Various AI model compression techniques

for allowing inference speed up and energy saving without significant accuracy losses

2022. 10. 07.

Data Mining & Quality Analytics Lab.

방병우

KOREA UNIVERSITY Data Mining Quality Analytics

발표자 소개

방병우 (Bang Byungwoo)

- 고려대학교 산업경영공학과
- Data Mining & Quality Analytics Lab(김성범 교수님 연구실)
- 박사 과정(2022.03 ~)

관심 연구 분야

- Anomaly Detection
- Multivariate Time Series Data
- Natural Language Processing

E-mail

- E-mail: byungwoo.bang@gmail.com

Contents

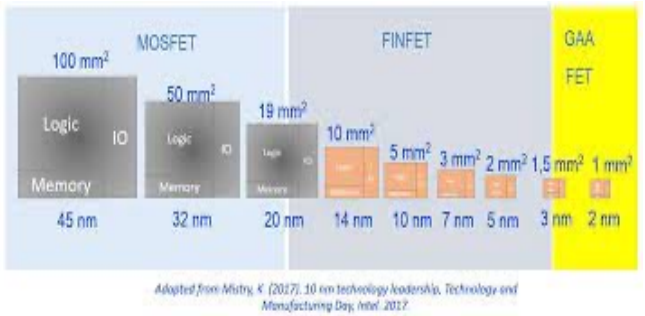
- 배경
- 모델 경량화 방법
  - Pruning
  - Knowledge Distillation
  - Quantization
- Conclusion

Introduction

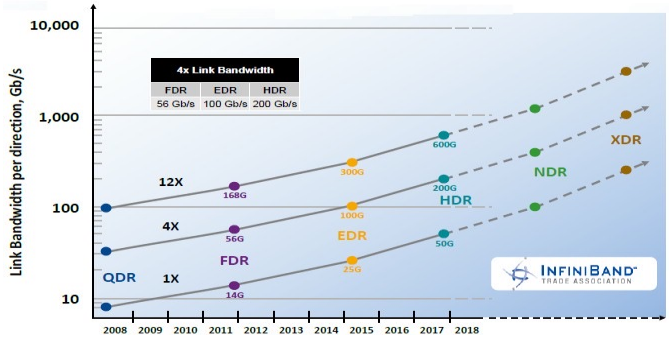
개선된 구조와 설계를 통해 발전중인 AI 가속기 성능

❖ 미세공정 개선, 개선된 알고리즘, HBM 메모리 등의 기능의 발전

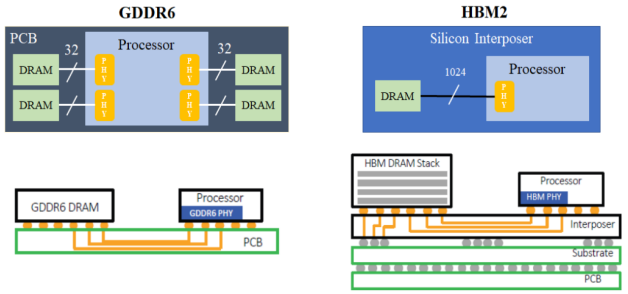
- 데이터 센터, 서버향 하드웨어의 경우 지난 5년 간 20배 넘는 성능 향상



