Transformer를 활용한 인공신경망의 경량화 알고리즘 및 하드웨어 가속 기술 동향

Trends in Lightweight Neural Network Algorithms and Hardware Acceleration Technologies for Transformer-based Deep Neural Networks

김혜지 (H.J. Kim, hyejikim@etri.re.kr) 초거대AI반도체연구실 선임연구원
여준기 (C.G. Lyuh, cglyuh@etri.re.kr) 초거대AI반도체연구실 책임연구원/실장

ABSTRACT

The development of neural networks is evolving towards the adoption of transformer structures with attention modules. Hence, active research focused on extending the concept of lightweight neural network algorithms and hardware acceleration is being conducted for the transition from conventional convolutional neural networks to transformer-based networks. We present a survey of state-of-the-art research on lightweight neural network algorithms and hardware architectures to reduce memory usage and accelerate both inference and training. To describe the corresponding trends, we review recent studies on token pruning, quantization, and architecture tuning for the vision transformer. In addition, we present a hardware architecture that incorporates lightweight algorithms into artificial intelligence processors to accelerate processing.

KEYWORDS AI 반도체, 가지치기, 경량 딥러닝, 모바일 딥러닝, 양자화

Ⅰ. 서론

GPT[1]와 같은 인공신경망 기반 언어모델의 비약적인 발전은 영상-음향-언어 연구의 융합을 이끌어 최근 사물 인식을 비롯한 공간 인식, 이미지-객체 생성 알고리즘 분야의 혁신적 성능을 달성하고 있다[2-4]. 즉, 언어모델의 핵심 구조인 Attention 모듈[5]이 종래의 다양한 영상처리 알고리즘과 언어 모델 간의 가교 역할을 함으로써 새로운 융합을 이끌어내고 있다.

이와 같은 고성능 인공지능 알고리즘은 이미 사용자 주변에 친숙히 다가왔다. ChatGPT[6]와 Stable Diffusion[7]은 학교, 직장에서의 업무와 취미 활동으로 다양하게 활용되고 있다. 고성능 알고리즘이 사용자에게 친숙해질수록 AI 서버의 물리적 거리 또한 사용자와 가까워져야 요구 성능을 만족할 수

* DOI: https://doi.org/10.22648/ETRI.2023.J.380502
* 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 차세대지능형반도체 기술개발사업의 일환으로 하였음[2020-0-01308, 딥러닝 초소형 코어 어레이 기반 지능형 모바일 프로세서].

있다. 즉, 단말기(Edge Device)와 AI 서버의 속도-에너지-생산비용에 대한 시스템 효율을 극대화하기 위해 Transformer 기반 신경망 구조에 적용 가능한 경량화 알고리즘 및 가속화 하드웨어 구조 연구가 필요하다.

본고는 Transformer를 활용한 인공신경망 알고리즘의 경량화 방법론과 경량화에 특화된 하드웨어 가속기 구조에 대해 소개한다. 경량화 알고리즘은 종래의 CNN(Convolutional Neural Network) 구조에 널리 사용되고 있었다. 기존의 경량화 특성을 Transformer 구조에 어떻게 적용할지 2023 CVPR 학회에 소개된 Vision Transformer(ViT) 경량화 연구를 중심으로 관련 기법을 정리하며, 경량화 알고리즘의 추가적이고 반복적인 연산으로 인해 실시간 처리 속도가 저하되는 것을 방지하기 위해 2021 HPCA, 2023 ISCA 학회에 소개된 경량화 관련 하드웨어 가속기 연구를 요약 서술한다.

II. 신경망 경량화 기술의 배경

인공신경망의 경량화는 신경망의 불필요한 연산의 제거를 통해 메모리 사용량과 연산 복잡도를 감소시켜 궁극적으로 시스템의 에너지 소모량을 절감하고, 추론 및 학습 처리시간을 단축하는 신경망 최적화 방법론이다. 이 장에서는 인공신경망 연산의 제거 단위 및 자동화 방식에 따라 분류되는 대표적인 3가지 방법론을 소개한다.

1. 양자화(Quantization)

양자화는 신경망에서 사용되는 학습 파라미터 및 연산 입출력 데이터에 대하여 종래의 32비트 부동소수점(FP32) 연산기 대신 FP16, FP8 또는 8비트 정수형(INT8) 연산기를 활용하여 더 빠른 처리속도와 낮은 에너지 소모량을 달성하기 위한 경량화 방법이다. 이때 정확도 손실은 최소화하도록 데이터 포맷과 비트 수(Bit-Width)를 세밀하게 조절한다. 경량화에 의한 제거 대상은 데이터의 Bit-Width이다. 이 방법은 인공신경망을 처리하는 AI 프로세서에서 양자화에 사용되는 데이터 타입을 처리하는 하드웨어가 존재해야 실질적인 성능 향상을 체감할 수 있다. 따라서 AI 프로세서 구조 연구와 병행되는 경량화 분야이다.

