엣지 디바이스를 위한 자동 모델 경량화 및 성능 비교 연구

이문영, 김현우, 전승협, 박성천
한국전자통신연구원
{munyounglee, kim.hw, shjeon00, scpark}@etri.re.kr

A Study on Automatic Model Compression and Performance Comparison for Edge Devices

Munyoung Lee, Hyunwoo Kim, Seung Hyub Jeon, Seong-Cheon Park
Electronics and Telecommunications Research Institute

요 약

연산 능력과 메모리 용량에 제한이 있는 다양한 엣지 디바이스에서 인공지능을 활용하는 사례가 늘어남에 따라 인공지능 모델을 효율적으로 만드는 경량화 기술에 대한 관심이 증가하고 있다. 각 엣지 디바이스에 적합한 효율적인 경량 모델을 찾기 위해서는 사전 학습모델에 대해 경량화 알고리즘의 파라미터를 변경하는 반복적인 실험을 통해 다양한 조건 변화에 따른 성능을 비교하면서 최적의 모델을 탐색하는 과정이 필요하다. 이에 본 연구에서는 오픈 소스 AutoML 툴킷인 Neural Network Intelligence의 경량화 알고리즘을 기반으로 동작하는 자동 모델 경량화 및 성능 비교 시스템을 개발하였다. 실험을 통해 자동 모델 경량화 및 성능 비교 시스템을 활용하여 경량화 알고리즘의 파라미터 변화에 따른 다양한 사전 학습모델의 성능을 시각화하여 비교하는 활용 사례를 소개하고, 응용 시나리오에 적합한 경량 모델을 탐색하는 과정에 사용할 수 있음을 확인하였다.

Ⅰ. 서론

인공지능 기술의 빠른 발전과 함께 다양한 종류의 엣지 디바이스(edge device)에서 인공지능을 활용하는 사례가 증가하고 있다. 일반적으로 엣지 디바이스는 연산 능력과 메모리 용량에 제한이 있기 때문에 인공지능 모델의 크기를 줄이면서도 성능 하락을 최소화하는 경량화 기술에 대한 관심이 높아지고 있다.

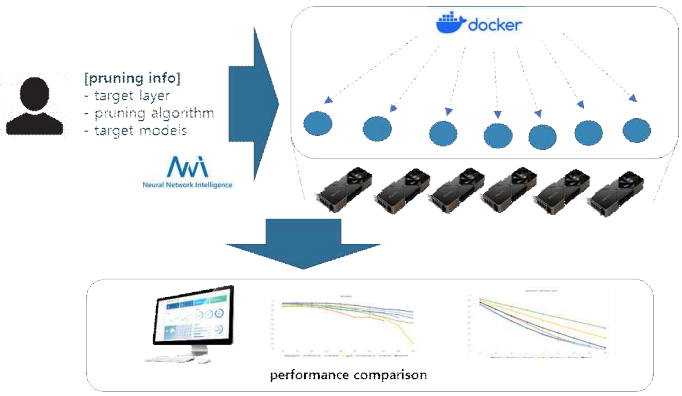
각 엣지 디바이스에 적합한 효율적인 인공지능 모델을 찾기 위해서는 타겟 응용 시나리오에서 이용 가능한 인공지능 모델을 확보한 후, 다양한 경량화 기법을 적용해야 한다. 즉, 경량화 알고리즘의 파라미터를 변경하면서 반복적인 실험을 통해 다양한 조건에 따른 성능을 비교하면서 최적의 경량 모델을 탐색하는 과정이 필요하다.

본 연구에서는 Microsoft에서 개발한 오픈 소스 AutoML 툴킷인 Neural Network Intelligence(NNI)[1]의 경량화 알고리즘을 기반으로 동작하는 자동 모델 경량화 및 성능 비교 시스템을 제안한다. 본 연구를 통해 엣지 디바이스를 위해 경량화 모델을 자동적으로 탐색하여 성능을 비교하는 시스템의 활용 가능성을 확인하였다.

Ⅱ. 본론

1. 가지치기(Pruning) 기법

경량화 기술은 효율적인 학습 및 추론을 위해 인공지능 모델의 크기를 줄이고, 계산 및 메모리 요구사항을 최적화하는 기술을 의미하고, 가지치기(pruning), 양자화(quantization), 지식 증류(knowledge distillation) 등의 기법[2]이 있다. 본 연구에서 활용한 가지치기(Pruning) 기법은 인공지능 모델의 효율성을 높이기 위해 신경망을 구성하는 노드 간 연결을 제거하는 기법을 의미한다. 기본적인 개념은 모델의 각 가중치 값이 결과에 미치는 영향이 다르다면 중요도가 낮은 가중치를 제거하여 모델의 파라미터를 줄여서 효율성을 높이는 것이다. 이를 위해 일부 가중치의 값을 0으로 수정하는 방식으로 모델을 구성하는 노드 간 연결을 제거하는 효과를 낸다. 제거할 가중치는 가지치기 알고리즘에 의해 선정된다.



