Glow 컴파일러 확장을 통한 혼합정밀도 양자화

(Mixed-Precision Quantization via Glow Compiler Extension)

이 제 민, 유 미선 , 권 용 인, 박 제 만, 김 태 호
한국전자통신연구원
(Jemin Lee, Misun Yu, Yongin Kwon, Jeman Park, Taeho Kim)
(Electronics and Telecommunications Research Institute)

Abstract: Deep neural networks (DNN) have been applied in various domains with remarkable performance improvements. Recently, the Glow compiler has been proposed to efficiently process deep neural networks. The Glow compiler provides quantization that reduces the model size by shrinking the precision of the data type. However, quantization leads to accuracy degradation of DNN models due to low precision data type. In this paper, we employ a mixed-precision method to compensate for the accuracy degradation. To do that, we extend the Glow compiler to support the layer-wise mixed-precision. With the Glow extension, we applied the mixed precision to ResNet18v1 and measured the variation of accuracy degradation depending on layers of DNN to investigate which layers contribute most to the accuracy loss. As a result, accuracy loss according to quantization for each layer shows high variation. From the experiment results, the desired accuracy could be achieved by sequentially preserving the full precision for the sensitive layers.

Keywords : quantization, mixed-precision, neural network compiler

Ⅰ. 서 론

딥뉴럴넷은 이미지처리, 자연어처리 등의 다양한 응용 분야에 적용되고 있다. 최근 딥뉴럴넷을 효율적으로 처리하기 위한 전용 컴파일러 연구들인 Glow[1], TVM[2], MLIR[3]들이 제안되었다. 이러한 컴파일러들은 다양한 최적화 기술들을 제공한다. 이러한 최적화 기술 중 하나인 양자화는 연산자 정밀도를 조절하여 낮은 비트의 데이터 유형 연산자로 대치하는 것이다. 하지만 양자화를 거치면 딥뉴럴넷 모델의 정확도 하락시키며, 이를 보상하려는 방법으로 양자화 인지 학습 기법들이 많이 연구되어 왔다[4].

학습 과정을 거쳐서 정확도를 복구하는 방법은 컴파일 단계에서 적용하기 어려우므로 본연구에서는 혼합정밀도를 이용해서 특정 레이어를 양자화하지 않고 원본 정밀도를 유지함으로써 정확도를 유지하는 방법을 사용한다. 혼합정밀도를 레이어 별로 적용하기 위해서 기존 오픈소스 딥러닝 컴파일러인 Glow에 직접 구현하였다. 확장된 Glow 컴파일러를 이용해서 ResNet18 딥뉴럴넷 모델에 혼합정밀도 기법을 적용하여 48개의 레이어 각각에 대해서 양자화가 미치는 영향을 분석했다.

실험 결과 ResNet18v1의 48개의 레이어 각각을 양자화했을 때의 Top-1 정확도 하락은 다양하게 나타났으며, 특정 레이어의 경우 양자화 수행 시 모든 레이어를 양자화했을 때보다 정확도 하락이 크게 발생시키는 현상을 발견했다. 이러한 정확도 하락을 양자화에 대한 민감도로 정의하고 민감도가 높은 순서대로 해당 레이어들을 기존 정밀도를 유지하도록 하여 효과적으로 타겟 정확도에 맞는 혼합정밀도 모델을 생성할 수 있음을 보였다.

*Corresponding Author (leejaymin@etri.re.kr)
이제민, 유미선, 권용인, 박제만, 김태호: 한국전자통신연구원

※ 이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2018-0-00769,인공지능 시스템을 위한 뉴로모픽 컴퓨팅 SW 플랫폼 기술 개발)

Ⅱ. 배경 지식

1. Glow 컴파일러의 양자화

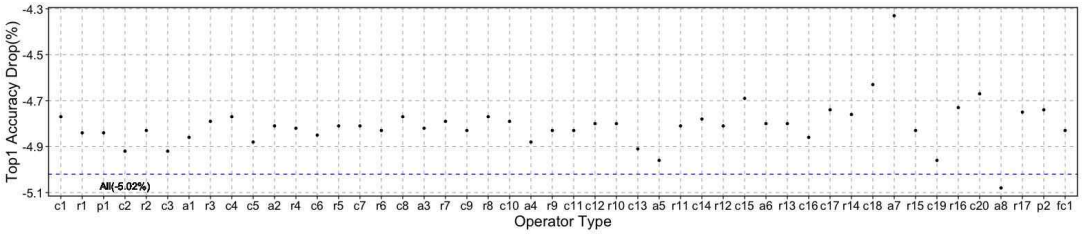

그림 1 레이어별 양자화에 따른 정확도 하락 (민감도 분석). 파란 선은 모든 레이어를 양자화했을 때의 정확도 하락을 나타냄.

Fig. 1. Accuracy drop depending on layer-wise quantization. Blue line denotes accuracy drop when all the layers are quantized.

Glow 컴파일러는 양자화를 위해서 다음 두 과정을 거친다. 1) 피쳐맵들의 데이터 분포를 알기 위해서 켈리브레이션을 맨 처음 수행하며, 2) 수집된 데이터 분포에 기반을 둬서 양자화를 위한 스케일 정보를 계산하여 실제 양자화를 수행한다.

정확도 하락이 발생하면 이를 보완하기 위해서 keep-original-precision-for-nodes 설정을 통해서 특정 연산자 타입 전체를 양자화하지 않는 기능을 제공한다. 해당 설정을 사용할 경우 연산자 전체가 양자화되지 않는다. 하지만 이러한 방법은 정밀하게 레이어를 선택해서 양자화 여부를 결정하지 못하므로 사용에 제약이 따른다. ResNet18v1 딥뉴럴넷의 경우 컨볼루션 연산자가 전체 48개의 레이어 중에서 20개를 차지한다. 컨볼루션 전체에 대해서 양자화를 수행하지 않는 것은 혼합정밀도를 사용하는 데 있어서 부담이 큰 방법으로 기본 Glow 컴파일러에서 제공하는 기능은 사용에 있어서 많은 제약을 가져온다.

