닝은 많은 분야에서 좋은 성과를 내고 있지만, 성능에만 초점을 맞춘 나머지 하드웨어적 제한이 있는 환경에 적용하기 힘들다는 문제점이 발생한다. 이를 해결하기 위해 고안된 것이 모델 압축이다. 모델 압축 방법은 새롭게 제안한 Weight Recycle 기법을 사용한다. Weight Recycle 기법이란 convolution layer의 크기를 줄이고 회전시킨 후 합쳐 줄인 부분의 weight를 대체하는 모델 압축 기법이다. 본 논문에서는 Weight recycle 모델 압축 기법을 사용하여 큰 성능의 하락 없이 convolution layer의 weight의 크기를 고정적으로 1/4로 줄일 수 있었다.

1. 서론

최근 컴퓨터 비전 분야에서 딥러닝이 가져온 성과는 대단하다. 하지만 이전까지의 연구들은 성능에만 초점을 맞추어 진행되었기 때문에 상용화에 있어서는 아직까지 여러 가지 문제들이 존재한다. 하나의 예로, 일반적으로 딥러닝 모델이 깊은 신경망으로 이루어질수록 좋은 성능을 보이는데, 깊은 신경망일수록 모델의 매개변수 수가 많아 저장 공간을 많이 차지하게 된다. 이와 같은 이유로 딥러닝 기반 방법을 하드웨어적 제한이 있는 환경에 적용하기 힘들다. 이를 해결하려 한다는 관점에서 모델 압축은 매우 중요한 연구 주제다.

본 논문에서는 Classification 문제에 새로 제안된 Weight Recycle 모델 압축 기법을 사용하여 딥러닝 모델의 Weight 압축에 대해 연구하려고 한다.

2. 관련 연구

2.1. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding[1]

3가지의 Model compressing 기법에 관해 연구한 논문이다. Pruning, Quantization, Huffman coding 이 세 가지 기법에 관해 연구되었다.

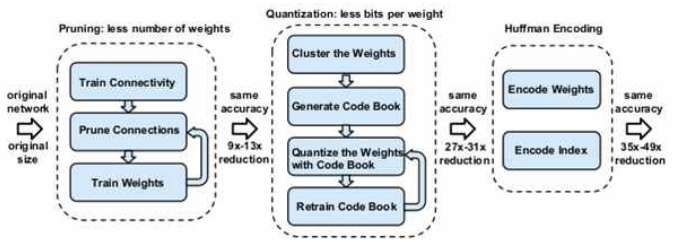

Figure 1: The three stage compression pipeline: pruning, quantization and Huffman coding. Pruning reduces the number of weights by 10×, while quantization further improves the compression rate: between 27× and 31×. Huffman coding gives more compression: between 35× and 49×. The compression rate already included the meta-data for sparse representation. The compression scheme doesn't incur any accuracy loss.

그림1. Deep Compression Pipeline

Pruning은 모델의 Weight들의 값을 조사한 다음 threshold를 정하고 그보다 0에 가까운 Weight들은 0으로 바꾼 뒤 더이상 계산, 학습시키지 않는

방법이다. Quantization은 Pruning 후 남은 모델의 weight들을 어느 정도의 기준으로 묶은 다음에 한 묶음에 있는 weight 값들을 한가지 값으로 통일시키는 방법이다. Huffman coding은 앞서 Pruning과 Quantization을 한 후에 많이 쓰이는 값부터 순서대로 작은 bit를 할당하는 방법이다.

2.2. Constructing Fast Network through Deconstruction of Convolution[2]

Active Shift Layer를 사용한 모델 압축에 관해 연구된 논문이다.

Active Shift Layer이란 Weight의 Channel을 Shift 시켜 1x1 convolution layer로도 그보다 큰 convolution layer의 역할을 할 수 있게 만들어주는 기법이다.

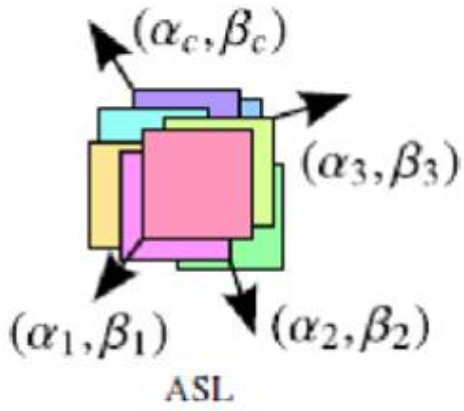

그림2. Active Shift Layer

3. 제안 방법

본 논문에서 새로운 모델 압축 기법인 Weight Recycle 기법을 제안하였다. Weight Recycle 기법이란 기존의 Convolution layer의 weight를 1/4로 압축시킬 수 있는 기법이다. 기존의 convolution layer의 output channel 크기를 기준으로 1/4로 줄인 후 90도, 180도, 270도로 각각 회전시킨다. 그 후 output channel의 차원을 기준으로 모두 합친다.

이렇게 되면 기존의 convolution layer의 weight를 1/4만 사용하여 기존의 weight 크기를 사용한 원래 성능을 보존하는 기법이다.