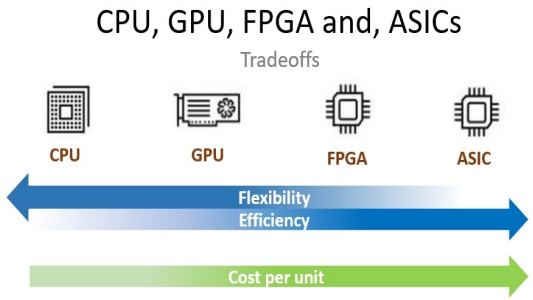
미세공정 개선



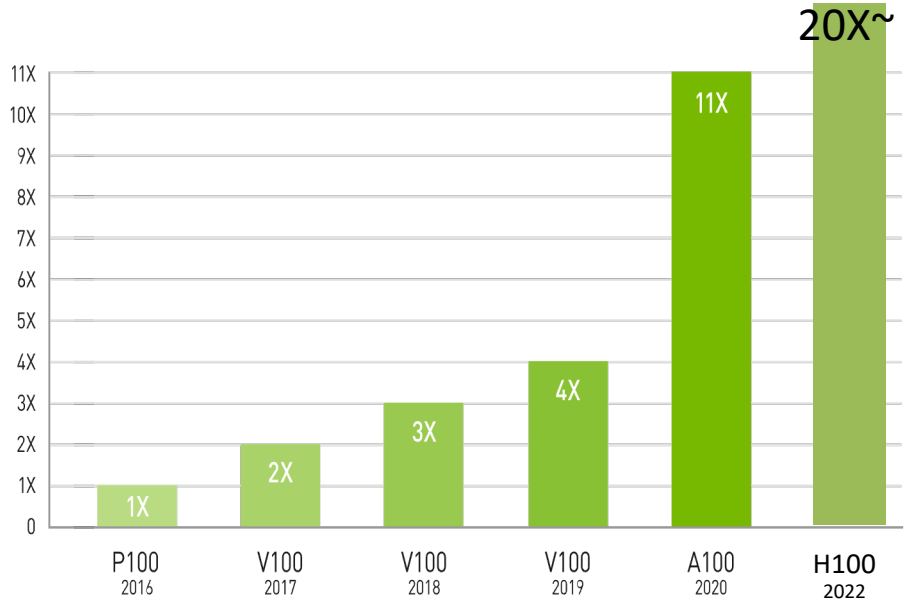
Network 성능개선



HBM Memory의 출현



전용 가속기의 등장



NVIDIA 제품 기준

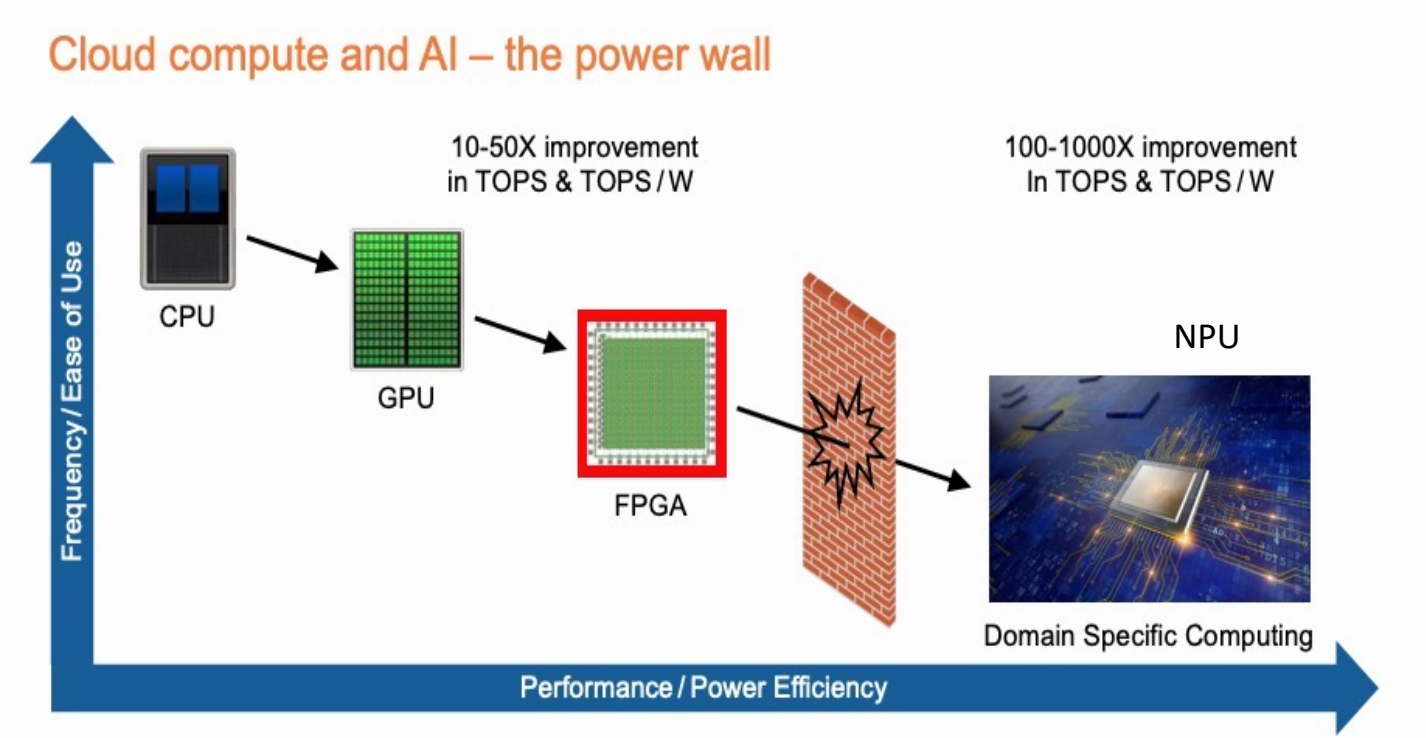
https://www.nvidia.com/en-us/data-center/a100/

서버향 가속기와 다른 모바일 디바이스 환경

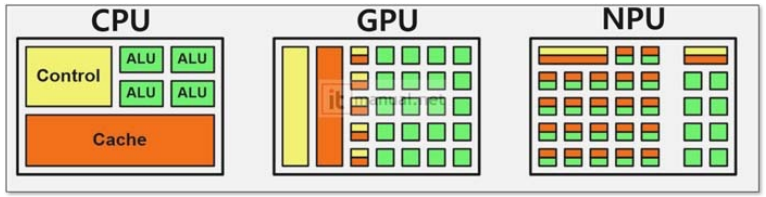
휴대폰의 경우 1분 60도, 10분 48도의 안전 규격(EN563)

❖ **서버향 AI 가속기와는 다르게 온도, 전력등에 대한 제약으로 큰 성능 개선 기대 불가**

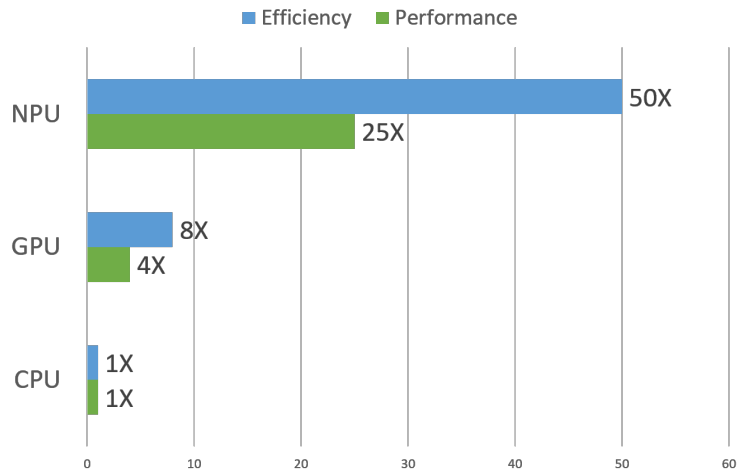
- 한계를 개선하기 위해 특정 텐서 연산에 특화된 전용프로세서(NPU)로 개선 수행



