이기종 컴퓨팅과 복수 신경망 추론 환경에서 높은 처리량을 위한 스케줄러 관한 연구

차주형, 박제만, 권용인*

*한국전자통신연구원

aoikazto@naver.com, jeman@etri.re.kr, *yongin.kwon@etri.re.kr

A Study on Scheduler for High Throughput in Heterogeneous Computing and Multiple Deep Learning Models

Joo Hyoung Cha, Jeman Park, Yongin Kwon*

*Electronics and Telecommunications Research Institute.

요 약

연산 장치마다 신경망 모델을 동시 추론이 불가하거나 특화된 연산이 상이하다. 따라서 작업 순서를 재배열하거나 연산 장치마다 선호되는 신경망 모델 지정을 통해 성능 향상할 수 있다. 이기종 컴퓨팅 환경과 다중 신경망 환경에서 높은 처리량을 제공하기 위해 연산 장치별 선호하는 신경망 모델을 파악하고 재배열하는 스케줄러 도입이 필요하다. 본 논문에서는 신경망 모델 추론 작업을 분배하는 스케줄러와 성능 분석하는 프로파일러를 구현한 뒤 스케줄러별 성능 변화를 비교한다. 실험 결과 스케줄러에 따라 최대 62.72% 성능 차이가 발생하였다.

Ⅰ. 서 론

최근 이미지나 영상을 인식하거나 의미론적 정보를 얻기 위해 딥 러닝 기술이 많이 사용되고 있다. 또한 하나의 정보를 추출하기 위해 복수 개의 신경망 모델을 사용되고 있다. 이로 인해 딥 러닝 연산량이 점차 증가하게 되면서 단일 연산 장치에서 수행 시 높은 부하가 발생하였다.

이를 위해 이종의 컴퓨팅 환경에서 복수 개의 신경망 연산을 수행하기 위한 연구가 활발하게 진행되고 있다. 기존에는 이종의 NPU에서 복수 개의 신경망 모델을 자원 이용률 기준으로 작업 분배하였다[1]. 그러나 이는 각 연산 장치의 특성을 이용하지 않아 이기종 컴퓨팅 상황에 적합한 스케줄러라 보기 어렵다. 또한 다양한 스케줄링 알고리즘을 적용한 여러 종류의 연산 장치를 함께 연산에 참여하여 비교 분석한 연구가 없었다.

본 논문에서는 스케줄러에 따른 성능 변화를 분석하는 것을 목표로, 이기종 컴퓨팅 환경과 다중 신경망 추론 환경에서 높은 처리량을 제공하기 위해 다양한 스케줄링 알고리즘의 성능 변화를 비교한다. 또한 특정 연산 장치에 최적화된 연산 수행하도록 작업을 재배열했을 때 성능을 확인한다.

Ⅱ. 시스템 환경 및 구성

M 개의 연산 장치와 N 개의 신경망 모델을 고려하여 스케줄링 수행하기 위해 스케줄러 구현이 필요하다. 이를 위해 그림 1과 같이 응용 계층, 스케줄링 계층, 하드웨어 계층으로 3개의 모듈로 추상화하여 구성하였다.

먼저 응용 계층에서 신경망 모델을 추상화하여 스케줄러의 대기열에 작업 등록함으로써 스케줄링 알고리즘이 시작된다. 스케줄러에는 프로파일러와 작업 재배열하는 스케줄러, 결과 값을 반환하는 콜백이 존재한다.

프로파일러는 스케줄러에서 신경망 연산을 수행한 뒤 추론 속도를 분석하여 연산 효율성 및 선호도를 계산한다. 이를 통해 스케줄러는 실시간으로 요청되는 환경에서 연산 선호도를 기준으로 작업 순서를 재배열한다.

스케줄러는 프로파일러를 통해 결정된 연산 선호도의 값에 따라 할당된 신경망 연산을 하드웨어 계층에서 수행하도록 스케줄링한다. 수행한 결과 값을 콜백에 전달하여 응용 계층으로 전달하는 구조이다.

