비디오 코덱 기반의 딥러닝 가중치 압축

김승환[1], 박은수[1], 굴람 무즈타바[2], 류은석[1]

성균관대학교[1], 가천대학교[2]

whitekomani@skku.edu, espark804@skku.edu, *mujtaba@gc.gachon.ac.kr, esryu@skku.edu

Video Codec Based Deep Learning Weight Compression

Kim, SeungHwan Park, Eun-Soo Ghulam Mujtaba[2] Ryu, Eun-Seok

Sungkyunkwan Univ.[1] , Gachon Univ.[2]

요 약

최근 인공신경망 모델은 다양한 분야에서 뛰어난 결과를 나타냈다. 하지만 모델의 성능과 함께 복잡도와 크기가 높아지면서 연산성능, 네트워크 환경 등 한계가 존재하는 휴대용 장비에서는 이러한 네트워크를 학습하는 것은 부적절하다. 학습된 가중치를 사용하는 과정에서 휴대용 장비의 제한된 자원을 효율적으로 활용하기 위해서는 가중치의 압축이 필수적이다. 이 논문에서는 비디오 코덱을 통해 딥러닝 모델의 가중치를 압축하는 방법을 소개한다.

1. 서 론

최근 컴퓨터 비전 분야에서 인공신경망 모델은 뛰어난 성능을 보이는 기법으로 다양한 분야에서 사용되고 있다. 특히 Convolution Neural Network(CNN) 모델은 이미지 인식 분야에서 인간의 정확도를 뛰어넘는 결과를 보이며 다양하게 이용되고 있다. 하지만 대부분 모델은 높은 연산 복잡도와 가중치가 크기 때문에 모바일 기기에서 사용하기에는 부적절하다. 예를 들어 VGGNet은 이미지 분류에서 뛰어난 성능을 나타냈지만 500MB가 넘는 크기를 가지고 있다[1]. 모바일 환경에서 인터넷을 통해 이러한 모델을 전송받는 것은 어려움이 있다. 따라서 모바일 기기에 모델을 전송하기 위해서는 모델을 압축하는 과정이 필요하다.

모델을 압축하는 방법에 대해서는 다양한 방법들이 발표되었다. 모델 압축의 목표는 정확도를 많이 감소시키지 않으면서 모델의 크기를 줄이는 것으로 대표적인 방법은 양자화와 Pruning이다. 양자화는 모델을 구성하는 각 가중치를 표현하는 bit의 수를 줄이는 방법이다. Pruning은 훈련된 모델에서 중요하지 않은 가중치를 잘라내는 방법으로 가중치의 수를 줄이는 방법이다. 본 논문에서는 양자화를 통해 가중치를 압축하는 과정에서 High Efficiency Video Coding(HEVC)[2]을 사용하는 방법에 대해 제시한다.

2. 관련연구

Weight Pruning과 양자화는 다양한 연구에서 그 효과가 검증되었다. Weight Pruning은 가중치의 수를 줄이는 효과와 함께 수렴속도 또한 빠르게 하는 효과가 있다[3]. 양자화는 GPU를 사용할 수 없는 환경이나 모바일 기기에서도 빠른 처리속도를 달성하며 모델의 크기를 압축했다[4]. 본 절에서는 이 두 가지 기법을 바탕으로 모델을 압축하는 방법들을 소개한다.

2.1 Weight Pruning

Weight Pruning은 불필요하거나 중복된 연결을 제거하는 방법으로 [5]에서 처음 소개되었다. 전체 모델을 학습시킨 후, Pruning을 수행한 후, 재학습을 하는 과정을 발표하며 가중치를 초기화하고 학습하는 것은 성능이 좋지 못하다는 문제점을 발표했다[6]. 이런 문제점은 Pruning 전의 모델의 초기값을 다시 사용하는 방법을 통해 해결되었다[7].

Weight Pruning과 양자화를 모두 이용하여 모델을 압축한 연구도 발표되었다. [8]는 Weight Pruning을 통해 불필요한 연결을 제거한 후, 가중치를 양자화하고 Huffman Coding을 통해 양자화된 가중치를 압축하는 방법을 제시했다. 그 결과 크기는 49배 압축되었지만, 정확도는 유지되었다.

2.2 가중치 양자화

모델의 각 가중치는 32bit float로 표현된다. 가중치의 비트를 줄이는 양자화에 대한 연구도 발표되었다. [9]는 Fully-connected Layer와 Convolution Layer 모두 k-means clustering을 통하여 가중치를 양자화하였다. 그 결과 압축률 23배 이상을 달성하면서 정확도의 감소는 3% 정도로 나타났다.

본 논문에서는 양자화된 가중치를 압축하는 과정을 비디오 압축에 사용하는 HEVC로 대체하는 방법을 제시한다. 이 방법을 VGGNet[1]에 대해 21배 압축하며 ImageNet 데이터셋으로 알려진 ILSVRC2012에 대한 실험 결과 정확도의 감소는 1% 이하로 나타났다.