GPU의 경우 성능이 좋지만 발열과 소모되는 전력 커서, Training이나 여러 모델을 동시에 Serving 하는데 효과적



CPU – GPU – NPU 간 구조 차이



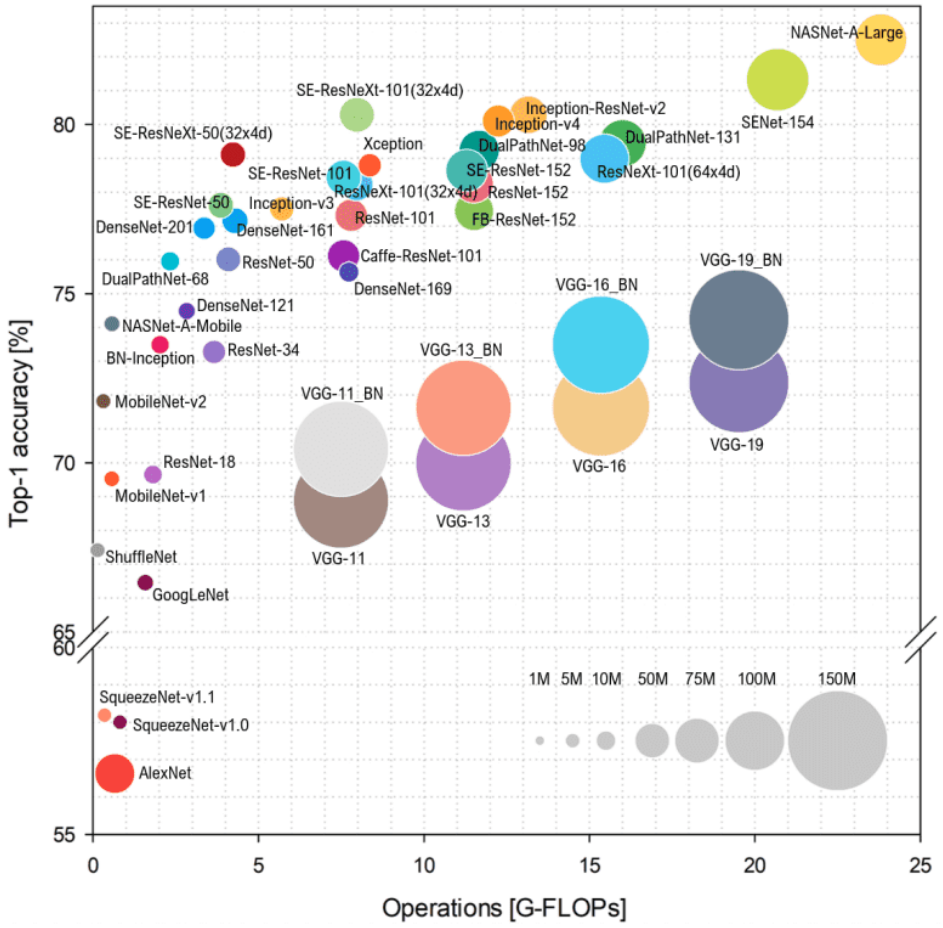
CPU, GPU, NPU 간 성능 비교

https://semiwiki.com/semiconductor-manufacturers/globalfoundries/281970-specialized-accelerators-needed-for-cloud-base[illegible]-training/

https://thetechrevolutionist.com/2017/09/huaweis-neural-processing-unit-np[illegible]and.html

NPU를 탑재했지만, AI 모델이 요구하는 모델 크기가 지속 증가 중

❖ 저전력, 고성능을 위한 제품인 NPU 환경에서, 지속적으로 커져가는 AI 모델을 Serving하기 힘들어지고 있음.



4년간 7.9% 향상 하는동안 파라미터수 36배 증가

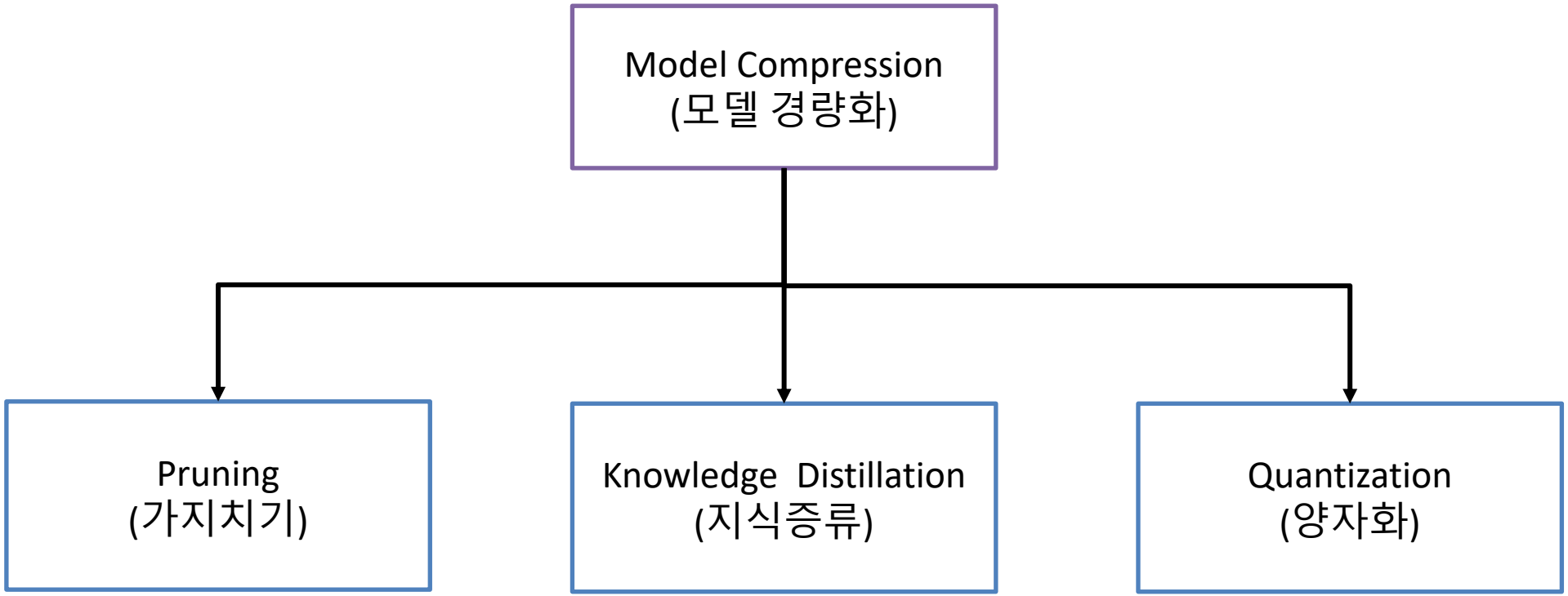
- **파라미터 기준**
  ILSVRC 우승 모델 top-1 정확도
  Google Net : 74.8% with 4M (2014년)
  Squeeze-and Excitation Network : 82.7% with 145.8 (2018년)

- **연산량 기준**
  AlexNet(2012년) 대비 NASNet(2018년) 의 경우 25배의 연산량 증가

제한된 NPU에, 커져만 가는 모델을 어떻게 장착할 수 있을까?
➔ 모델 경량화(Model Compression)

모바일 디바이스에 필수적인 모델 경량화 (Model Compression)