기존의 압축 기법들은 성능과 압축률이 계속 바뀌는 데 비해 본 논문의 기법을 사용하면 고정적으로 convolution layer의 weight 압축률을 1/4로 줄일 수 있고 또한 weight의 회전으로 인한 data augmentation 효과도 얻을 수 있을 것이다.

weight를 1/4로 고정적으로 압축함으로 인해 필요 없는 feature들을 제거하고 필요한 feature들을 기존의 convolution layer보다 4배 더 집중적으로 학습할 수 있을 것이다.

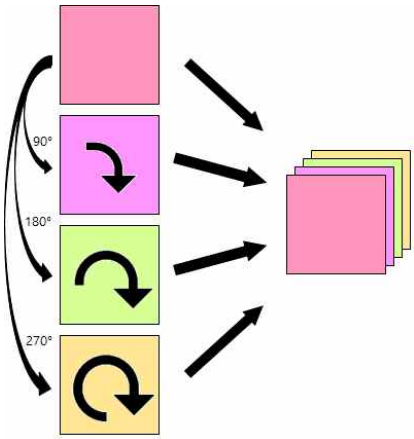

그림3. Weight Recycle

4. 실험 결과 분석 및 토론

본 논문에서 실험에 사용한 모델은 LeNet5[3], ResNet18[4], ResNet34, ResNet50, VGG19[5]이다. Loss function은 CrossEntropyLoss를 사용하였고 optimizer로는 Adam을 사용하였다. Epochs 수는 200으로 고정하였고 학습률에 변화를 주며 실험을 진행하였다. 실험의 결과는 아래의 표에 나와 있다. 보면 Baseline accuracy보다 Weight Recycle 모델 압축 기법을 적용한 결과가 더 좋게 나오는 경우가 있다. 이는 Weight Recycle 모델 압축 기법을 사용하면 Weight가 회전하면서 입력이 회전하는 듯한 효과도 생기게 된다. 이로 인해 data augmentation 효과가 생기게 되어서 Weight Recycle 모델 압축 기법이 더 성능이 좋게 나오는 결과가 생긴다고 예상한다.

Model_lr	Baseline	Ours
LeNet5_0.001	74.316	73.944
LeNet5_0.0008	**74.400**	73.656
ResNet18_0.001	95.968	88.158
ResNet18_0.0005	**96.912**	90.166
ResNet18_0.0001	95.822	95.890
ResNet34_0.001	73.944	87.678
ResNet34_0.0005	96.426	91.064
ResNet34_0.0001	**96.692**	95.200
ResNet50_0.001	73.944	87.280
ResNet50_0.0005	96.426	92.352
ResNet50_0.0001	98.692	**98.960**
VGG19_0.001	93.118	84.906
VGG19_0.0005	**96.542**	87.962

표1. 본 실험에서 Dataset은 CIFAR 10을 사용하였다.

LeNet5와 ResNet의 실험 결과를 보면 기존의 성능에서 크게 하락하지 않은 채 Weight의 크기를 1/4로 줄이는 대에 성공하였다. 하지만 VGG19의 경우 기존의 성능과 큰 차이가 나는 결과가 나왔다. 이는 VGG의 경우 다른 모델보다 Weight의 수가 많아 같은 1/4이라도 다른 모델에 비해 많은 수의 Weight가 압축이 되고 그 부분을 Weight Recycle 모델 압축 기법의 장점들이 메꿔주지 못하기 때문이라고 생각한다.

5. 결론

본 논문에서는 새롭게 제안된 Weight Recycle 모델 압축 기법을 사용하여 여러 딥러닝 모델의 압축에 도전하였다.
대부분의 모델들은 큰 성능의 하락 없이 Weight의 수를 1/4로 줄이는 데 성공하였다. 하지만 VGG19를 대상으로 실험한 결과 큰 성능의 하락이 일어났다.
위의 결과들로 적정 수의 Weight 수는 Weight Recycle 모델 압축 기법으로 Weight의 수를 1/4로 줄여도 Data augmentation 등의 이점으로 보완할 수가 있지만, Weight의 수가 많아지게 되면 이러한 이점들로 보완할 수 없게 된다는 것을 추측할 수가 있다.
이 추측들을 추후의 실험을 통하여 증명해 낼 필요가 있고 또한 더욱 여러 모델들을 대상으로 한 실험과 Weight Recycle 모델 압축 기법의 성능 개선 또한 추후의 연구를 진행할 것이다.

참고 문헌

[1] Song Han, Huizi Mao, William J. Dally, “Deep Compression: Compressiong Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding”, In International Conference on Learning Representations (ICLR), 2016.

[2] Yunho Jeon, Junmo Kim, “Constructing Fast Network through Deconstruction of Convolution” In Neural Information Processing Systems(NIPS), 2018.

[3] Yann LeCun Leon Bottou Yoshua Bengio and Patrick Haffner, “GradientBased Learning Applied to Document Recognition”, 1998.

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, “Deep Residual Learning for Image Recognition”, 2015

[5] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015.