실시간 추론 시스템을 위한 Deadline-Aware TOPSIS 기반 스케줄링 기법

박정애, 윤수연, 김혁만*
국민대학교, 국민대학교, *국민대학교
barkjungae@kookmin.ac.kr, 1104py@kookmkin.ac.kr, *hmkim@kookmin.ac.kr

A Deadline-Aware TOPSIS-Based Scheduling for Real-Time Inference Systems

Park Jungae, Yoon SooYeon, Kim Hyeokman*
Kookmin Univ, Kookmin Univ, *Kookmin Univ

요 약

엣지 환경에서의 딥러닝 추론 서비스는 제한된 연산 자원과 다양한 워크로드 특성으로 인해 효율적인 작업 스케줄링이 필수적이다. 그러나 기존의 단일 기준 기반 스케줄링 기법은 모델 이질성, 부하변화, 자원병목과 같은 현실적인 조건에서 안정적인 성능을 보장하지 못한다. 본 논문은 이러한 한계를 단계적으로 분석하고 다중 QoS 기준과 deadline 민감도를 통합한 Deadline-Aware TOPSIS(DA-TOPSIS) 스케줄링 기법을 제안한다. 제안 기법은 지연시간, deadline 만족도, 시스템 부하를 종합적으로 고려하여 작업을 동적으로 분배한다. 시뮬레이션 기반 실험 결과, 도착 간격이 0.08초 이하인 고부하 환경에서 RULE 및 EDF 스케줄러의 deadline miss ratio가 최대 0.48까지 급격히 증가한 반면, DA-TOPSIS는 동일 조건에서 miss ratio를 0.08~0.15 수준으로 유지하며 기존 단일 기준 및 TOPSIS 기반 기법 대비 가장 안정적인 deadline 만족 성능을 보였다.

Ⅰ. 서론

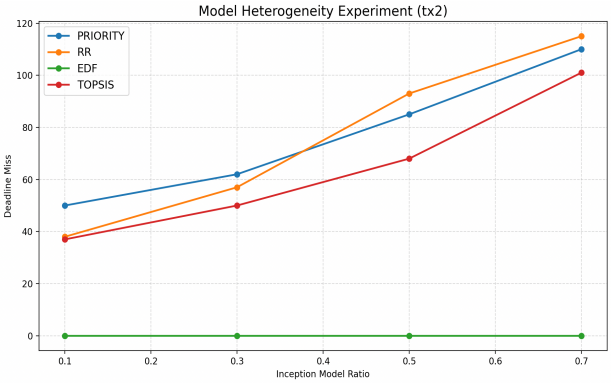
최근 엣지 컴퓨팅 환경에서는 제한된 연산 자원과 이질적인 딥러닝 워크로드로 인해 실시간 추론 작업의 효율적인 스케줄링이 중요해지고 있다. Jetson Xavier 및 TX2와 같은 엣지 환경에서는 연산 복잡도가 상이한 모델들이 동시에 실행되며, 이는 시스템 지연과 QoS에 직접적인 영향을 미친다[2],[5]. 기존 스케줄링 기법들은 deadline이나 처리 시간과 같은 단일 기준에 기반하여, 모델 복잡도 변화나 자원 병목 상황에서 성능 저하를 보인다[3]. EDF는 deadline 만족 측면서는 효과적이지만 GPU 부하나 작업 중요도를 반영하지 못하는 한계가 보고되고 있다[4],[5]. 다중 기준 의사결정 기법인 TOPSIS는 복수의 성능 지표를 고려할 수 있으나, 기존 적용에서는 deadline 민간도가 충분히 방영되지 않았다[1]. 본 연구는 이러한 한계를 실험적으로 분석하고, deadline 민감도를 통합한 Deadline-Aware TOPSIS(DA-TOPSIS) 스케줄링 기법을 제안한다.