❖ 모델 경량화 기술은 정확도의 손실을 기존 모델 대비 최소화 하면서 모델 크기와 연산량을 줄이는 기술



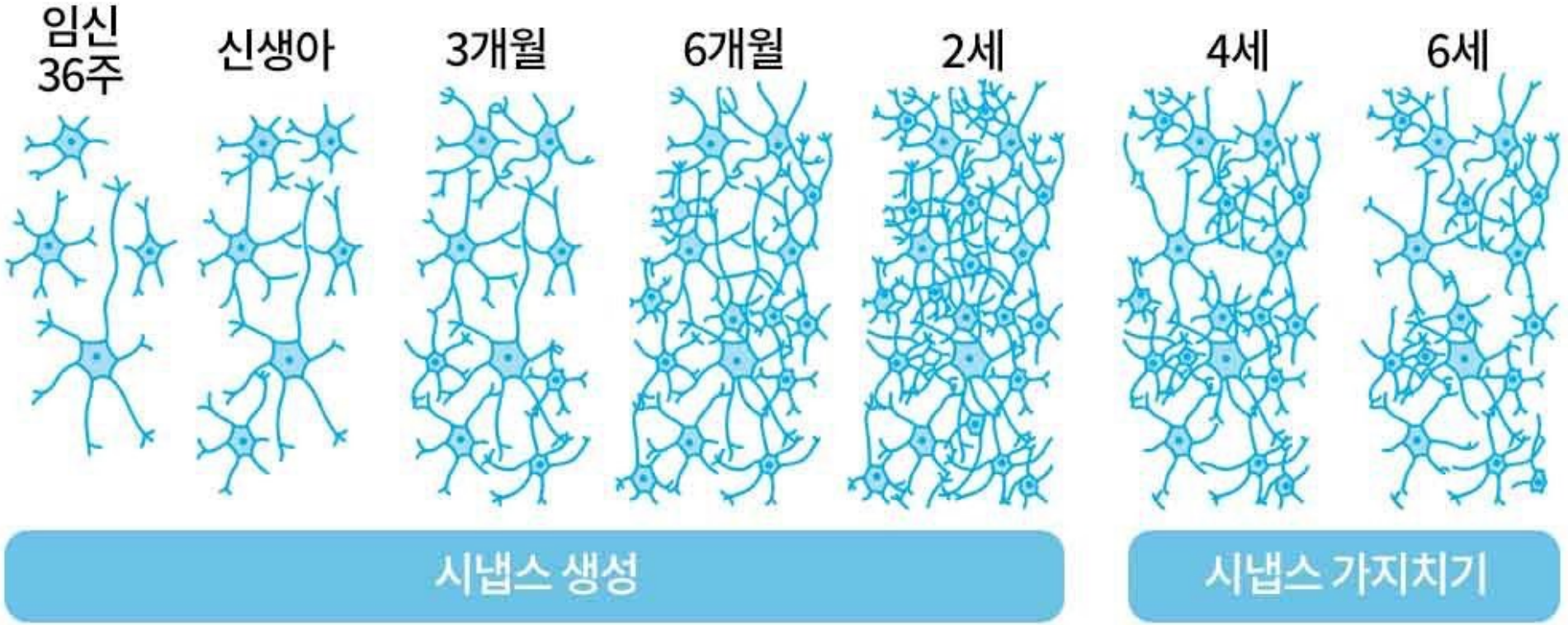
Various Model Compression Techniques

- Pruning

Pruning (가지치기)

❖ 뇌 발달에 중요한 뉴런

- 뉴런을 연결해주는 시냅스의 수는 만 3세까지 새로운 연결을 만들며, 반복과 연습을 통해 연결 강화
- 4세 이후 사용되지 않는 신경 연결은 가지치기 함.

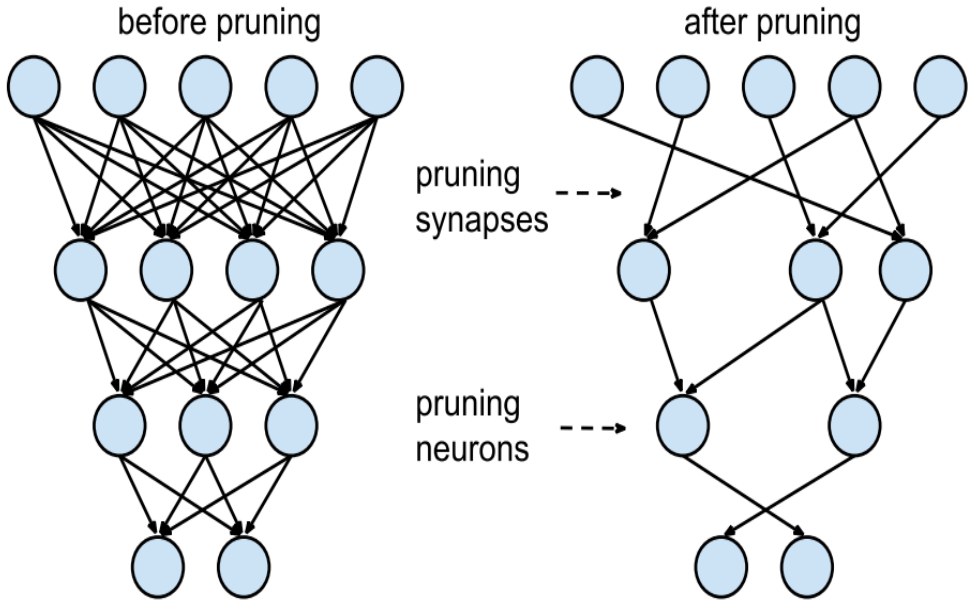


Pruning (가지치기)

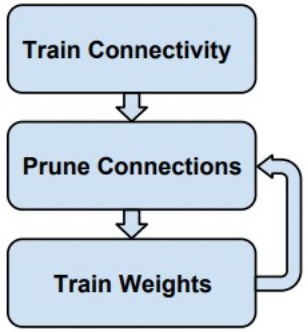
❖ Model에서, Weight들 중 중요도가 낮은 weight의 연결을 제거하여 모델의 파라미터 수를 줄이는 방법

- "Learning both Weights and Connections for Efficient Neural Networks" 2015, Song Han 교수

https://arxiv.org/pdf/1506.02626.pdf



Pruning



Pruning 순서

신경망 학습 과정 중, 출력 결과에 영향을 거의 주지 않는 간선 존재

→ 특정 보다 낮은 값을 갖는 뉴런들에 대해선 연결을 제거하고, 재학습을 반복

→ 연산량과 파라미터 수를 줄이게 되어 성능이 개선된다.

