AI Accelerator

영남대학교
차세대 컴퓨터 시스템 연구실
석사과정생 이원호

ConfuciuX: Autonomous Hardware Resource Assignment for DNN Accelerators using Reinforcement Learning

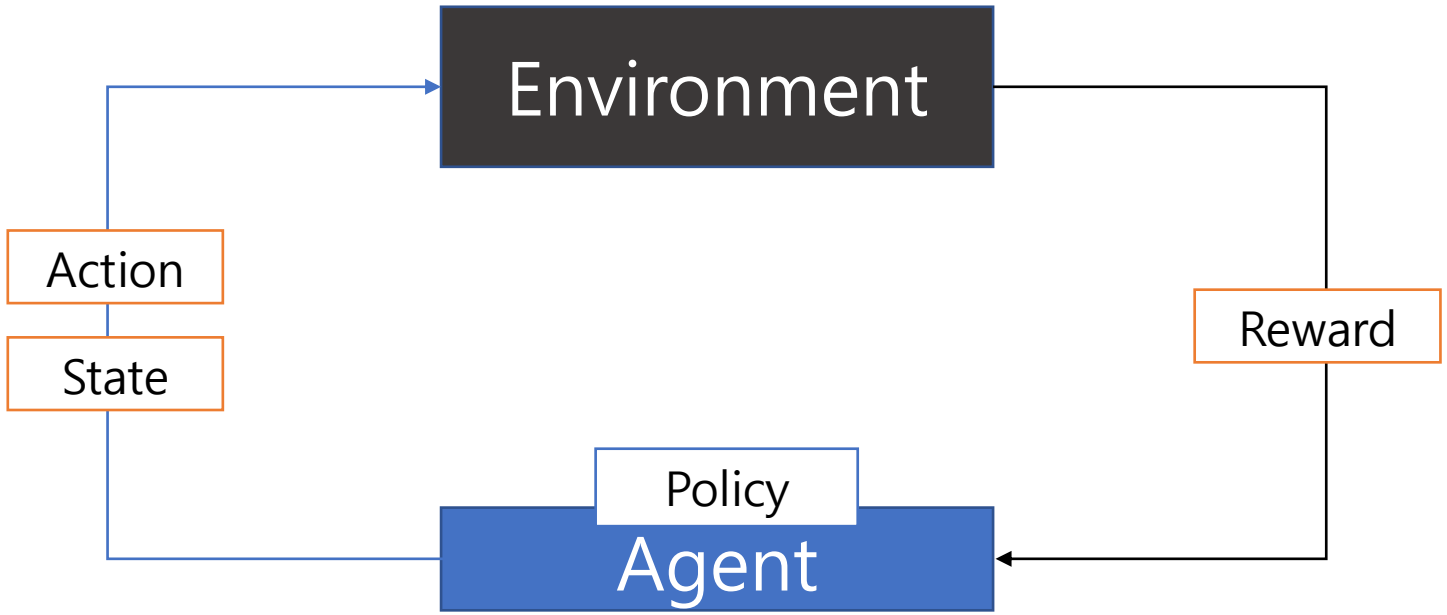
2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), 2020

차세대 컴퓨터 시스템 연구실

이원호

Reinforcement Learning

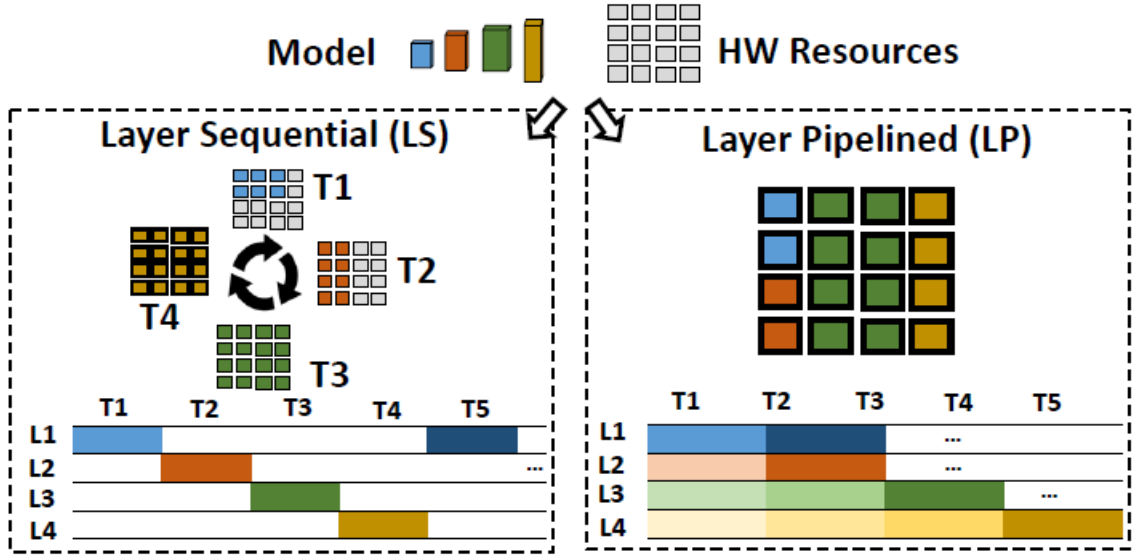
- 머신 러닝의 한 갈래로 데이터 없이 Agent가 환경과 상호 작용하며 그 보상으로 Policy 학습하는 방식