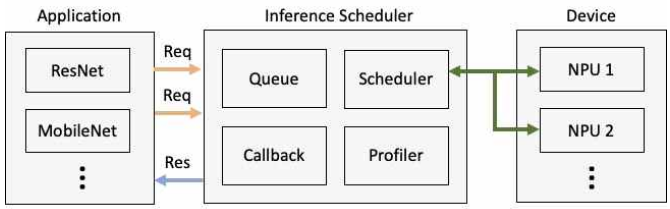

그림 1. 실시간으로 최적의 추론 처리량을 탐색하는 스케줄러 구조

Ⅲ. 실험 환경

1. 연산 장치 구성

복수 개의 신경망 모델과 이기종 컴퓨팅 환경을 구성 후, 스케줄러의 성능을 평가하기 위해 동일한 제품이 아닌 서로 제조사가 다른 제품으로 하였다. 실험에 사용된 임베디드 보드는 Asus사의 Tinker Edge R를 사용하였으며, NPU 2개를 추가로 탑재하여 환경 구성하였다.

표 1. 연산 장치별 지원하는 API와 선호되는 데이터 타입 형식

프로세싱 유닛 명	백엔드	데이터 타입	복수 추론
CPU (A72)	ArmCL	FP 32	O
Intel NCS 2	ncAPI	FP 16	X
Google Coral TPU USB	edgetpu	INT 8	O
Rockchip NPU (RK1808)	rknpu	INT 8	O

표 1에 Intel NCS 2는 대부분의 연산 장치와 달리 복수 개의 신경망 모

델을 내부 메모리에 적재하여 추론 연산을 수행할 수 없는 문제가 있다. 이를 통해 기존 알고리즘인 자원 이용률을 기준으로 스케줄링하는 것이 아닌 복합적인 스케줄링 알고리즘이 필요하다는 것을 알 수 있다.

2. 스케줄러 구성

스케줄러는 응용 계층으로부터 입력되는 작업 순서를 저장하고 재배열한다. 본 실험에서 사용한 스케줄러는 아래의 4가지 종류이다.

FCFS 스케줄러는 선입 선출 알고리즘으로 추론 대기열에서 순차적으로 유휴 상태인 연산 장치에 추론하도록 하는 알고리즘이다. 이를 통해 모든 연산 장치는 항상 연산에 참여하도록 하는 스케줄링 알고리즘이다.

Affinity 스케줄러는 특정 신경망 모델 연산 수행을 하나의 연산 장치에서 수행하도록 1 대 1 사상하는 방식이다. 사상하는 방식은 모델마다 최소 비용을 가지는 연산 장치를 구한 뒤 비용이 가장 높은 연산 장치 순으로 사상한다.

Short Job First(SJF) 스케줄러는 입력되는 신경망 모델에서 연산 장치 간의 평균 시간 연산 시간 정보를 활용하여 연산 시간을 예측한다. 해당 정보를 활용하여 작업이 가장 빠르게 종료가 되는 연산부터 우선적으로 수행한다.

Priority 스케줄러는 신경망 모델별 대기열을 생성한 뒤, 신경망 모델의 파라미터와 연산 장치별 추론 속도의 비율을 계산하여 효율이 높은 신경망 모델을 우선하여 추론한다. 우선순위는 임의로 지정하거나 프로파일러를 통해 결정될 수 있다.

스케줄러의 작업 분배 알고리즘에 따라 프로파일러가 필요한 경우와 아닌 경우가 존재한다. 이는 Affinity와 SJF, Priority(우선순위) 스케줄러에서 우선순위를 계산하기 위해 이전에 연산을 수행했던 정보와 초기 정보가 필요로 하기 때문이다. 하지만 FCFS(First Come First Served) 스케줄러의 경우 필요로 하지 않는다.

Ⅳ. 실험 결과

실험에 사용된 신경망 모델은 Tensorflow의 Model Zoo[4]에서 MobileNetV2[2], ResNet50[3], ResNet101[3]를 다운로드한 뒤 연산 장치에 적합한 모델로 변환하여 사용하였다. 실험에 사용되는 영상의 크기는 1(N), 224(H), 224(W), 3(C)이며 데이터 구조는 NHWC로 통일하였다.

제시한 스케줄링 기법 중 Affinity와 SJF, Priority 스케줄러는 신경망 모델별 우선순위 계산이 필요하다. 이를 위해 평균 추론 시간과 신경망 모델의 파라미터의 수는 표 2와 같다.