〔그림1〕 Docker 기반 자동 모델 경량화 비교 시스템

2. 시스템 구조

그림1은 자동 모델 경량화 및 성능 비교 시스템의 개념도를 나타낸다. 본 시스템은 Microsoft에서 공개한 오픈 소스 AutoML 툴킷인 Network Intelligence(NNI)를기반으로 동작한다. NNI는 심층 신경망(Deep Neural Networks)를 압축하기 위한 모델 압축 프레임워크를 제공하고 있어서 L1Norm, L2Norm, FPGM Pruner 등의 다양한 가지치기 알고리즘을 사용할 수 있다. 자동 모델 경량화 및 성능 비교 시스템에서는 NNI 기반 모델 자동 경량화 모듈을 가상 도커 위에서 분산 병렬 실행할 수 있게 구성하였고, 목표 계층(target layer), 가지치기 알고리즘(pruning algorithm), 사전 학습모델(pre-trained models)에 대한 정보를 입력하면 주어진 조건에서 가지치기 비율(pruning rate)을 10%에서 80%까지 변화시키면서 다양한 조건에 따른 경량화 모델의 성능을 비교하여 시각화된 결과를 제공한다. 본 연구에서는 NNI에서 제공하는 경량화 알고리즘을 적용하는 예시를 보여주고 있지만, 본 시스템을 모델 압축 알고리즘을 제공하는 타 프레임워크와 연동하거나 자체적으로 개발한 경량화 알고리즘을 적용하여 성능을 비교하는 방식의 활용도 가능하다.

3. 성능 비교

자동 모델 경량화 및 성능 비교 시스템의 활용 가능성을 확인하기 위해 PyTorch에서 제공하는 Imagenet 데이터로 학습된 사전 학습모델[3] 중, 표1의 7종을 대상으로 conv2D layer에 L1NormPruner 알고리즘[4]을 적용한 경량 모델들을 비교하였다. 사전 학습모델에 가치기기 기법을 적용 후에는 CIFAR-10 데이터셋[5]으로 미세조정(fine-tuning)을 진행한 후, 성능을 비교하였다. 성능 비교 지표로는 추론 정확도(accuracy)와 파라미터 감소율(parameter reduction ratio)을 사용하였다.

〔표1〕 사전 학습모델 7종 및 가치치기 비율

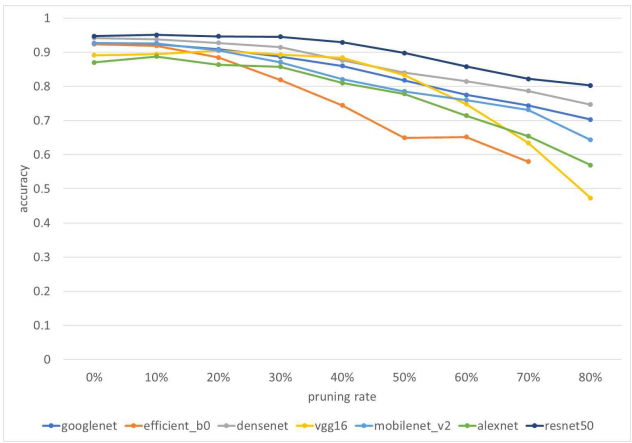
No.	pre-trained model	pruning rate
1	googlenet	10%~80%
2	efficient_b0	
3	densenet121	
4	vgg16	
5	mobilenet_v2	
6	alexnet	
7	resnet50	

그림2는 사전 학습모델 7종에 대해 가지치기 기법을 적용한 경량 모델에 대한 추론 정확도(accuracy) 성능을 보여준다. 가지치기 비율이 높아질수록 신경망을 구성하는 노드 간 연결이 사라지기 때문에 경량 모델의 추론 정확도가 떨어지는 것을 확인할 수 있다. 경량 모델 7종에 대해 10%에서 80%의 비율로 가치치기를 수행 시, resnet50 모델의 성능 하락이 가장 적은 것을 확인할 수 있다. 실험 결과에 따르면 사전 학습모델의 추론 정확도가 약 95%인 resnet50 모델의 가중치의 80%를 가지치기 했을 경우에도 약 80%의 추론 정확도를 유지하는 것을 확인할 수 있다. 반면 vgg16 모델은 가지치기 비율이 40%를 넘어가면서 추론 정확도가 급격하게 떨어지는 특징을 보인다.