DNN Deployment scenarios

- Layer Sequential (LS)
  - Layer 별로 매핑하고 실행하는 방식
  - On-chip 에 적합하지 않음
  - 클라우드 플랫폼에서 활용함

- Layer Pipelined (LP)
  - 전체 Layer를 매핑하고 실행하는 방식
  - 최적화된 모델에 대해 유용함
  - IoT 플랫폼에서 활용함



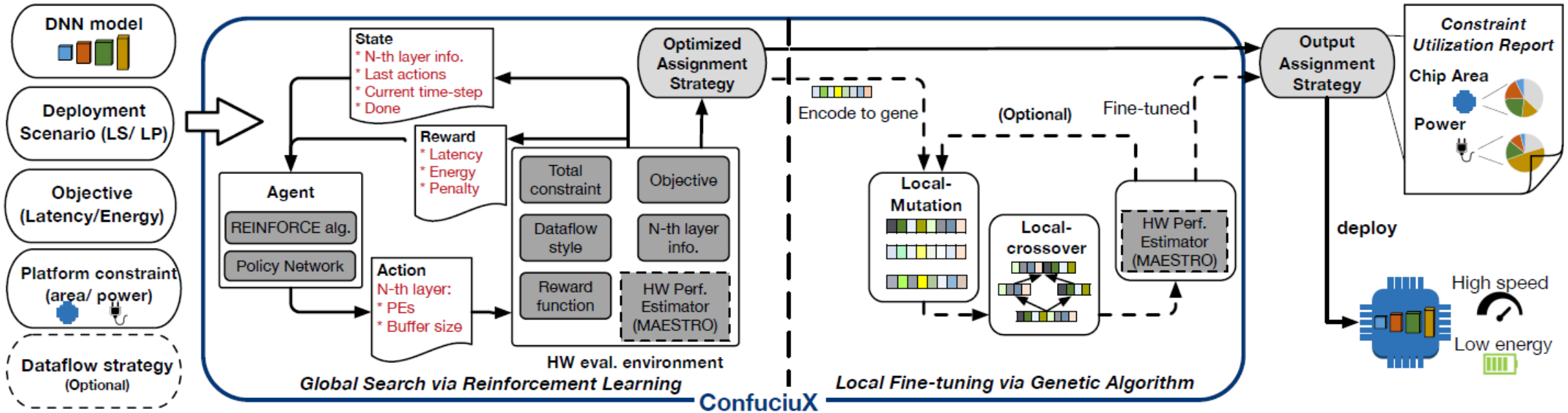
문제점 및 동기

- DNN 연산 중 DRAM으로의 데이터 이동을 줄이기 위해 data reuse는 DNN 가속기에서 효율성을 보여줌
- 어떤 데이터(activation/weights/output)를 재사용할 지는 가속기의 dataflow에 따라 결정됨
- 이에 따라 다양한 dataflow에 대한 탐색과 비교는 많았음
- DNN에 대한 area/power의 자원 제약을 충족하면서 성능/에너지에 최적화할 수 있는 dataflow가 주어지면 이에 따른 on-chip hardware resource allocation strategy에 대한 연구는 부족함
- 연산과 메모리 사이의 균형을 맞추기 위한 설계 공간의 선택은 증가함
  - 여러 탐색을 통한 수동 조정이 어려움
  - 다른 DNN과 layer의 형태에 따라 다른 재사용 양을 고려할 때 특정 휴리스틱을 생각하기 어려움

제안사항

- 주어진 모델과 dataflow 타입에 따라 최적화된 하드웨어 자원 할당 탐색
- 탐색 과정을 지도하기 위해 REINFORCE 알고리즘 활용
  - 강화학습 환경으로 Data-centric notation을 활용한 MAESTRO 도구 활용
  - 자세한 하드웨어 성능 비용 모델을 보상으로 활용
- 추가적인 fine-tuning을 위해 유전 알고리즘 활용

Overall Architecture



Algorithm-hardware Co-design of Attention Mechanism on FPGA Devices

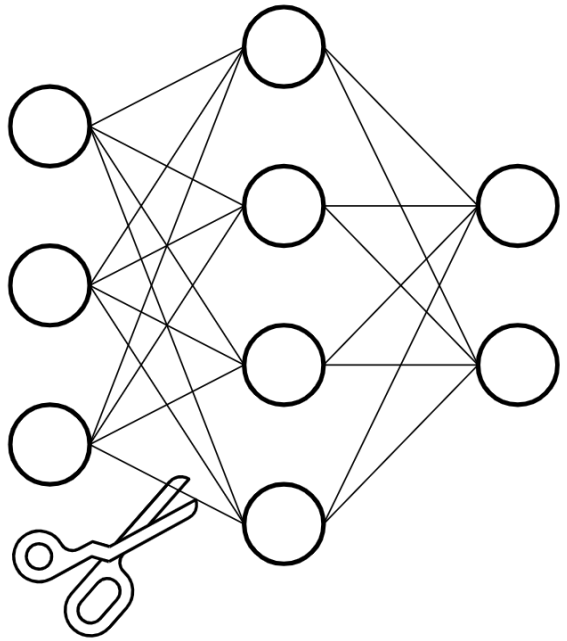
ACM Transactions on Embedded Computing Systems (TECS), 2021