AlexNet Model

Accuracy Loss : 0.1% (42.78% → 42.77%)

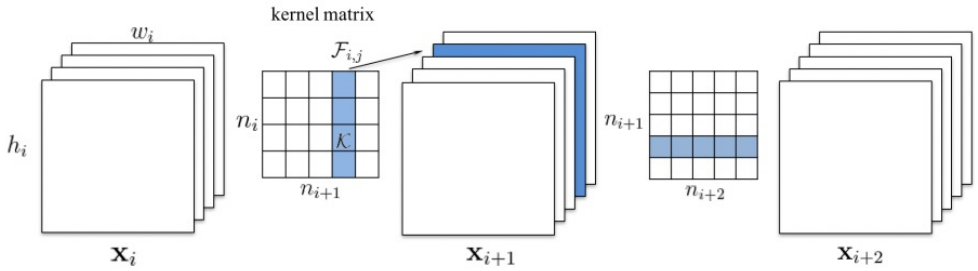
파라미터개수: 9배 개선(61M → 6.7M)

Pruning (가지치기)

❖ Filter 및 Channel 들까지 확장하는 가지치기 방법들

- “Pruning Filters for Efficient ConvNets”, 2016, Hao Li

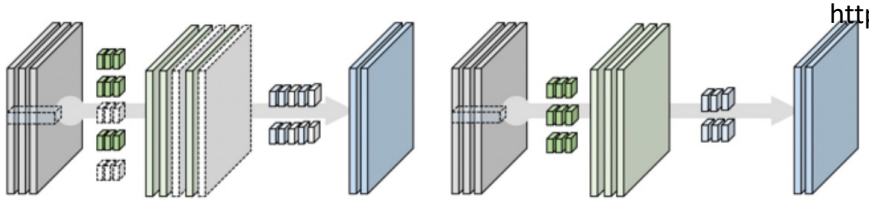
https://arxiv.org/abs/1608.08710



Filter와 Feature Map에 적용

- “Channel Pruning for Accelerating Very Deep Neural Networks”, Yihui He

https://arxiv.org/abs/1707.06168



Channel 에 적용

ResNet-50 모델의 경우 0.6% 개선하면서 모델 크기를 15배 경량화(102.5MB → 6.7MB)

- 단점: 지속적인 가지치기 및 클러스터링 연산을 수행을 위해 소요되는 시간이 매우 많다.

Pytorch등의 framework에서 사용 가능

https://pytorch.org/tutorials/intermediate/pruning_tutorial.html

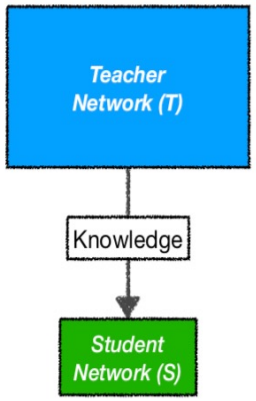
Various Model Compression Techniques

- Knowledge Distillation

Knowledge Distillation (지식 증류)

❖ Training과 Inference는 서로 다른 task이므로, 모델 또한 서로 다르게 사용되어져야 한다.

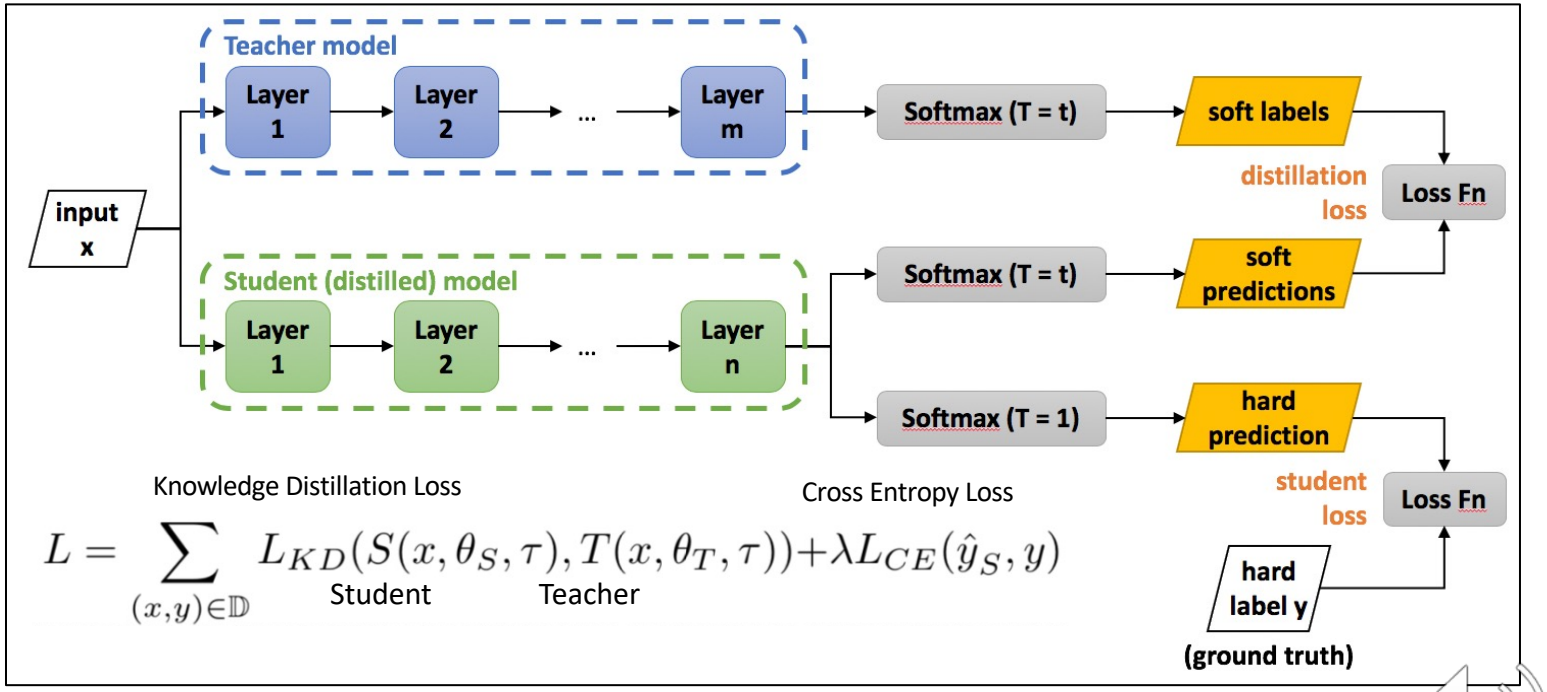
- "Distilling the Knowledge in a Neural Network", 2014, Geoffrey Hinton
- Student 모델 학습시 모델의 Loss와 Teacher 모델의 Loss를 동시에 반영하여 Student 모델 학습에 활용



1. **Teacher Network (T)**
   - **cumbersome model**
     ex) ensemble / a large generalized model
   - (pros) excellent performance
   - (cons) computationally expansive
   - can not be deployed when limited environments

2. **Student Network (S)**
   - **small model**
   - suitable for deployment
   - (pros) fast inference
   - (cons) lower performance than T