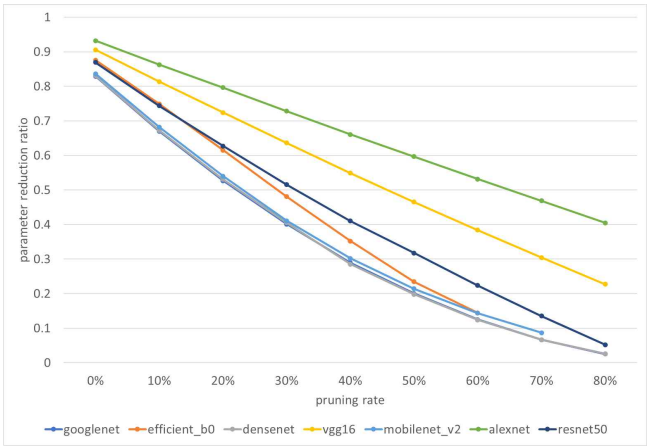
그림3은 가지치기 기법을 적용하여 인공지능 모델의 파라미터가 얼마나 감소하였는지를 보여주는 파라미터 감소율(parameter reduction ratio)에 대한 성능 비교 결과이다. 가지치기 비율이 증가함에 따라 alexnet, vgg16, resnet50 모델은 선형적인 파라미터 감소를 보이지만, 타 모델들은 가지치기 비율이 40%를 넘어가면서 파라미터 감소율이 줄어드는 모습을 보인다. 이와 같이 자동 모델 경량화 및 성능 비교 시스템을 활용하면 경량화 알고리즘의 파라미터 변화에 따른 다양한 사전 학습모델의 성능을 시각화하여 비교한 후, 응용 시나리오에 적합한 경량 모델을 선택할 수 있다.

Ⅲ. 결론

본 연구에서는 각 엣지 디바이스에 적합한 경량 모델 탐색에 소요되는 반복적인 실험 및 성능 비교에 소요되는 시간을 단축하기 위해 오픈 소스 AutoML 툴킷인 NNI의 경량화 알고리즘을 기반으로 동작하는 자동 모델 경량화 및 성능 비교 시스템을 제안하였고, PyTorch에서 제공하는 사전 학습모델 7종을 대상으로 가치치기 알고리즘인 L1NormPruner을 conv2D layer에 적용하여 경량화 알고리즘의 파라미터 변화에 따른 다양한 사전 학습모델의 성능을 시각화하여 비교하였다. 실험 결과, 가치치기 비율에 따른 사전 학습모델들의 추론 정확도 변화와 파라미터 감소율 변화를 비교할 수 있었고, 응용 시나리오에 적합한 최적의 경량 모델을 탐색하는 과정에 자동 모델 경량화 및 성능 비교 시스템을 활용하여 다양한 조건에 따른 경량 모델의 성능을 비교할 수 있음을 확인하였다. 향후에는 국산 신경망 처리장치(Neural Processing Unit, NPU)가 탑재된 엣지 디바이스를 지원하기 위해 시스템에 딥러닝 컴파일러 기술을 연동하고, 다양한 경량화 방법을 적용한 경량화 모델을 효율적으로 탐색하는 연구를 진행할 계획이다.



〔그림2〕 모델별 accuracy 성능 비교



〔그림3〕 모델별 parameter reduction ratio 성능 비교

ACKNOWLEDGMENT

이 논문은 2024년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. 2022-0-00454, 스마트 엣지 디바이스 SW 개발 플랫폼 개발)

참 고 문 헌

[1] Neural Network Intelligence, https://nni.readthedocs.io

[2] M. GUPTA and P. Agrawal, "Compression of Deep Learning Models for Text: A Survey, " ACM Transactions on Knowledge Discovery from Data (TKDD), Volume 16, Issue 4, Jan 2022.

[3] PyTorch models, https://pytorch.org/vision/main/models.html

[4] H. Li et al, "Pruning Filters for Efficient ConvNets," International Conference on Learning Representations (ICLR), 2017.

[5] A. Krizhevsky and G. Hinton, G, "Learning multiple layers of features from tiny images, " Technical Report, 2009.