차세대 컴퓨터 시스템 연구실

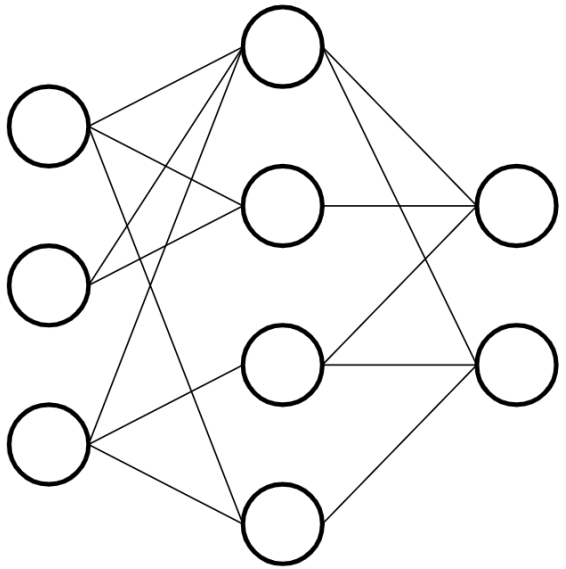
이원호

Pruning

- 인공지능 모델 추론 속도를 향상시키기 위한 최적화 방법
- 해당 연산과 연관없는 가중치를 비활성화 시켜 연산량을 줄이는 방식
  - 필요없는 값을 0으로 변환시켜 전달되지 않도록 함
- 0으로 변환시켜 모델 가중치의 sparsity에 따라 압축 가능



Before pruning



After pruning

문제점 및 동기

- Multi-head self-attention은 특징 추출 및 순차 데이터 분석에서 널리 활용됨 (기계 번역, 언어 모델링, 이미지 처리 등)
- 기존 시스템에서 이런 attention 메커니즘 기반의 대규모 파라미터와 정교한 모델 구조를 자원이 제한적인 장치에 배포하기 위해 모델 사이즈를 줄이려는 시도가 있음
  - 특정 작업을 위해 더 작은 모델로의 최적화 -> robustness 훼손
  - 모델의 robustness 훼손을 없애기 위해 Model compression 활용
- 실행 시간동안 FPGA에서 데이터 버퍼 할당과 커널 계산이 고정적인 문제 (FPGA의 자원 효율성을 극대화하기 위함)
- 모델 압축을 진행했을 때 weight의 형태와 크기가 변형됨 -> 효율성 저하
- FPGA에서 압축된 가중치에 대해 버퍼 설계와 커널 계산이 비효율적임

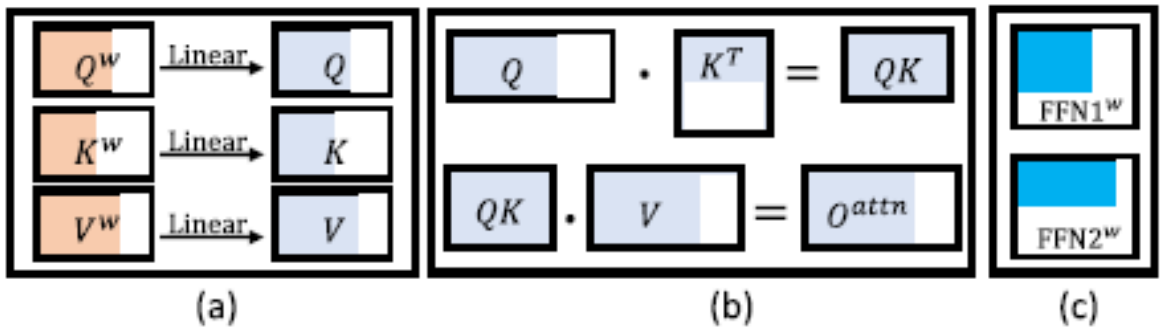
Pruning method for attention

- TransformerZip (2019)
  - Magnitude-based pruning
  - To reduce weight size
- HAT(Hardware Aware Transformers) (2020)
  - Crops the weight in both dimension
  - To reduce weight size and form regularly shaped weights
- FTRANS (2020)
  - Utilize block-circulant matrix to replace selected weights
  - Reduce model memory footprint