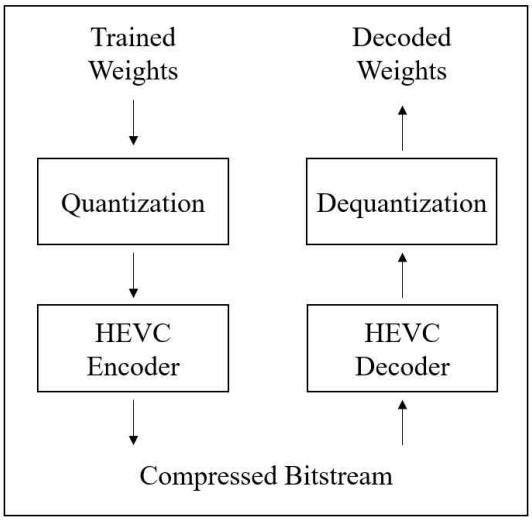

그림 1 Video Encoder 기반의 압축 개념도

3. Video Encoder 기반의 가중치 압축

본 논문에서 제시하는 가중치 압축방법은 그림 1과 같이 가중치를 8bit 정수로 양자화한 후 이를 HEVC Encoder를 통해 압축한다. 본 장에서는 양자화 방법과 압축방법을 설명한다.

가중치는 가중치 행렬의 Scale Factor S를 계산하고 가중치를 8bit 정수로 양자화한다. S는 가중치 행렬당 하나씩 할당되는 값으로 양자화된 가중치를 복원할 때 사용한다. 최적의 양자화 단위를 찾기 위한 다양한 연구가 발표되었지만[9], 본 연구에서는 양자화된 가중치에 대한 비디오 인코더의 압축 효과를 보기 위해 연산 복잡도가 낮은 간단한 양자화 방법을 사용한다.

$$S = \frac{\max(|W|)}{127} \quad (1)$$

$$W_q = round(\frac{W}{S}) \quad (2)$$

식 (1)과 (2)에서 행렬 W는 양자화 전의 32bit 실수형 가중치를, W_q는 양자화된 가중치 행렬을 나타낸다.

양자화된 가중치는 8bit 정수의 행렬로 많은 이미지 형식에서 사용하는 형태이다. 이를 바탕으로 [8]에서는 Huffman 부호화 기술을 통해 부호화하였다. 본 연구에서는 이 과정에서 다양한 부호화 기법이 사용된 HEVC를 사용하여 더 효율적으로 압축한다. HEVC Encoder에서 Constant Rate Factor(CRF)를 통한 가변 Quantization Parameter(QP)를 제공한다.

4. 실험 및 결과 분석

본 논문에서는 VGG-16모델의 두 개의 Fully-Connected Layer를 압축하고 ImageNet 데이터셋으로 실험했다. 모든 실험과정은 재학습을 거치지 않고 진행되었으며 원본 가중치는 미리 학습된 가중치를 사용한다. ImageNet 검증 데이터셋은 1000가지 클래스에 대한 50000개의 이미지로 구성된다[10].

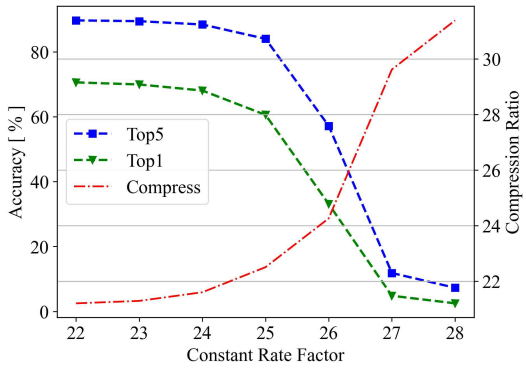

그림 2 CRF에 따른 압축률과 정확도

결과는 그림 2와 같이 CRF를 22로 설정하고 압축했을 때 정확도는 89.6%로 손실은 1% 미만이지만 21배의 압축률을 기록했다. CRF 25이상에서는 정확도가 급격하게 감소했다.

5. 결론

본 연구에서는 양자화하여 압축한 가중치 행렬에 대해 HEVC Encoder의 Intra mode를 사용하여 압축할 수 있다는 것을 보였다. 이 방법을 사용했을 때 정확도의 손실이 1% 이하에서 압축률 21배를 기록했다. 이 방법은 재학습을 거치지 않고 바로 사용할 수 있다. 또한, HEVC는 다양한 모바일 기기에서 하드웨어 단계에서 최적화되어 있다. 따라서 본 논문에서 제안하는 알고리즘은 모바일 기기에서도 사용할 수 있다.

사사문구

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터육성지원사업의 연구결과로 수행되었음 (IITP-2020-2017-0-01630)

참고문헌

[1] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[2] Sullivan, G. J., Ohm, J. R., Han, W. J., & Wiegand, T. (2012). Overview of the high efficiency video coding (HEVC) standard. IEEE Transactions on circuits and systems for video technology, 22(12), (pp. 1649-1668).

[3] Frankle, J., & Carbin, M. (2018). The lottery ticket hypothesis: Finding sparse, trainable neural networks. arXiv preprint arXiv:1803.03635.

[4] Gong, Y., Liu, L., Yang, M., & Bourdev, L. (2014). Compressing deep convolutional networks using vector quantization. arXiv preprint arXiv:1412.6115.

[5] LeCun, Y., Denker, J. S., & Solla, S. A. (1990). Optimal brain damage. In Advances in neural information processing systems (pp. 598-605).

[6] Han, S., Pool, J., Tran, J., & Dally, W. (2015). Learning both weights and connections for efficient neural network. In Advances in neural information processing systems (pp. 1135-1143).

[7] Frankle, J., & Carbin, M. (2018). The lottery ticket hypothesis: Finding sparse, trainable neural networks. arXiv preprint arXiv:1803.03635.

[8] Han, S., Mao, H., & Dally, W. J. (2015). Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149.

[9] Wu, J., Leng, C., Wang, Y., Hu, Q., & Cheng, J. (2016). Quantized convolutional neural networks for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4820-4828).

[10] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). Imagenet large scale visual recognition challenge. International journal of computer vision, 115(3), (pp. 211-252).