2. 가지치기(Pruning)

가지치기는 경량화에 의한 제거 대상이 개별 데이터인 경우 Weight Pruning(i.e., Sparse) 분야로 분류되며, Matrix상의 행-열 또는 그룹화된 단위인 경우 Structured Pruning 또는 Channel Pruning으로 분류된다. 나아가 특정 레이어나 모듈 단위로 제거하는 경우 Layer Pruning 분야로 확장된다. Weight Pruning의 경우 0을 표현하는 데이터의 수가 증가하게 되므로, CSR(Compressed Sparse Row) 포맷과 같은 Spares-to-Dense 기법을 이용하여 하드웨어상의 처리효율을 증가시키는 기법과 연동되기도 한다. 즉, 가지치기 기법은 보다 큰 범위의 데이터를 신경망 구조적으로 제거하는 기법이므로, 별도의 하드웨어 구현 없이 직접적인 속도향상을 체감할 수 있다.

3. 경량 자동화(AutoML)

경량 자동화는 경량화의 대상이 되는 데이터의 포맷, 비트 수, Matrix 행-열의 개수 등에 대해 어떤 부분을 얼마만큼 제거할지에 대해 강화학습 또는 특정 Rule을 기반으로 자동으로 탐색하여 결정하는 기법이다. 경량화의 비용 함수를 처리 속도, 메모리 사용량, 또는 에너지 사용량 등 어떤 값을 최소화하

고자 하는지에 따라 경량화의 대상 부분을 얼마만큼 제거할지를 결정한다. 이는 경량화하는 과정에서 불필요한 시행착오를 최소화하여, 최적화 시간을 줄이고 경량화 후 성능을 극대화하는 데 목적이 있다.

Ⅲ. Transformer 기반 모델의 경량화

최근 AI 비전 분야에 Transformer를 활용한 ViT (Vision Transformer) 연구가 활발히 진행되고 있다. 이 장에서는 2023 CVPR 학회에 소개된 최신 Vision Transformer 경량화 알고리즘 연구를 소개한다.

1. GPUSQ

Transformer가 자연어처리 분야에 성공적으로 적용된 이후 비전 분야까지 활발히 확장되고 있다. Transformer는 내부적으로 Self-attention과 Cross-attention block이 적층으로 쌓여 있으며 고차원의 텐서 곱셈 연산으로 구성되어 있어서 GPU 또는 AI 가속기를 활용하여 연산 가속이 필요하다. NVIDIA GPU Tensor Core에서 제공하는 Sparse 및 Low-precision 기능을 활용하여 연산 가속이 가능하지만, 이를 위해선 많은 전처리 작업이 필요하며 Tensor Core와 같은 특정 하드웨어에 최적화된 경량화 연구는 많이 진행되고 있지 않다. GPUSQ[8]는 NVIDIA가 직접 참여하여 수행된 연구로, GPU-friendly Sparsity and Quantization 기법을 의미한다. NVIDIA A100 GPU와 AGX Orin의 Tensor Core를 대상으로 하여 그에 최적화된 연산이 가능하도록 신경망을 경량화한다.

가. Motivation

기존에 SP(Structured Pruning)와 양자화 각각에 대한 연구는 진행되고 있었지만, 두 가지 기능을 통합한 연구는 거의 진행되지 않았다. GPUSQ는 NVIDIA GPU에서 제공하는 2:4 Sparsity 기능을 최대 활용할 수 있도록 신경망을 Sparse 모델로 재학습하는 과정에서 FP16과 INT4, INT8 데이터 타입의 사용을 극대화하는 Quantization 및 Pruning 통합 신경망 압축 방식을 제안한다(그림 1 참고).

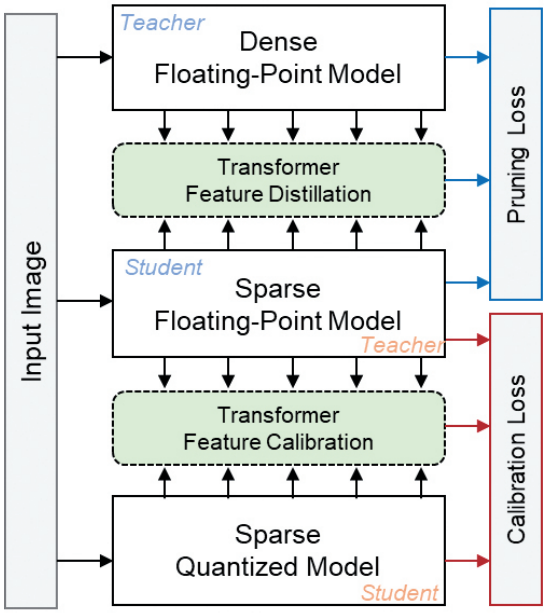

그림 1 GPUSQ 구조도

나. Approaches

학습된 ViT 모델을 2:4 SP기법을 이용하여 Patch Embedding, Linear Projection, Feed Forward Layers (i.e., GEMM) 연산에 대해 50% 비율로 0을 만든다. 이때 Sparse 모델을 재학습하는 과정은 KD(Knowledge Distillation) 기법을 활용한다. SP를 적용할 모델을 Student, Dense 모델을 Teacher로 정의하여 SP 모델의 결과가 Teacher 모델의 결과를 모방할 수 있도록 재학습을 수행한다. 이때 모든 Transformer 블록에 대해 유사성을 탐색하지 않고, Critical 특징맵