Compression impact

- 기존 compression 방법론은 weight의 large memory footprint size를 효율적으로 줄일 수 있음

- 하지만 weight compression 과 computation이 함께 고려되지 않았기 때문에 on-chip 자원의 효율성이 떨어짐

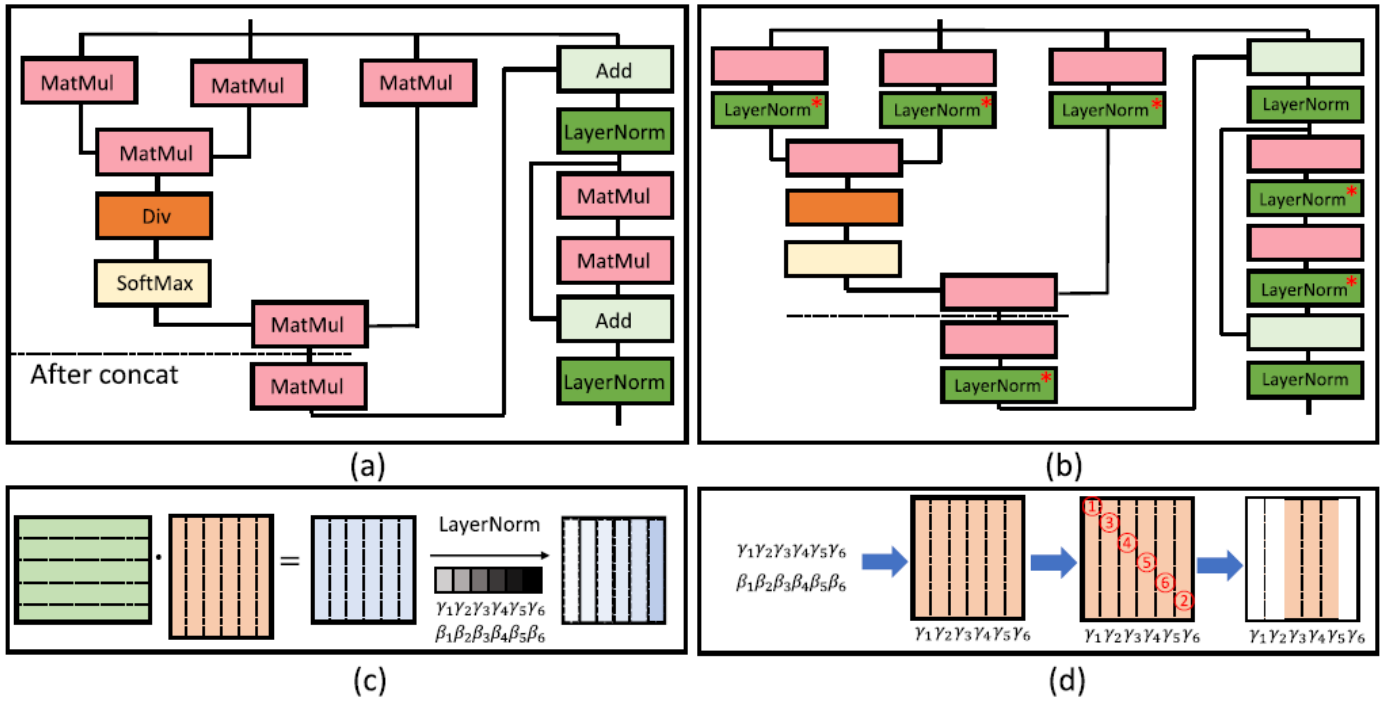


제안사항

- 버퍼 할당에 대해 모델 압축의 영향을 고려
- Attention model이 압축되는 동안 커널 계산을 요구
- 새로운 구조적 pruning method 제안 (with memory footprint awareness)

Compression analysis

- Weight가 있는 각 층에 대해 LayerNorm을 추가
- Row-wise scaling 진행
- γ의 값 = Compression Significant factor
- γ의 값이 작은 row에 대해 pruning



$$x_{scaled} = \frac{x - E[x]}{\sqrt{Var[x] + \epsilon}} * \gamma + \beta$$

(a) Encoder (b) LayerNorm insertion of encoder
(c) LayerNorm scaling factor γ and β
(d) Weight significance based on γ

Memory footprint aware pruning

- Column-wise pruning with memory footprint awareness
- 앞선 row-wise pruning의 γ 값을 기준으로 최소 ratio를 제한한다.
- 해당 값을 키우면서 pruning을 진행하고 model accuracy에 영향을 준다면 다시 줄인다.