표 2. 신경망 모델과 연산 장치 간 평균 추론 시간 및 우선순위 정보

	MobileNetV2	ResNet50	ResNet101
신경망 파라미터의 수	3,538,984	25,636,712	44,707,176
CPU (A72)	86.355 ms (1)	582.072 ms (3)	1201.923 ms (2)
Intel NCS 2	27.459 ms	50.743 ms	94.509 ms
Google Coral USB	3.393 ms (1)	54.086 ms (3)	108.260 ms (2)
Rockchip NPU	10.499 ms (2)	21.345 ms (1)	31.770 ms (3)

평균 추론 시간을 기준으로 각 스케줄러의 우선순위를 생성한 결과 Affinity 스케줄러는 1 대 1 사상하는 방식이므로 단일 모델 추론만 수행하게 된다. 이로 인해 Rockchip NPU가 ResNet101, Intel NCS 2는 ResNet 50, Google Coral은 MobileNetV2를 담당하여 연산을 수행하게 된다. SJF의 경우 연산 수행 시간이 짧은 모델부터 우선하여 수행한다. 따라서 입력된 대기열에서 작업 순서를 항상 MobileNetV2, ResNet50, ResNet101 순으로 정렬한 뒤 순차적으로 연산한다. Priority는 표 2의 괄호 안에 있는 숫자를 기준으로 우선순위를 매겨 스케줄링을 수행한다. 연산 장치마다 지정된 우선순위는 임의로 지정하였다.

그림 2는 스케줄링 알고리즘과 Intel NCS 2의 연산을 수행할 신경망 모델에 따른 성능 비교 분석을 수행하였다. 신경망 모델별 1,000회, 총 3,000회를 수행하여 성능을 측정하였다. 또한 시행마다 10회 반복하여 평균 소요 시간을 계산하였다.

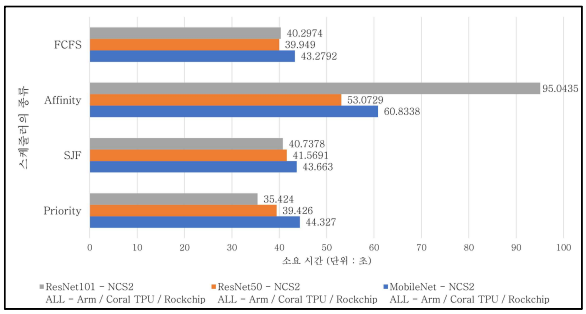

그림 2. 스케줄러 별 신경망 모델 추론 처리량 성능 비교

복수 개의 신경망 모델을 내부 메모리에 적재하지 못하는 Intel NCS 2에 적재할 모델에 따라 성능이 변화되는 것을 알 수 있다. 따라서 NCS 2에 적재한 모델과 스케줄러를 변화하면서 소요된 시간을 측정하였다.

추론 처리량은 Intel NCS 2가 ResNet101을 연산하였을 때 추론 처리량이 추세적으로 향상 되는 것을 알 수 있다. 또한 스케줄러는 Priority, FCFS, SJF, Affinity 순으로 높았으며 성능 차이는 최대 62.72%의 차이가 발생하였다.

Ⅴ. 결론

본 논문에서 이기종 컴퓨팅 환경에서 복수 개의 모델 추론할 때 연산 장치마다 상이한 처리 속도에 맞춰 작업 분배를 한다면 처리량이 향상되는 것을 알 수 있다. 향후 연구로 이기종 컴퓨팅 환경에서 복합적인 스케줄링 알고리즘과 우선순위를 계산하는 알고리즘을 통해 성능을 개선하고자 한다.

ACKNOWLEDGMENT

이 논문은 2023년도 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] Sang Cheol Kim, et al. "Development of NPU Operating System Platform for Inference of Multiple Neural Networks on Multiple Heterogeneous NPU Devices." KIISE Transactions on Computing Practices 26.12 (2020): 561-566.

[2] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[3] Sandler, Mark, et al. "Mobilenetv2: Inverted residuals and linear bottlenecks." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[4] Hongkun Yu, et al. "TensorFlow Model Garden." . (2020).