Idea



Knowledge Distillation의 기본 구성

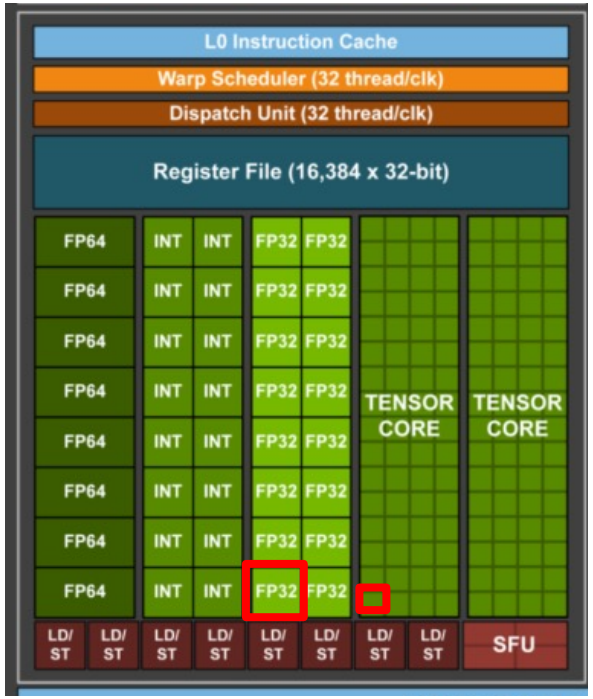
Various Model Compression Techniques

- Quantization

Quantization (양자화) 기술의 개요

❖ 본래 아날로그 신호를 디지털로 변환하는데 사용하나 AI에서는 모델을 경량화한다는 것을 의미

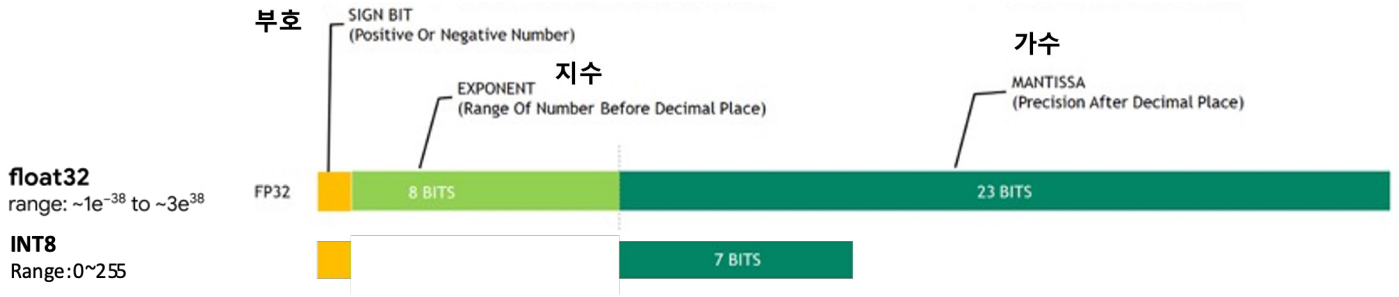
- AI Model의 경우 FP32로 구성하여 개발하고 있으나 좀더 낮은 비트로 개발하여 속도를 향상시키고자 함.



NVIDIA A100 Streaming Multiprocessor

	INPUT OPERANDS	ACCUMULATOR	TOPS	X-factor vs. FFMA	SPARSE TOPS	SPARSE X-factor vs. FFMA
A100	FP32	FP32	19.5	1x	-	-
	TF32	FP32	156	8x	312	16x
	FP16	FP32	312	16x	624	32x
	BF16	FP32	312	16x	624	32x
	FP16	FP16	312	16x	624	32x
	INT8	INT32	624	32x	1248	64x
	INT4	INT32	1248	64x	2496	128x

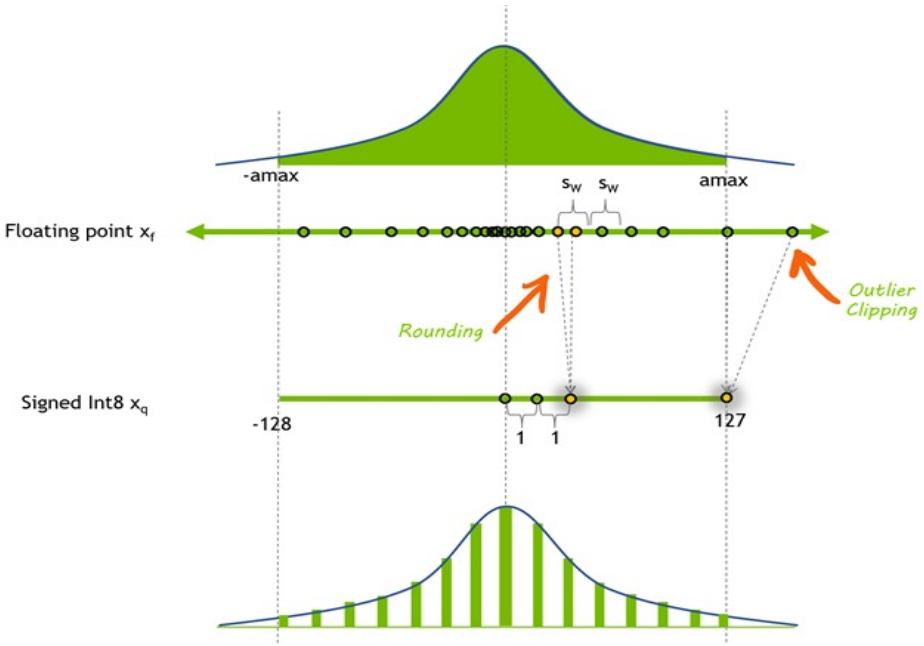
FP32로 계산된 모델을 INT8로 변환한다면 32배 성능 개선 가능



FP32 구조와 INT8 차이

Quantization (양자화) 기술의 개요

❖ 약간의 손실을 감안하며 모델의 사이즈를 줄이는 방법



FP32				INT8		
-3.57	4.67	-3.97	quantization →	33	255	22
-1.74	2.34	-1.76		82	192	81
-4.75	-0.06	3.07		1	127	212

$$f_q(x,s,z) = \text{Clip}(\text{round}(\frac{x}{s}) + z) \quad f_q(x,s,z) : \text{quantized value}$$