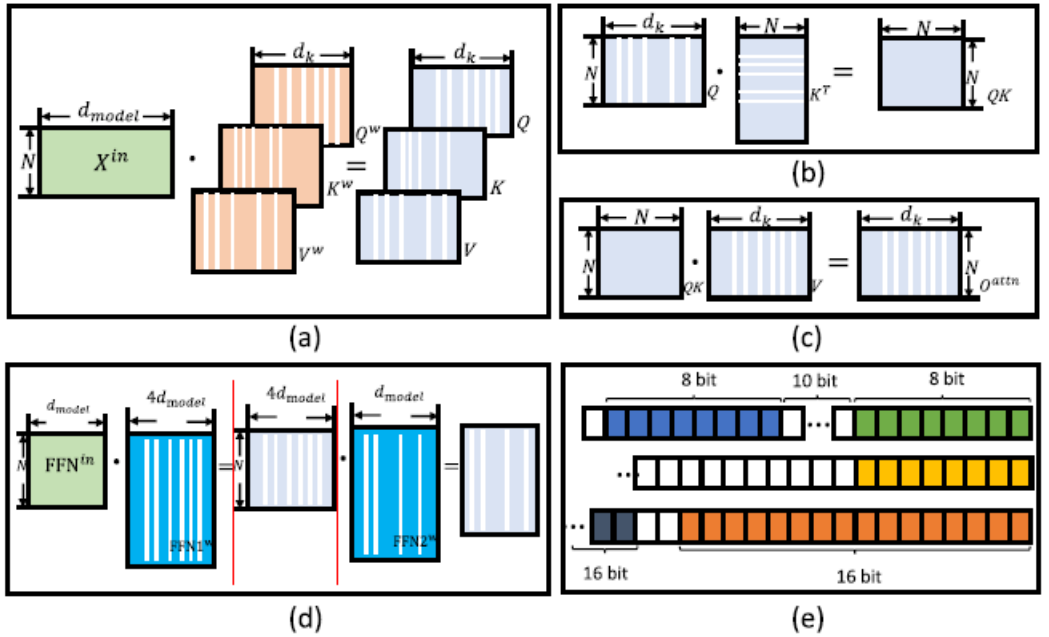


Fig. 6. (a) Sparse pattern of weight Q, K, and V in the proposed technique. (b) Sparse pattern of $Q * K^T$. (c) Sparse pattern of $QK * V$. (d) Sparse pattern in FFN operations. (e) INT8 multiplication encoding overview.

Challenges in accelerating sparse Transformer

- Multi-size multiply-accumulate
  - MAC 연산의 크기가 다르기 때문에 kernel이 이에 맞게 설계되지 않았다면 전체적인 성능 하락을 초래함
- Inefficient INT8 computation
  - 현대 FPGA의 DSP는 높은 대역폭을 지원하는 데, 이는 INT8 연산을 진행하기에 너무 과하다.
- Multiplying with zero
  - 열에 대해 Query와 Key값은 sparse한 특성을 가지기 때문에 0의 곱셈은 컴퓨팅 자원에 대해 심각한 성능 저하를 일으킴
- Compressed matrix restoration
  - Memory footprint는 unpruned weight를 유지하고 있을 때만 줄일 수 있다. Pruned weight를 복구 할 수 있어야 한다는 의미를 가짐

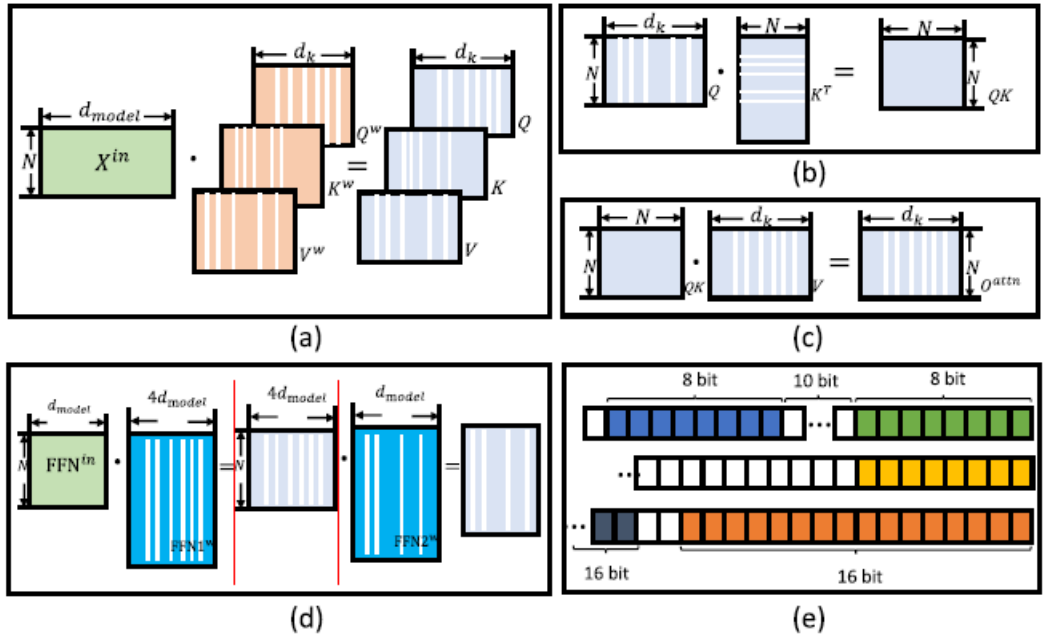


Fig. 6. (a) Sparse pattern of weight Q, K, and V in the proposed technique. (b) Sparse pattern of $Q * K^T$. (c) Sparse pattern of $QK * V$. (d) Sparse pattern in FFN operations. (e) INT8 multiplication encoding overview.