을 설정하고, 그에 대한 값들이 집중적으로 유사해지도록 KD를 이용한다. 이어서, 정수형 데이터 타입의 양자화 모델로 재학습은 KD와 QAT(Quantization-Aware Training)을 통합하여 활용한다. SP모델을 Teacher로 하고 양자화 모델 사이의 Calibration Loss 항을 학습에 추가하여 최종 Loss가 감소하도록 FP16 SP모델에서 INT4 또는 INT8를 사용하는 양자화 모델로 변환한다. 해당 연구가 적용된 GPUSQ-ViT 모델은 ImageNet Classification, COCO Detection, ADE20K Segmentation 응용에 대해 약 6.4~12.7배 모델 사이즈 감소 및 30~62배 연산량 감소 효과를 달성했다. 또한, 실제 A100 GPU에서 수행했을 때 1.39~1.79배 지연시간 감소 및 3.22~3.43배 처리량 증가 효과를 보였다.

2. NIPQ

NIPQ(Noise proxy-based Integrated Pseudo-Quantization)[9]는 신경망의 정수형 양자화에 관련된 연구로, 학습된 신경망 모델의 파라미터(Weight)와 특징맵(Activation) 데이터를 3~8Bit 사이의 정수형으로 표현하여 추론 속도를 향상시키는 복합 데이터 타입 양자화 방법론이다. 본 연구는 PQT(Pseudo-Quantization Training) 기법의 한 분야로, 데이터의 양자화 과정에 Noise Proxy와 자동화 기법을 도입하여 4Bit 이하의 낮은 비트 수를 신경망의 레이어마다 복합적으로 적용하는 경우에도 학습에 대한 안정성을 유지하도록 했다.

가. Motivation

STE(Straight-through Estimator)는 양자화 파라미터(e.g., 최대-최솟값, 중심값)를 학습 과정에서 결정하기 위해 적용되는 함수로, 미분 불가능한 함수에 근사치를 부여하여 Gradient Back-Propagation을 가능하게 하는 기법이다. 그러나 STE는 데이터의 표현 가능한 정밀도가 매우 낮아지면 학습이 불안정해지는 특성이 관찰되면서, 양자화 노이즈를 부여하는 PQT 기법이 대안으로 사용되고 있다. 본 연구는 기존의 STE와 PQT가 가지는 불안정성에 의한 양자화 한계를 분석하여, 보다 양자화 오차를 최소화하면서 저정밀도 복합 데이터 타입 학습의 안정성을 향상하는 방법론을 제안한다.

나. Approaches

NIPQ는 기존의 PQT에 Noise Sampling 기법과 Boundary Truncation 기법을 통합하여 저정밀도 정수형 양자화에 대해서 안정적인 학습이 가능한 파이프라인을 제안했다(그림 2 참고). Min-Max Boundary와 데이터의 비트 수는 양자화 오차와 연산 복잡도를 최소화하도록 PQT 과정 중 학습을 통해 결정되며, 양자화는 학습 파라미터와 특징맵 모두 적용 가능하다. 구체적으로, 비트 수를 결정하는

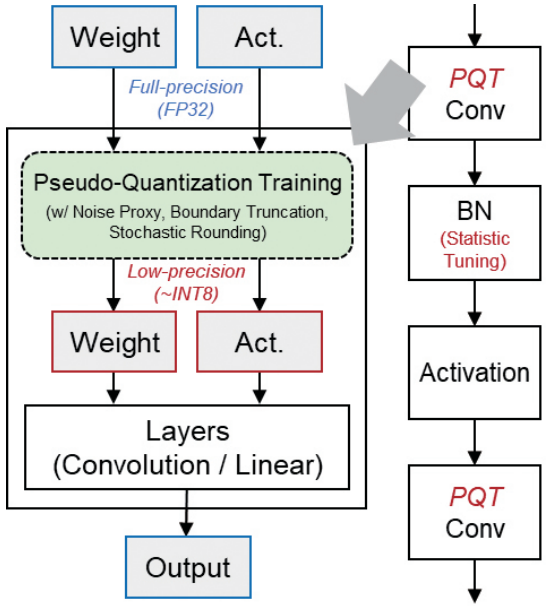

그림 2 NIPQ 구조도

과정에서 후보 값에 대한 정밀도를 향상하기 위해 Stochastic Rounding이 사용되었다. 양자화 노이즈는 양자화 오차분포를 따르도록 샘플링된 값을 부여해야 하지만, 본 연구는 학습 속도의 저하를 방지하기 위해 Uniform 분포의 노이즈를 사용했다. 그로 인한 인식 성능 저하는 최종적으로 QAT를 적용하여 충분히 회복 가능하며, NIPQ 파이프라인이 모두 끝난 뒤 최종적으로 BN(Batch Normalization) 레이어의 Statistics를 변경된 데이터 분포에 맞도록 업데이트하여 안정된 결괏값을 도출했다.