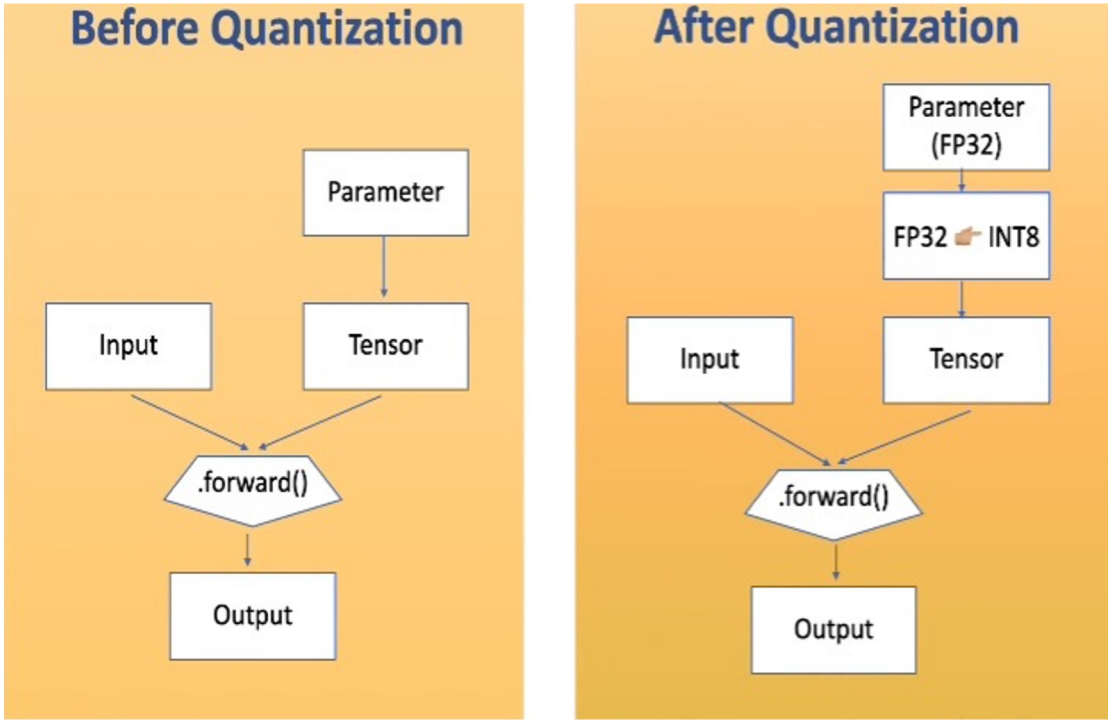
$$s : \text{scale} \quad s = \frac{\beta - \alpha}{\beta_q - \alpha_q} \qquad z : \text{zero-point integer} \quad z = \text{round}(\frac{\beta\alpha_q - \alpha\beta_q}{\beta - \alpha})$$

1. 전체 -4.75~4.67까지가 폭이라면, 9.42(4.67 – (-4.75)) 을 INT8의 영역으로 대응
2. 즉, INT8에서 1은 0.037. 9.42/255 =0.037
3. FP0의 수치가 변환된 중심값은 129. ((4.57*0-(-4.75)*255)/(4.67-(-4.75))
4. -3.57 값은 33으로 변환됨. round(-3.57/0.037) + 129= 33

- Clamping 과 Rounding 사용, symmetric/asymmetric의 적절한 사용 필요
- 메모리 Bandwidth 요구치를 맞출 수 있고, Resource를 적게 사용하게 되어 Energy를 절감 할 수 있음.
- 하지만, 낮은 비트너비로 인해 표현할 수 있는 모델 사이즈 축소: 정확도의 손실로 이어지는 문제 발생

Post Training Quantization

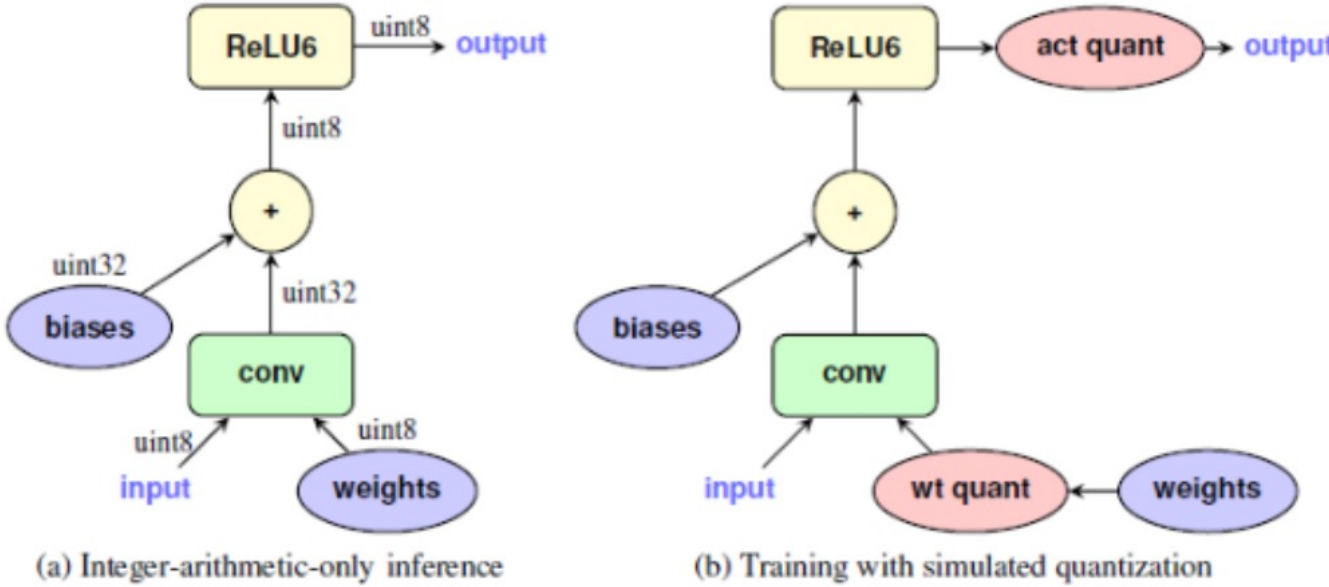
❖ Weight 값들에 대하여 Quantization을 하는 방법으로, 학습이 끝난 모델에 대해 Quantization Error를 최소화



- 단순하고, 빠르게 만들 수 있음. 또, 파라미터 Size가 큰 대형모델에 대해 정확도 하락의 폭이 작음
- 단점: 파라미터 Size가 작은 소형 모델에 대하여 정확도 하락이 큼