Challenges in accelerating sparse Transformer

- 각 행렬 IN과 WEI의 곱으로 OUT을 표현
- Loop Unrolling을 통해 OUT의 열을 병렬적으로 계산

- 동일한 단일 사이즈 연산으로 바꿨기 때문에 Multi-size multiply-accumulate, Inefficient INT8 computation 문제를 해결함
- 위 방식을 활용하면 Multiplying with zero, Compressed matrix restoration 문제도 자연스럽게 해결됨

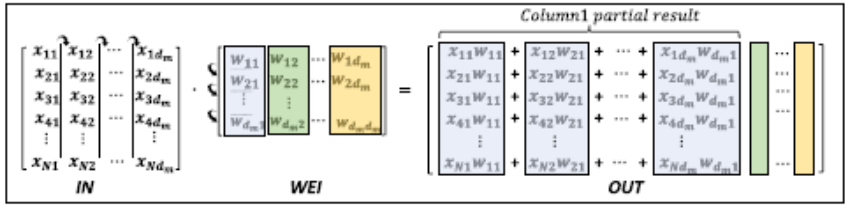


Fig. 7. Unified computing pattern in element-wise multiplication and addition.

```
for (col_wei=0; col_wei< C_WEI; col_wei++) {
  for (row_wei=0; row_wei< R_WEI; row_wei++) {
    for (row_in=0; row_in< R_IN; row_in++)      {
      OUT[row_in][col_in] +=IN[row_in][row_wei]*WEI[row_wei][col_wei]
} } }
/* R represents the number of rows */
/* C represents the number of columns */
```

Fig. 8. The pseudocode of the computing pattern optimization in a manner of loop iteration.

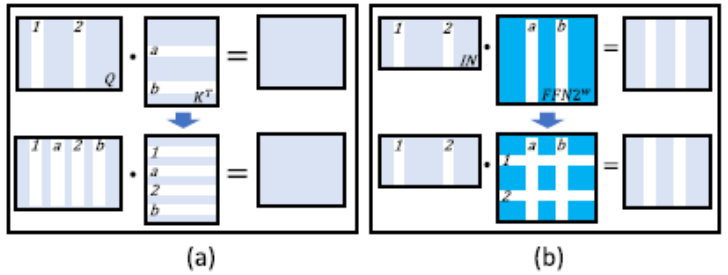


Fig. 9. Multiplicand and multiplier data alignment in (a) sparse $Q * K^T$ (b) sparse linear layer.

ViA: A Novel Vision-Transformer Accelerator Based on FPGA

IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2022

차세대 컴퓨터 시스템 연구실

이원호

문제점 및 동기

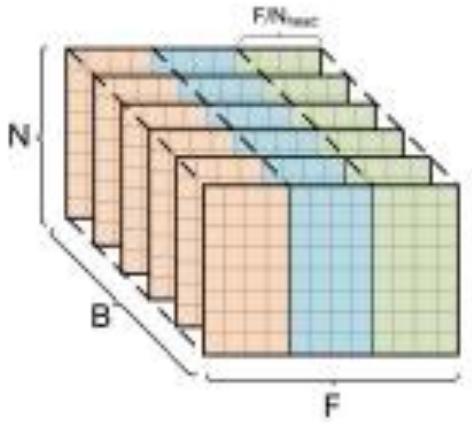
- Transformer의 자연어 처리 영역 뿐 아니라 컴퓨터 비전의 영역에서도 좋은 성능을 보여줬지만, 그에 따른 많은 계산양과 메모리를 요구함
- 컴퓨터 비전에서 트랜스포머의 도입 (ViT 모델)이 지속적으로 논의되고 있음

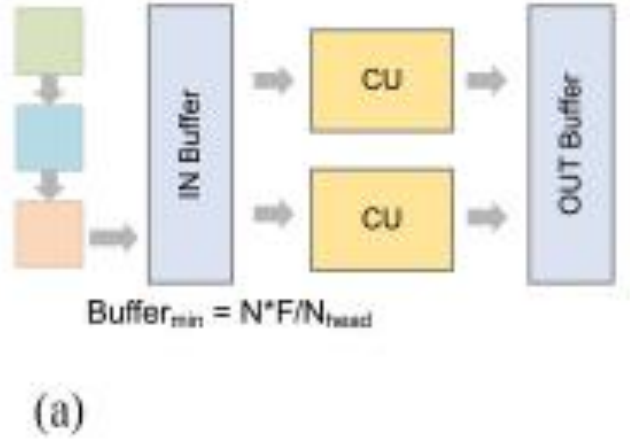
- 기존 FPGA에서 Transformer에 대한 논의는 NLP에 한정됨
- CV와 NLP에서 사용하는 데이터 형태와 특징 그리고, 모델의 구조가 다름
  - 데이터의 차이로 인한 locality 문제 발생
  - 모델 구조의 차이로 인한 path dependence 문제 발생

Data Difference

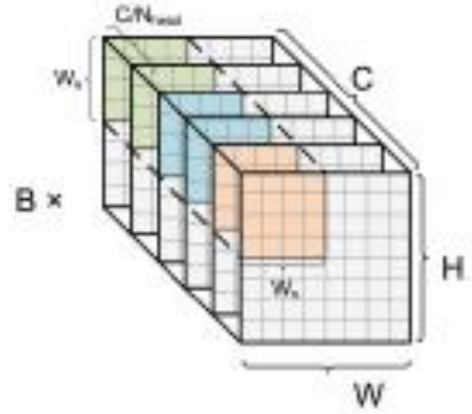
- CV에서의 활용에서 이미지 데이터는 patch size로 나눌 필요가 있음

