딥러닝 활성화 데이터 압축을 위한 연속 길이 부호화 방법

*김성제 *이승호 *홍민수 *정진우
한국전자기술연구원
*sungjei.kim@keti.re.kr

Run-Length Coding for deep-learning activation data compression

*Kim, Sungjei *Lee, Seungho *Hong, Minsoo *Jeong, Jinwoo
Korea Electronics Technology Institute

요약

최근 다양한 응용 분야에서 딥러닝을 적용한 사례가 나오고 있으며, 딥러닝 네트워크 경량화 또는 압축 기법을 적용해 정확도는 최대한 유지하면서 에너지 효율을 개선하려는 연구도 활발하게 이루어지고 있다. 이에 본 논문에서는 딥러닝 추론 과정에서 중간 데이터로 도출되는 활성화 데이터의 압축을 위해 연속 길이 부호화 방법을 적용해보고 압축률과 개선점에 대해 분석 한다.

1. 서론

딥러닝 기술의 확산과 더불어 다양한 응용 분야에서 딥러닝 추론 기법을 적용한 사례가 나오고 있으며, 최근에는 모바일, 임베디드 플랫폼 등과 같은 경량 단말에도 적용하려는 연구도 활발하게 이루어지고 있다 [1-3]. 특히 전력 효율이 중요한 모바일 단말에 탑재되는 초저전력 HW 가속기의 경우, 외부 메모리와 가속기 간의 전송 데이터의 양을 줄여 전력 효율을 개선하는 연구를 진행하였다 [4-6].

딥러닝 네트워크 가중치나 중간 데이터인 활성화 데이터 (또는 특징맵) 등은 가속기 내부 메모리에서 저장하거나 한 번에 처리할 수 없어 최대한 외부 메모리로 전송되는 데이터양을 줄여야 하는데, 이 때 데이터양을 줄이기 위해 다양한 변환 기법, 양자화 기법, 엔트로피 코딩 기법 등이 사용되었다 [6, 7].

본 논문에서는 영상 분류용 딥러닝 네트워크인 EfficientNet-B0를 기준 모델 [8]로 하여, 다양한 연속 길이 부호화 기법을 활성화 데이터의 압축에 적용해보고, 압축 성능 및 개선점에 대해서 분석 한다.

2. 활성화 데이터의 압축

활성화 데이터는 딥러닝 추론 과정에서 매 레이어마다 얻을 수 있는 중간 데이터로서, 활성화 함수의 출력이라는 의미로 활성화 데이터 (Activation data)라고 하거나 컨볼루션 필터링의 결과물이라는 의미로 특징맵 (Feature Map)이라고도 불린다. 활성화 데이터는 현재 레이어의 출력이면서 바로 다음 또는 미래 레이어의 입력 (Skip Connection)으로 사용되기 때문에 가속기 내부 또는 외부 메모리에 저장하는 것이 필수적이다.

Y.-H. Chen *et al.* [4]은 AlexNet 또는 VGG-16 등의 딥러닝 네트워크가 활성화 함수가 ReLU인 것을 고려 (0보다 작은 입력 값은 0으로 출력)하여, 연속 길이 부호화 방법을 이용하여 활성화 데이터를 압축하는 방법을 HW로 구현하여 외부 메모리에 저장하는 데이터양을 개선하였다. 연속 길이 부호화 방법은 인코딩과 디코딩 방법이 다른 엔트로피 코딩 방법에 비해 구현하기에 용이하고, 연속된 심볼이 많이 출현하는 환경에서는 압축률도 우수하여 0 값이 많은 활성화 데이터를 압축하기에 적합하다.

3. 연속 길이 부호화 방법

연속 길이 부호화 방법은 연속된 심볼의 숫자 (Run-length)를 세어, 심볼의 개수를 줄이고 대신 심볼의 Run-length를 압축하는 방식으로 그림 1과 같다. 그림 1의 상단에 파란색 사각형 내의 숫자들은 심볼 정보이고, 제일 하단에 있는 주황색 사각형들은 비트스트림 출력이다. 그림 1의 예에서 심볼 2는 5번 나오는데, 이 경우 연속 길이 부호화 방법은 2를 8비트로 표현하고 5를 3비트로 표현해, 비트스트림을 생성한다.

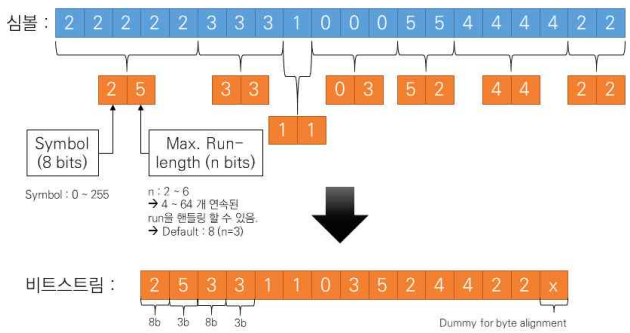


그림 1. 임의 심볼 기준 연속 길이 부호화 방법 (SRLC)

그림 2는 연속 길이 부호화 방법의 변형된 형태로, 연속된 임의 심볼을 모두 세는 것이 아니라, 0 심볼의 연속된 숫자만 세는 방법으로 0 심볼에 대한 값을 보내지 않아도 되는 장점을 갖는다.