3. DepGraph

DepGraph[10]는 경량화를 위해 정보를 제거하는 과정에서 신경망의 레이어 간 상호 의존성을 분석하고, 실질적인 데이터의 중요도를 평가하기 위해 DepGraph(Dependency Graph)를 생성하여 경량화에 활용하는 기법이다. 신경망 전반에 걸쳐 중요도를 탐색하는 기법은 종래의 CNN 중심의 신경망 구조를 넘어서 RNN(Recurrent Neural Network), GNN(Graph Neural Network) 및 Transformer 모델에 대해 일반화된 개념으로 경량화 알고리즘을 적용할 가능성을 제시했다.

가. Motivation

Structured Pruning 방식은 신경망에서 채널 혹은 필터 단위의 넓은 범위로 그룹화된 학습 파라미터들에 대해 불필요한 데이터를 제거하여 연산을 가속하는 데 사용되고 있다. 그러나 어떤 방식으로 파라미터를 그룹화해야 인식 성능이 저하되지 않고 연산을 가속할 수 있는지는 신경망 아키텍처의 종류에 따라 일반화하기 어려운 문제였다.

나. Approaches

DepGraph는 다양한 신경망 구조에 적용 가능한 일반화된 SP 기법을 제안했다(그림 3 참고). 기존 연구들은 경량화 과정에서 서로 다른 신경망 레이어들을 동시에 Pruning하거나, 한번 제거된 파라미터는 모든 레이어에 일관되게 불필요하다는 전제하에 경량화 규칙을 설정했다. 이러한 판별 기준으로 인해 얕은 경량화에도 인식 성능의 저하가 크게 발생하는 경우가 많았다. 해당 연구는 신경망 레이어의 입출력을 세분화한 표기법을 제시하여, 레이어 간(Inter-Layer) 의존성과 레이어 내부(Intra-Layer) 의존성에 대한 관계도를 생성한다. 관계도상의 의존성에 기반하여 각 연결 부분마다 별도의 Pruning 규칙이 적용된 파라미터 그룹이 형성되고, 그룹 단위 데이터에 대해 Sparse 기법을 적용하여 경량화에 대한 민감도를 평가한다. 이러한 방식은 기존의 레이어마다 독립적으로 경량화 민감도를 평가하던 방식과 비교하여, 보다 전역적 신뢰도가 높은 경량화 가중

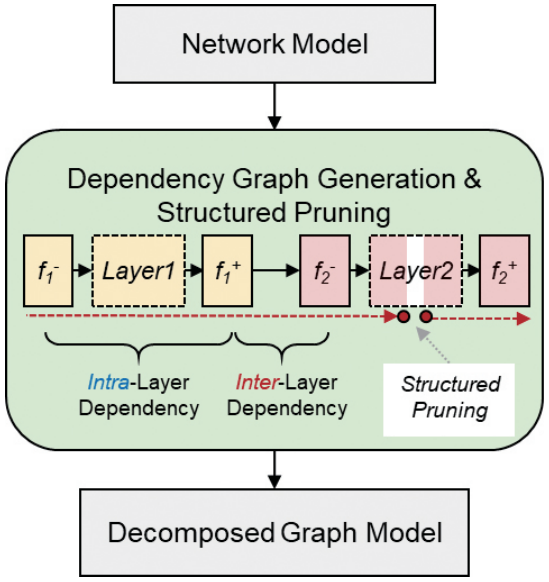

그림 3 DepGraph 구조도

치를 생성함에 따라 높은 신경망 압축률을 달성할 수 있다.

이 방식은 다양한 CNN 계열 모델과 ViT, 3D Point Cloud 네트워크(DGCNN) 및 언어모델 등에 적용하여 다양한 신경망 아키텍처에 대해 일반적인 방식으로 경량화가 가능함을 실험적으로 보였다.

4. TPS

TPS(Token Pruning and Squeezing)[11] 기법은 Transformer 블록의 Token 정보를 그 자체로 제거하지 않고 보존 영역에 해당하는 주변 Token에 정보를 압축 분산하여 전체 연산량을 줄이면서 정보 손실을 최소화하는 기법이다. 특히 Vision 분야의 경우 2D 이미지는 공간적 영역 정보가 중요함에 따라, 정보의 중요도가 낮은 Token이라도 객체를 형성하는 데 의미 있는 정보로 활용될 수 있다. 이러한 공간적 특성을 고려하여 본 연구는 Token의 제거보다 Token을 압축하는 방식에 집중한 연구를 제시했다.

가. Motivation

ViT의 경량화를 위한 기법으로 Transformer 블록에 적용되는 Token Pruning 기법이 널리 사용되고 있다. 이 방식은 중복된 Token 값들에 대해 불필요한 정보를 제거하여, 제거된 Token 값과 관련된 하위 연산 및 메모리 사용을 감소시킴으로써 높은 속도향상을 달성할 수 있는 기법이다. 그러나 어떤 Layer의 Token을 얼마만큼 제거하는지에 따라 큰 성능 편차가 나타나며, 제거된 Token 또한 데이터의 특성에 따라 정확도에 중요한 영향이 있음이 실험을 통해 확인되고 있다. 해당 연구는 정보의 중요도가 낮은 Token을 직접 제거하지 않고, 주변 Token에 정보를 분산시켜 융합형 Token 그룹을 정의하는 방법론을 제안한다(그림 4 참고).