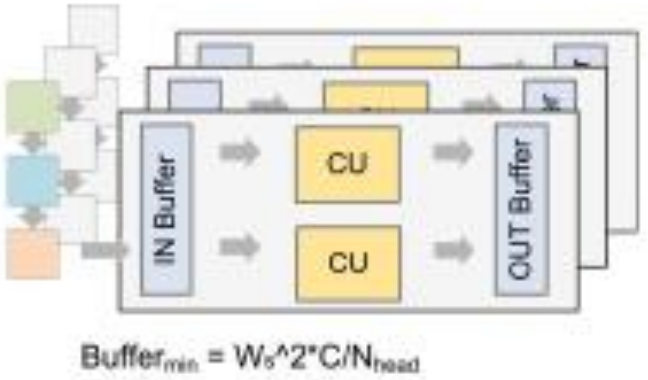
- 사용하는 데이터의 주요 차원에 차이가 있음
  - NLP <B, N, F>
    (Batch size, Sentence length, Future size after embedding)
  - CV <B, H, W, C>
    (Batch size, Height, Width, Channel size)





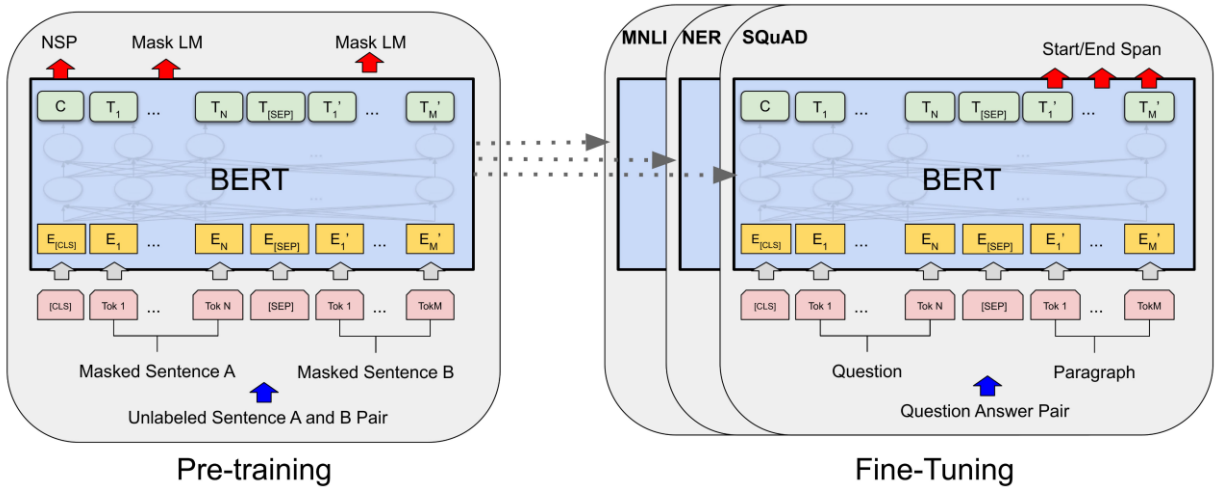
(a)

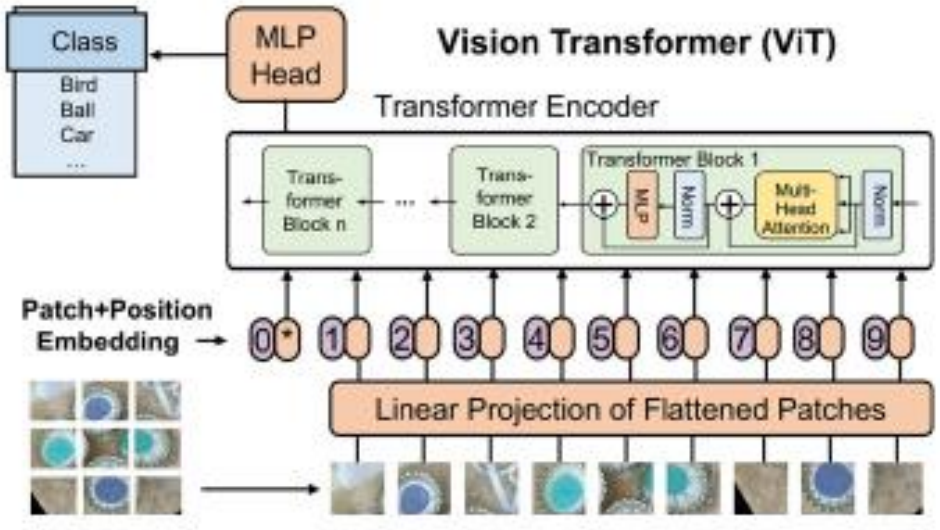




(b)