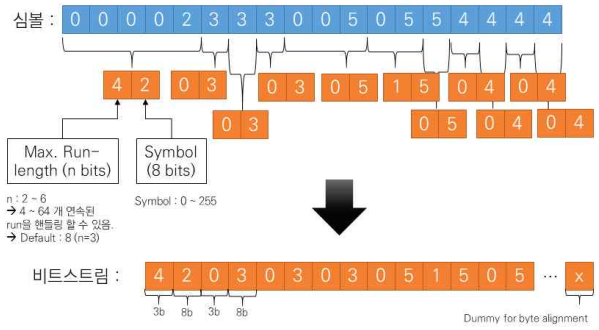


그림 2. 0 심볼 기준 연속 길이 부호화 방법 (ZRLC)

연속 길이 부호화 방법은 연속된 심볼 숫자가 많을수록 압축률이 커지지만, 연속된 심볼의 숫자가 적다면 오히려 압축률에서 전혀 이득을 볼 수 없어 원본 데이터보다 출력 비트스트림의 크기가 커지는 한계점을 가지고 있다.

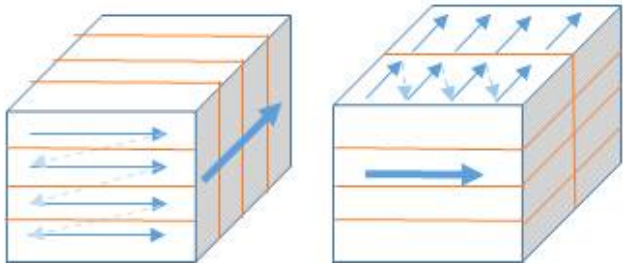
4. 실험 및 결과

본 논문에서는 EfficientNet-B0 [8]을 기반으로 연속 길이 부호화 방법을 테스트해보았다. EfficientNet은 2019년에 구글에서 제안한 영상 분류 및 인식을 위한 딥러닝 네트워크로 종래의 딥러닝 기법에 빕해 파라미터 수가 획기적으로 작으면서도 다양한 네트워크 구조 (Squeeze- and-Excitation, Elemental-wise multiplier/adder 등) 뿐 아니라, 활성화 함수도 Linear와 Swish, Sigmoid 등이 사용되었다. 본 논문에서는 Swish 대신 ReLU6로 변경하여 재학습을 수행하였고, 이를 기반으로 연속 길이 부호화 방법의 성능을 측정하였다. 테스트에 사용된 데이터셋은 Cifar-10과 ImageNet을 사용하였다.

표 1. 연속 길이 부호화 방법에 따른 압축률 차이

Cifar-10 Dataset				
방법	SRLC	SRLC +BYPASS	SRLC+BY PS+SCAN	ZRLC+BY PS+SCAN
압축률	20.17%	23.87%	8.68%	36.22%
ImagNet Dataset				
방법	SRLC	SRLC +BYPASS	SRLC+BY PS+SCAN	ZRLC+BY PS+SCAN
압축률	9.25%	11.25%	11.25%	19.06%

연속 길이 부호화 방법 중 임의 심볼 기준 부호화 방법은 표 1에서 SRLC로 Cifar-10 Dataset에 대해서 약 20.17% 압축률을 개선하였다. 여기에 Linear나 Sigmoid와 같은 활성화 함수를 갖는 경우는, 연속 길이 부호화 방법을 사용하지 않도록 적용하여 (BYPASS) 추가적으로 약 3% 압축률을 개선하였다. 여기에 활성화 데이터를 부호화하는 순서를 그림 3과 같이 바꾸어서 측정해보았으나 큰 성능 향상은 없었다. 하지만, 0 심볼 기준으로 부호화하는 방법 (ZRLC)과 신규 스캔 방식을 사용하였을 때, 그 성능 향상 효과가 큰 것을 확인할 수 있었다.



(a) 기존 스캔 방식 (b) 타일 분할 스캔 방식

그림 3. 활성화 데이터 스캔 방식

4. 결론

본 논문에서는 활성화 데이터의 압축을 위해 다양한 연속 길이 부호화 방법을 EfficientNet-B0에 대해서 적용해보았다. ZRLC와 채널을 먼저 스캔하는 타일 분할 스캔 방식을 적용할 때, Cifar-10 데이터셋의 활성화 데이터에 대해서 약 36.22% 압축 성능을 보이는 것을 실험적으로 확인하였다.

Acknowledgements

이 논문은 2020년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2019-0-01351, 활성화/커널데이터의 압축/복원을 통한 초저전력 모바일 딥러닝 반도체 기술 개발)

참 고 문 헌

[1] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, “Binarized Neural Networks”, in Advances in Neural Information Processing Systems (NIPS), pp. 4107-4115, 2016.
[2] M. Rastegari, V. Ordonez, J. Redmon, and Ali Farhadi, “XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks”, in European Conference on Computer Vision (ECCV), pp.525-542, 2016.
[3] J. Lin, W. Chen, Y. Lin, J. Cohn, C. Gan, and S. Han, “MCUNet: Tiny Deep Learning on IoT Devices”, in Advances in Neural Information Processing Systems (NIPS) 2020.
[4] Y.-H. Chen, J. E. and V. Sze, “Eyeriss: A Spatial Architecture for Energy-Efficient Dataflow for Convolutional Neural Networks”, International Symposium on Computer Architecture (ISCA) 2016.
[5] S. Han, X. Liu, H. Mao, J. Pu, A. Pedram, M. Horowitz, W. J. Dally, “EIE: Efficient Inference Engine for Compressed Deep Neural Network”, International Symposium on Computer Architecture (ISCA), 2016.
[6] S. Han, H. Mao, W. J. Dally, “Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding”, in International Conference on Learning Representations (ICLR), 2016.
[7] Y. Wang, C. Xu, C. Xu and D. Tao, "Packing Convolutional Neural Networks in the Frequency Domain," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, no. 10, pp. 2495-2510, 1 Oct. 2019.
[8] M. Tan, and Q. Le “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks”, International Conference On Machine Learning (ICML) 2019.