Quantization Aware Training

❖ Post Quantization 기법은 모델에 따라 성능 저하가 발생가능 ➔ Training을 진행하면서 Quantization을 함께 수행



- Quantization 이후 모델의 정확도 감소 폭을 최소화 할 수 있음.
- 단점: 모델 학습 이후 추가 학습이 필요함

More sophisticated Quantization Method

Paper

❖ 다양한 Quantization 방법 접근 및 개선

- "ZeroQuant: Efficient and Affordable Post-Training Quantization for Large-Scale Transformers.", 2022, Zhewei Yao
  - ➢ CNN 모델이 아닌 Transformer 모델까지 확장

https://arxiv.org/pdf/2206.01861.pdf

- “HAQ: Hardware-Aware Automated Quantization with Mixed Precision”, 2018, Kuan Wang
  - ➢ 모바일 디바이스, GPU 별 가속기 하드웨어 구조에 적합한 Mixed Precision 구조

https://arxiv.org/abs/1811.08886

- “ZeroQ: A Novel Zero Shot Quantization Framework”, 2020, Yaohui Cai
  - ➢ 4~8bit Integer를 활용한 Mixed Quantization Framework 방법 소개

https://arxiv.org/pdf/2001.00281.pdf

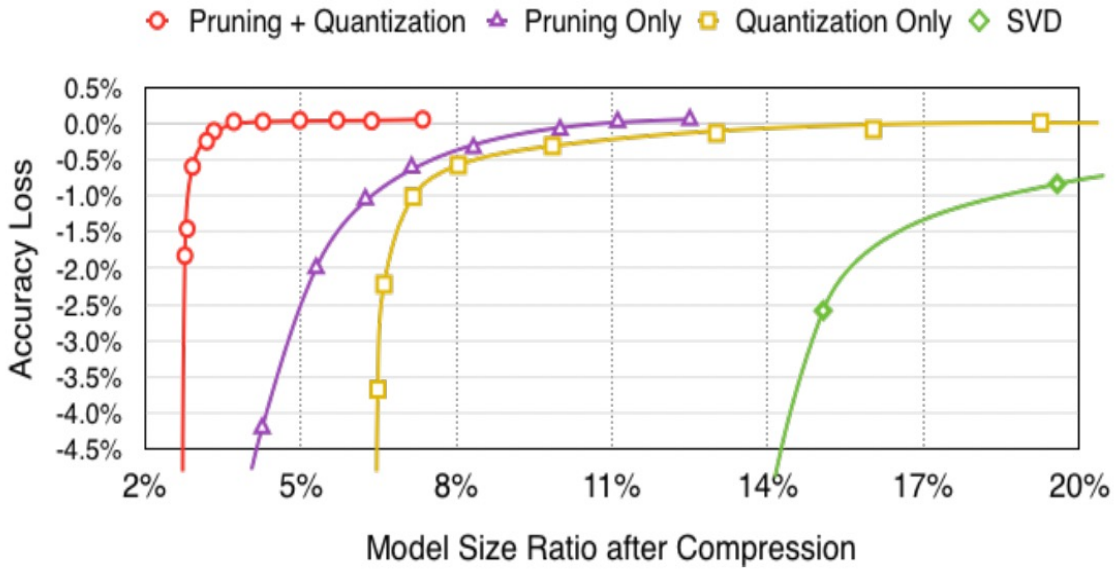
- “Post training 4-bit quantization of convolutional networks for rapid-deployment”, Ron Banner
  - ➢ Analytical Clipping for Integer Quantization(ACIQ)
  - ➢ Per channel bit allocation, Bias-Correction

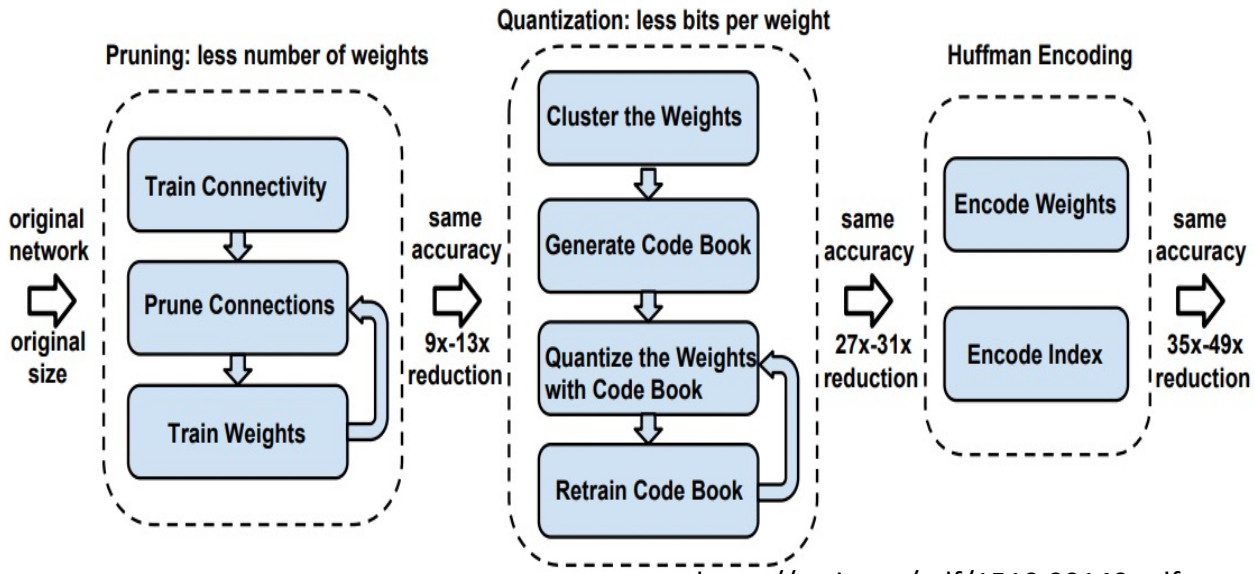
https://openreview.net/pdf?id=[illegible]

Pruning – Quantization 함께 수행

❖ Pruning과 함께 Quantization 및 Huffman Coding을 함께 하는 시도

- "DEEP COMPRESSION: COMPRESSING DEEP NEURAL NETWORKS WITH PRUNING, TRAINED QUANTIZATION AND HUFFMAN CODING", Song Han





https://arxiv.org/pdf/1510.00149.pdf

- Pruning과 Quantization 만으로도 적은 Accuracy Loss와 함께 처리속도를 49배까지 개선

Conclusion

- AI 모델 크기와 연산량이 커 감에 따라 Training과 Inference에서 요구되는 전산자원 및 비용이 증가 중
- 특히, Floating Point 32 bit로 학습된 AI model은 모바일 디바이스에 지나치게 클 수 있어 경량화가 요구됨.
- 모델 경량화 기술로 Pruning, Knowledge Distillation, Quantization의 방법 존재
- Quantization은 Post Training Quantization, Quantization Aware Training 방법 등이 있으며 지속발전 중이다.
- Pruning과 Quantization 등의 기술은 함께 사용 시 효과적으로 성능을 개선할 수 있다.

감사합니다.