2. 혼합정밀도 양자화 지원

Glow 컴파일러가 레이어별로 양자화 적용 여부를 선택하는 기능을 지원하도록 기능을 확장했다. 이를 위해서 기존 transformForPrecisionMode() 패스를 수정하여 미리 설정한 레이어 이름들에 대해서는 양자화를 거치지 않고 다음 최적화 패스들을 수행하도록 했다. 이러한 변경 부분이 포함된 Glow 컴파일러 코드는 오픈소스로 공개했다*.

본 논문에서 고려한 혼합정밀도의 전체 탐색 범위는 다음과 같이 계산한다. 고려한 혼합정밀도 B는 FP32와 INT8로 2로 결정된다. 전체 레이어 수를 L이라 하면 탐색 범위는 B^L이 된다. ResNet18v1 기준으로 2^{48}이 되므로 탐색할 수 없다. 따라서 이전 혼합정밀도 연구들과 같이 각각의 레이어들은 서로 독립적이라는 가정을 본 연구에서도 사용한다[5-7]. 이 경우 B^L의 탐색 범위는 단순히 BL이 되므로 해결 가능한 문제가 된다.

Ⅲ. 실험 결과

1. 실험 방법

딥뉴럴넷 모델 중 많이 활용되는 ResNet18v1을 사용했으며 모델은 Mxnet Gluon에서 Resnet18v1**을 내려받아서 ONNX 형식으로 변환해서 실험에 사용했다. 각각의 레이어의 민감도를 측정하기 위해서 확장된 GLow 컴파일러를 사용했으며, 각각의 레이어를 양자화한 모델들에 대해서 이미지넷12 검증 데이터셋 이미지 50,000개 전체를 이용해서 Top-1 정확도를 계산했다.

실제 추론은 3090 GPU 위에서 수행했다. Glow 컴파일러의 openCL 백엔드를 이용해서 GPU 코드를 생성했으며, 하나의 모델에 대해서 Top-1 정확도를 측정하는데 55분 이상이 소요되었다. 생성된 모델은 총 48개로 모든 모델에 대해서 정확도를 측정하는데 약 44시간의 시간이 걸렸다.

2. 실험 결과

사용된 Resnet18v1은 컨볼루션 20개, ReLU 17개, 풀링 2개, 엘리먼트 더하기 8개, FC 1개로 구성되었다. 총 48개의 레이어에 각각에 대해서 양자화를 한 번씩 수행하여 모델의 정확도를 측정했다. 실험 결과는 그림 1과 같으며, 48개의 레이어 각각에 대해서 서로 다른 정확도 하락을 나타내는 것을 알 수 있다. 이러한 정확도 하락을 양자화에 대한

* https://github.com/etri/nest-compiler

** https://mxnet.apache.org/

민감도로 나타낸다. 모든 레이어에 대해서 양자화를 수행했을 때의 정확도 하락은 -5.02%이며, a8 레이어의 경우 양자화 수행 시 정확도 하락이 -5.08%로 전체 양자화 수행 시보다 더 크다. 따라서 측정된 민감도에 따라 영향이 큰 것부터 FP32 기존 정밀도를 유지하도록 하여 효과적으로 혼합정밀도 모델을 생성할 수 있음을 실험 결과 알 수 있다.

Ⅳ. 결 론

본 연구에는 양자화로 인한 정확도 하락을 보완하는 기법의 하나인 혼합정밀도 방법을 Glow 컴파일러에서 세밀하게 지원할 수 있도록 확장했다. 확장된 기능을 이용해서 레이어별로 혼합정밀도 양자화 모델을 생성했으며, 생성된 모델의 정확도를 모두 측정하여 민감도를 분석했다. 분석 결과 민감도는 모두 확연히 다름을 보였고 이러한 민감도가 큰 레이어부터 양자화를 수행하지 않음으로써 원하는 정확도의 모델을 효과적으로 얻을 수 있음을 보였다. 향후 연구로는 실제 Top-1 정확도를 측정하지 않고서도 양자화 민감도를 간접적으로 계산하는 방법을 고안하여 모델 탐색시간을 줄여볼 계획이다.

References

[1] Rotem, Nadav, et al. "Glow: Graph lowering compiler techniques for neural networks." arXiv preprint arXiv:1805.00907, 2018.

[2] Chen, Tianqi, et al. "TVM: An automated end-to-end optimizing compiler for deep learning." USENIX Symposium on Operating Systems Design and Implementation, 2018.

[3] C. Lattner, et al., "MLIR: Scaling Compiler Infrastructure for Domain Specific Computation," in 2021 IEEE/ACM International Symposium on Code Generation and Optimization (CGO), Seoul, Korea (South), 2021 pp. 2-14.

[4] Zhou, S., Wu, Y., Ni, Z., Zhou, X., Wen, H., and Zou, Y. DoReFa-Net: Training low bitwidth convolutional neural networks with low bitwidth gradients. arXiv preprint arXiv:1606.06160, 2016.

[5] Dong, Zhen, et al. "Hawq: Hessian aware quantization of neural networks with mixed-precision." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019.

[6] Dong, Zhen, et al. HAWQ-V2: Hessian Aware trace-Weighted Quantization of Neural Networks, Advances in Neural Information Processing Systems 33 (NeurIPS 2020).

[7] Yao, Zhewei, et al. "HAWQ-V3: Dyadic Neural Network Quantization." International Conference on Machine Learning 2021.