Model Difference





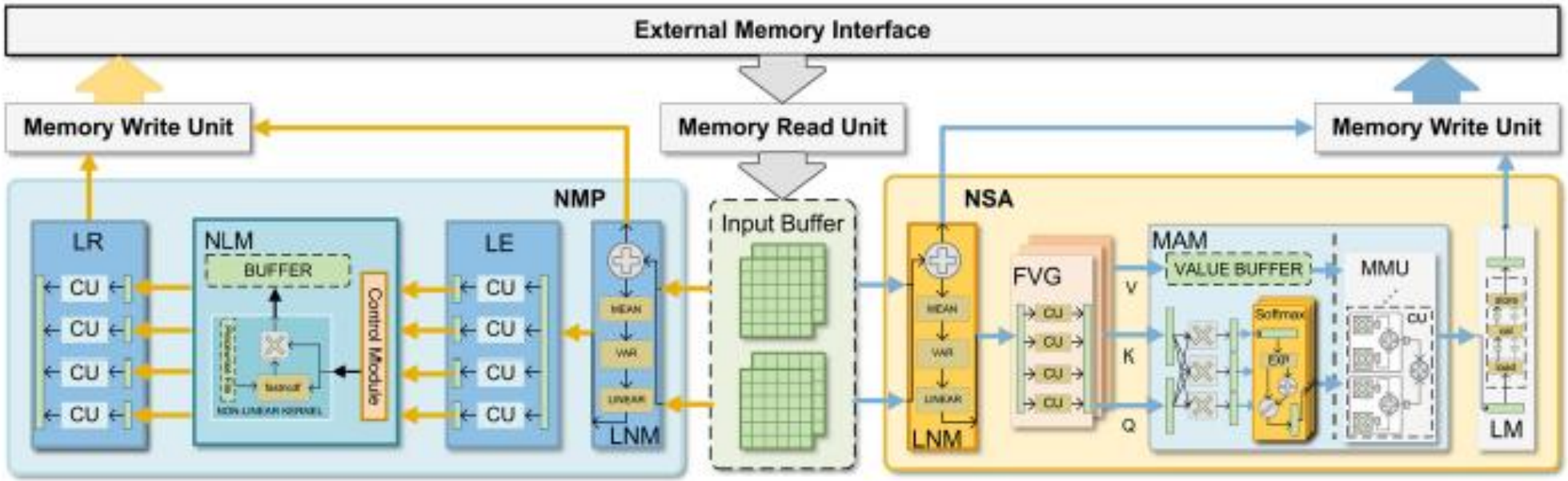
제안사항

- ViT 모델에 대한 분석을 기반으로 적절한 partition strategy 설계
  - 이미지에서 data locality의 영향을 줄임
  - 연산과 메모리 접근의 효율성 향상

- Half-layer mapping / throughput analysis
  - Path dependence의 영향을 줄임

- 최적화 전략을 기반으로 internal stream과 함께 두 가지 reuse processing engine 설계

Overall Architecture



Coordinated Batching and DVFS for DNN Inference on GPU Accelerators

IEEE Transactions on Parallel and Distributed Systems, 2022

차세대 컴퓨터 시스템 연구실

이원호

문제점 및 동기

- 인공지능 가속을 위한 가속기는 내외부적인 요인으로 인해 성능 저하가 발생함
  - 해당 논문에서는 특히 power cap에 집중함
- 일반적으로 power cap을 해결하는 방법은 DVFS (Dynamic Voltage Frequency Scaling)의 활용
- Batch size는 실행시간동안 항상 동일하게 유지되는 문제점
  - 이를 변경하기 위해서는 실행을 멈추고 새로운 DNN을 실행해야 함
- 일반적인 DVFS는 platform에 의존적인 문제가 존재함

제안사항

- 가속기에 입력의 배치 사이즈를 추가로 전달 -> 배치 사이즈가 전력 소비와 DNN 추론 성능에 미치는 영향 평가
- Fast & lightweight runtime system (BatchDVFS)의 설계와 구현
- 동적인 배치 작업에서 throughput과 power consumption 사이의 trade-off 관계에 대해 배치 사이즈를 조절
- 적절한 배치 사이즈에 대한 선택 시간을 줄이기 위해 이진 검색을 기반 알고리즘 설계
- 더 넓은 범위에서 전력 소비를 제어 가능
- Long running job에 대해 준최적 전략을 찾는 과정에서 overhead를 줄이기 위해 BOBD 제안
  - Batch size와 DVFS 결과의 조합으로 더 넓은 상태 공간을 탐색하기 위해 Bayesian Optimization 활용

Summary

- Design space에 대한 탐색 영역이 넓어짐에 따라 Heuristic 탐색의 불가로 강화학습 적용
- 고정적인 메모리 사이즈와 이를 해결하기 위한 압축의 적용 및 압축이 모델에 미치는 영향 기술
- 트랜스포머의 적용 범위가 CV로 확장됨에 따라 그에 맞는 데이터 구조 및 모델 구조의 차이와 그에 따른 새로운 구조의 적용
- Batch size가 추론에 미치는 영향을 기술하고 그에 따라 동적인 batch size의 선택에 대한 논의

Reference

- Algorithm-hardware Co-design of Attention Mechanism on FPGA Devices, ACM Transactions on Embedded Computing Systems (TECS), 2021
- ViA: A Novel Vision-Transformer Accelerator Based on FPGA, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2022
- ConfuciuX: Autonomous Hardware Resource Assignment for DNN Accelerators using Reinforcement Learning, 2020 53rd Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), 2020
- Coordinated Batching and DVFS for DNN Inference on GPU Accelerators, IEEE Transactions on Parallel and Distributed Systems, 2022
- Papers with code (BERT)
- Tensorflow Blog