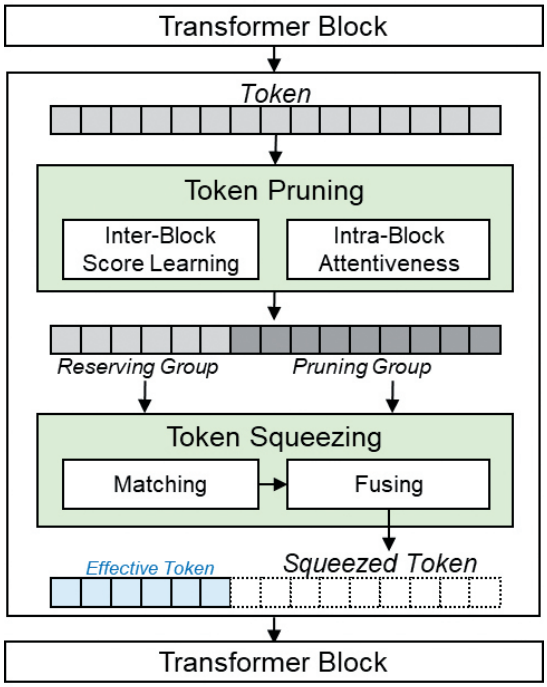

그림 4 TPS 구조도

나. Approaches

TPS는 Token Pruning과 제거된 정보를 주변 Token으로 압축하는 Squeezing 기법을 동시에 적용했다. 우선 Token Pruning은 Transformer 블록 전반에 걸쳐 학습된 Token Scoring 및 Masking 기법을 사용하는 dTPS 방식과 Transformer 블록 내부에서 Attention 연산으로 계산된 Token Attentiveness를 활용하는 eTPS 방식으로 나뉘며, Token의 중요도를 판단하는 기법은 기존의 연구를 따르도록 설계되었다[12-14]. 이 과정을 통해 Token은 제거 그룹과 보존 그룹으로 나뉜다. 제거 그룹에 속한 Token은 보존 그룹의 값 사이의 유사성을 계산하여 보존 그룹에서 자신과 가장 연관성이 높은 Host Token을 설정한다. 이후 앞서 Pruning 과정에서 계산되었던 Token의 중요도 가중치 정보를 활용하여 Host Token과 그에 관련된 제거 그룹 Token 값들을 융합한다.

해당 연구는 DeiT-Tiny 모델의 연산량을 35% 감소하는 동시에 ImageNet 분류 정확도는 1~6%가량 향상시킬 수 있다. 나아가 압축된 DeiT-Small 모델은 그보다 작은 DeiT-Tiny 모델보다 더 빠르면서도, 인식 정확도는 4.78% 상승하여 Token Squeezing 기법이 신경망을 효과적으로 압축할 수 있음을 실험적으로 보였다.

5. NViT

NViT[15]는 종래의 ViT 구조를 기반으로 각 Transformer 블록 전반에 걸쳐 각 연산 블록 특성에 맞는 최적의 모듈 크기를 탐색하여 개별 튜닝하는 전역적 경량화 파이프라인이다(그림 5 참고). NViT를 이용한 경량화는 ViT-Family와 동일한 규모에서 더 빠른 추론 속도와 높은 인식 정확도를 달성하도록 파라미터를 재분배한다. 이 과정에서 NVIDIA A100 GPU와 같은 특정 하드웨어의 구조적 특성을 반영하여 추론 속도가 개선된 NViT-Family를 생성하는 방법론을 제시했다.

가. Motivation

기존의 ViT 모델은 모든 Transformer 블록에 걸쳐 균일한 차원을 가지고 있다. ViT-Family를 구성함에 있어 마찬가지로 균일한 차원 비율로 규모를 키우거나 줄이는 방식으로 신경망 구조를 정의하고 있다. 이 과정은 규모를 조금씩 키우거나 줄여가며 학습 성능을 평가한 뒤 최종 구조를 결정하는 설계자의 경험에 의존된 방식이므로, 도출된 신경망 구조는 불필요한 연산과 비용을 야기할 수 있다. 따라서 신경망의 규모를 결정하는 결정적 구조적 특징을 도출하고, 이를 기반으로 한 구조 설계 자동화 방법론이 필요하다.

나. Approaches

NViT는 Transformer 내부의 연산구조를 Structured Pruning에 용이하도록 Attention 블록과 Multi-Layer Perception 블록을 파라미터 매개변수로 구조화함과 동시에, 병렬 연산이 가능하도록 Q, K, V Matrix의 의존성을 일부 분리한 Prunable Component 블록을 정의한다. 그리고 다수의 Transformer 블록에 대해 특정 블록의 연산이 강조되거나 차원이 축소되도록 개별적인 차원 최적화를 수행하는 Hassian 기반의 파라미터 재분배 규칙을 적용했다. 기존 연구에서 데이터 그룹의 Hessian-Norm이 작을수록 손실 표면이 평평한 특징을 갖고, 이는 데이터 그룹이 Pruning되더라도 손실값에 작은 변화를 가져왔다[16-18]. 이에 기반하여, 본 연구는 경량화 과정에서 간소화된 Hessian-Norm을 제안하고, 파라미터 그룹에 대한 중요도를 평가하는 기준으로 사용했다. 또한, 실제 추론 속도를 경량화 함수에 반

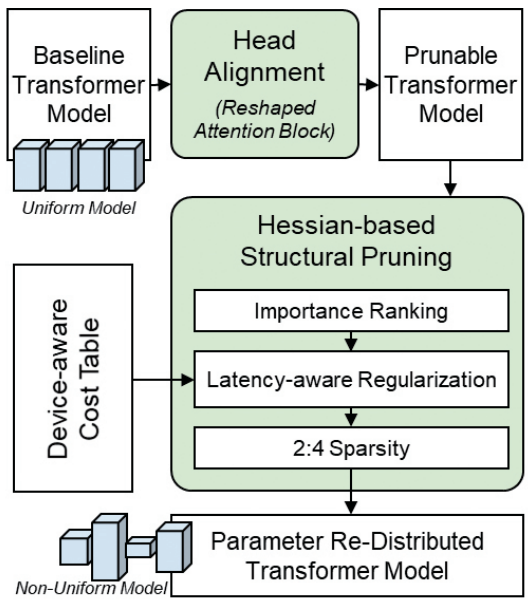

그림 5 NViT 구조도

영하기 위해 특정 하드웨어 추출한 Latency-Cost 테이블을 활용하여 파라미터의 중요도 평가 과정에 정규화된 값으로 부여했다.

Ⅳ. 경량화를 위한 AI 반도체 기술

신경망의 연산을 가속하기 위해 경량화 알고리즘을 도입하고 있으나, 부가적인 알고리즘으로 인하여 오히려 전반적인 연산 시간은 증가하는 현상을 초래할 수 있다. 특히, 특징맵(i.e., Activation)을 위한 경량화 알고리즘은 매번 새로 연산될 필요가 있다. 이 장에서는 실질적으로 경량화에 의한 복잡도 감소와 속도 향상을 달성하기 위해 알고리즘에 반복적으로 요구되는 연산을 NPU 하드웨어에 탑재하여 경량화로 인한 지연시간을 최소화하는 연구를 소개한다.

1. SpAtten

SpAtten[19]은 2021 HPCA 학회에 소개된 연구로, Attention 블록 연산과 경량화 알고리즘을 가속하기 위한 NPU 하드웨어 구조를 제안했다. 경량화는 Sparsity와 Token-Head Pruning을 수행하는 3가지 점진적 최적화 알고리즘이 적용되었으며, 해당 연산을 하드웨어 자체적으로 가속하여 실시간 처리 속도를 향상시키는 가속기 구조 연구까지 통합하여 진행됐다.

가. Motivation

자연어처리 분야의 핵심인 Attention 메커니즘은 복잡한 데이터 이동과 낮은 연산 밀도 특성으로 기존의 CPU 및 GPU와 같은 범용 플랫폼에서 처리하는 것보다 전용의 하드웨어를 통해 처리하는 것이 처리 속도 및 메모리 활용성 면에서 높은 효율을 달성할 수 있다. 따라서 Attention 블록의 가속기는 실시간으로 변하는 입력 데이터에 대해 인식 성능 저하를 최소화하면서 메모리와 연산 시간, 그리고 연산기 내부에서의 데이터 이동을 상황에 따라 최적 제어할 수 있는 연산기 구조 개발이 필요하다.

나. Architecture

SpAtten은 언어모델에 적용 가능한 하드웨어 구조를 제안한다(그림 6 참고). 경량화는 입력 문장에서 중요하지 않다고 판단된 Token과 Head에 집중적으로 적용되며 이는 문장에 따라 실시간으로 달라진다. 상위 레이어에서 제거 대상으로 분류된 Token 및 Head 정보는 하위 레이어까지 연쇄적으로 반영되어 전반적인 구조적 관점에서 제거된다. 하드웨어는 어떤 Token과 Head가 중요한지 순위를 결정하는 Top-K 엔진을 각 연산에 개별적으로 지원하여 경량화 연산을 가속한다. 또한, 메모리 사용량을 줄이기 위해 점진적으로 데이터 비트 수를 증감하는 양자화 알고리즘 및 양자화 판별 엔진을 추가했

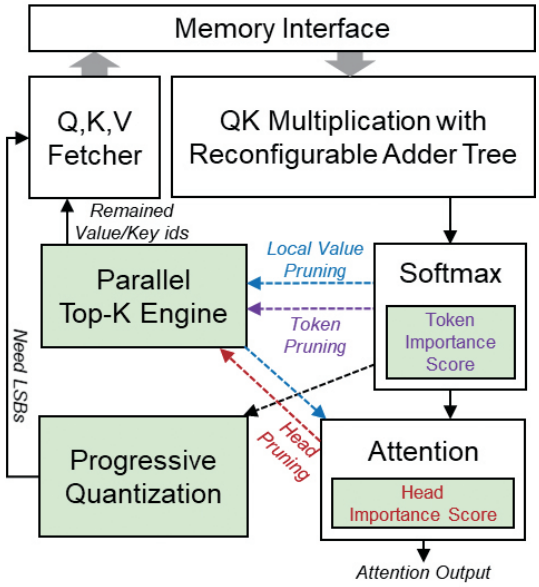

그림 6 SpAtten 구조도

다. Attention 블록의 Softmax 연산 입력에 해당하는 Q, K Matrix에 양자화를 적용하는 경우, Softmax 연산 결괏값이 특정 Token에 편향된 큰 값으로 존재한다면 양자화에 의한 에러는 크게 절감된다. 따라서 이 경우 데이터의 MSB 위주의 낮은 정보량만 사용 가능하다. 반면 전체 Token에 균일한 연산 결괏값이 분포한다면, MSB 데이터에 LSB 쪽 데이터를 추가하여 양자화 비트 수를 증가시켜 정확도 손실을 보상한다. 본 연구는 양자화 판별 하드웨어를 탑재하여 실시간으로 최적 비트 수를 탐색한다.

2. FACT

FACT(FFN-Attention Co-optimized Transformer)[20] 구조는 2023 ISCA 학회에 소개된 연구로, Attention 블록 연산과 전처리에 해당하는 QKV 행렬 생성 연산, 그리고 FFN(Feed-Forward Network) 연산으로 구성된 Transformer의 세 가지 주요 모듈을 최적화하는 Sparsity 및 Token Pruning 경량화 알고리즘을 제안하였으며, 이를 위한 전용의 하드웨어 가속기를 도입한 NPU 아키텍처이다(그림 7 참고).

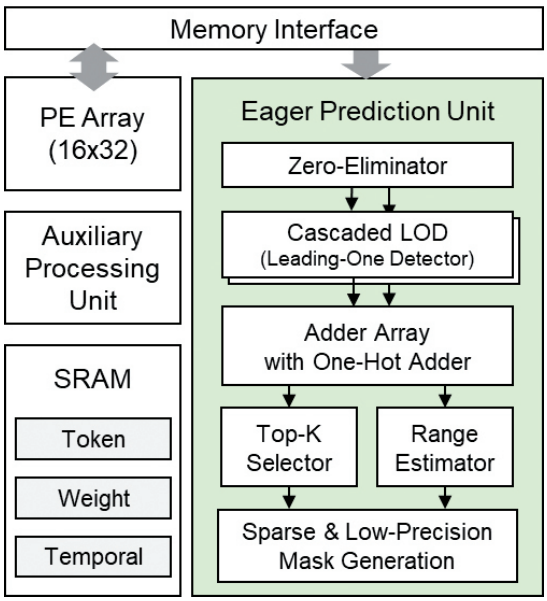

그림 7 FACT 구조도

가. Motivation

기존의 Transformer 관련 하드웨어 가속기 연구는 Attention 연산에 초점을 맞추고 있다. 그러나 FACT 저자는 Token 길이가 극도로 긴 경우에 한해서만 Attention 연산의 전력 점유율이 높은 특징을 발견했다. 즉, 대부분의 신경망 구조는(e.g., ViT, BERT, GPT-2, Swin) Token 길이가 1k 미만이며, 이 경우엔 QKV 생성 연산 및 FFN 연산의 전력 점유율이 전체의 대부분을 차지한다. 따라서 본 연구는 Sparsity 및 Token Pruning을 적용하여 Token 길이를 줄이고, 경량화 전용 하드웨어와 QKV 생성 및 FFN 연산 가속기에 집중한 저전력 NPU를 제시한다.

나. Architecture

FACT는 QKV Matrix의 경량화를 위한 희소화 마스크와 저정밀도화 마스크를 생성하는 EP(Eager Prediction) 알고리즘을 제안한다. 해당 마스크 생성 알고리즘은 하드웨어상에서 처리되므로, 연산 복잡도를 줄이기 위해 데이터를 로그계열로 변환하여 곱셈연산 없이 덧셈연산으로 경량화 마스크를 생성한다. QKV Matrix는 앞서 계산된 희소화 마스크를 가중치로 고려하여 Top-K에 해당하지 않는 값들에 대해 0으로 제거한다. 이 과정은 Softmax의 결괏값은 입력 QK Matrix에서 절댓값이 큰 값의 위주로 계산되는 특성을 따라, 가중치가 낮은 값을 0으로 설정해도 결괏값에 큰 오차가 발생하지 않는 특징을 차용했다. 저정밀도화 마스크는 FFN의 입력인 QK Matrix에 적용된다. 마스크를 가중치로 고려하여 Top-K 엔진을 통해 상위 가중치를 갖는 데이터는 INT8 데이터 타입을 사용하고, 가중치가 낮은 데이

터는 INT8의 MSB 영역으로 구성된 INT4 데이터 타입을 사용한다.

Ⅴ. 결론 및 맺음말

이제 Efficient Vision AI의 연구 흐름은 CNN 위주의 신경망 모델에서 Transformer 기반 ViT 모델을 경량화하는 주제로 넘어가고 있다. 나아가, 기존 CNN 모델에 사용되던 다양한 경량화 기법을 ViT 뿐만 아니라 다양한 응용 신경망 모델에 일반화되어 사용할 수 있는 범용 경량화 연구 분야 또한 발전하고 있다. 이는 시간이 지날수록 다양한 모델이 등장하고, 모델 다양성이 증가하더라도 여전히 경량화 기법은 필수적이며 보다 간편하게 적용하기 위한 연구들의 필요성이 증가하고 있음을 시사한다. 경량화는 모델의 추론 및 학습에 널리 사용되고 있는 만큼, AI 프로세서에서 경량화 연산을 지원하여 Transformer를 효율적으로 처리하는 가속기 연구 또한 중요한 시점이다.

약어 정리

BERT	Bidirectional Encoder Representations from Transformer
GPT	Generative Pre-trained Transformer
GPU	Graphics Processing Unit
INT	Integer Format
NPU	Neural Processing Unit
QKV	Query-Key-Value Matrix

참고문헌

[1] T. Brown et al., "Language models are few-shot learners," in Proc. NeurIPS 2020, (Vancouver, Canada), Dec. 2020, pp. 1877-1901.

[2] C.H. Lin et al., "Magic3d: High-resolution text-to-3d content creation," in Proc. IEEE/CVF CVPR 2023, (Vancouver, Canada), June 2023, pp. 300-309.

[3] U. Singer et al., "Make-a-video: Text-to-video generation without text-video data," arXiv preprint, CoRR, 2022, arXiv: 2209.14792.

[4] R. Huang et al., "Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models," arXiv preprint, CoRR, 2023, arXiv: 2301.12661.

[5] A. Vaswani et al., "Attention is all you need," in Proc. NIPS 2017, (Long Beach, CA, USA), Dec. 2017.

[6] https://openai.com/blog/chatgpt

[7] R. Rombach et al., "High-resolution image synthesis with latent diffusion models," in Proc. IEEE/CVF CVPR 2022, (New Orleans, LA, USA), June 2022, pp. 10684-10695.

[8] C. Yu et al., "Boost Vision Transformer with GPU-Friendly Sparsity and Quantization," in Proc. IEEE/CVF CVPR 2023, (Vancouver, Canada), June 2023, pp. 22658-22668.

[9] J. Shin et al., "NIPQ: Noise proxy-based integrated pseudo-quantization," in Proc. IEEE/CVF CVPR 2023, (Vancouver, Canada), June 2023, pp. 3852-3861.

[10] G. Fang et al., "Depgraph: Towards any structural pruning," in Proc. IEEE/CVF CVPR 2023, (Vancouver, Canada), June 2023, pp. 16091-16101.

[11] S. Wei et al., "Joint token pruning and squeezing towards more aggressive compression of vision transformers," in Proc. IEEE/CVF CVPR 2023, (Vancouver, Canada), June 2023, pp. 2092-2101.

[12] Y. Rao et al., "Dynamicvit: Efficient vision transformers with dynamic token sparsification," in Proc. NeurIPS 2021, (Virtual-only), Dec. 2021, pp. 13937-13949.

[13] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," arXiv preprint, CoRR, 2016, arXiv: 1611.01144.

[14] L. Youwei et al., "Not all patches are what you need: Expediting vision transformers via token reorganizations," arXiv preprint, CoRR, 2022, arXiv: 2202.07800.

[15] H. Yang et al., "Global vision transformer pruning with hessian-aware saliency," in Proc. IEEE/CVF CVPR 2023, (Vancouver, Canada), June 2023, pp. 18547-18557.

[16] S.M. Moosavi-Dezfooli et al., "Robustness via curvature regularization, and vice versa," in Proc. IEEE/CVF CVPR 2019, (Long Beach, CA, USA), June 2019, pp. 9078-9086.

[17] Y. Huanrui et al., "Hero: Hessian-enhanced robust optimization for unifying and improving generalization and quantization performance," arXiv preprint, CoRR, 2021, arXiv: 2111.11986.

[18] Y. Shixing et al., "Hessian-aware pruning and optimal neural implant," in Proc. IEEE/CVF WACVi 2022,

(Waikoloa, HI, USA), Jan. 2022, pp. 3880-3891.

[19] H. Wang, Z. Zhang, and S. Han, "Spatten: Efficient sparse attention architecture with cascade token and head pruning," in Proc. IEEE HPCA 2021, (Seoul, Rep. of Korea), Feb. 2021.

[20] Y. Qin et al., "FACT: FFN-attention Co-optimized transformer architecture with eager correlation prediction," in Proc. ISCA 2023, (Orlando, FL, USA), June 2023, pp. 1-14.