Ⅱ. 본론

엣지 환경 기반 DNN 추론 서비스는 단일 모델과 QoS 기준한 단순한 처리 구조를 넘어, 서로 다른 연산 복잡도의 모델들이 혼재된 워크로드 환경으로 빠르게 전환되고 있다. Jetson Xavier와 TX2와 같은 엣지 환경에서는 제한된 연산 자원과 메모리 대역폭으로 인해 deadline, latency, GPU load, QoS 등 다양한 요소가 동시에 스케줄링 성능에 영향을 미친다. 그러나 기존 스케줄링 기법들은 대부분 단일 기준에 기반하고 있어 이러한 복합적 특성을 충분히 반영하지 못하는 구조적 한계를 가진다. 본 연구는 이러한 한계를 실험적으로 분석하고, 다중 기준 기반 스케줄링의 필요성을 검증하기 위한 실험을 설계하였다.

2.1 Heterogeneous Model Mix 환경의 특징

첫 번째 실험에서는 MobileNet, Inception 모델과 같이 연산 복잡도와 GPU 사용량이 상이한 모델들의 혼재 비율을 점진적으로 증가시키며 스케줄러의 안정성을 평가하였다.

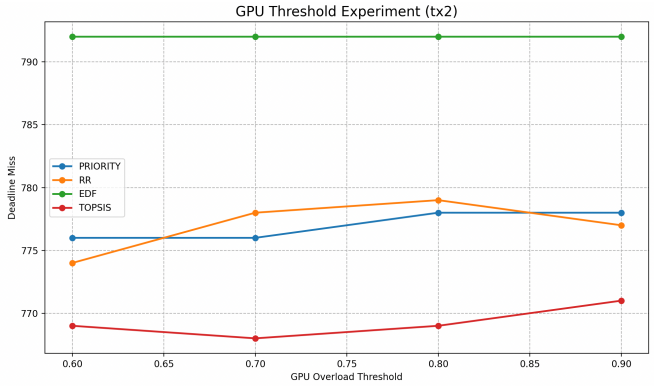


[그림 1. Heavy Model ratio vs Deadline Miss]

그림1과 같이 자원이 더 제한적인 TX2 장비를 기준으로 PRIORITY 및, Round-Robin 스케줄러는 heavy 모델 비중이 증가함에 따라 deadline miss가 급격히 증가하며 스케줄링 구조가 쉽게 붕괴되는 현상을 보였다. 이는 두 방식 모두 모델의 연산량이나 GPU 소비 특성을 고려하지 못한 채 단일 기준으로 작업을 정렬하기 때문이다. EDF는 deadline 기준에 의해 상대적으로 낮은 miss를 유지하였으나, QoS나 GPU load를 반영하지 못해 QoS 측면의 한계가 명확히 드러났다. 반면, TOPSIS 기반 스케줄링은 heavy 모델 증가 상황에서도 miss가 완만하게 증가하며 가장 안정적인 성능을 보였다. 이는 heterogeneous workload 환경에서 단일 기준 스케줄링이 근본적으로 취약함을 보여주며, 다중 기준 고려의 필요성을 시사한다.

2.2 자원 부족 상황에서의 GPU Overload 영향

두 번째 실험에서는 GPU 사용량이 특정 임계값(threshold)을 초과할 경우 추가 latecny penalty가 발생하도록 설정하여 자원 부족 상황을 모델링하였다.

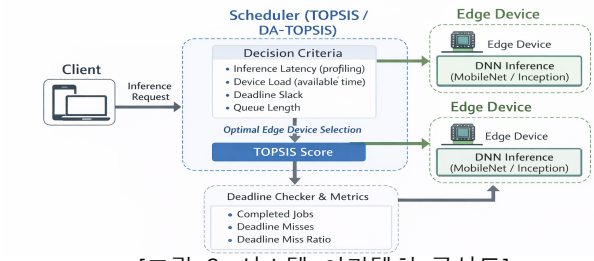


[그림 2. GPU Threshold vs Deadline Miss]

그림 2와 같이 threshold가 감소할수록 PRIORITY와 RR은 GPU 자원 부족을 고려하지 못해 deadline miss가 불규칙적으로 급증하였다. EDF는 deadline 기준 정렬 특성상 threshold 변화에 큰 영향을 받지 않았지만, 이는 GPU overload를 회피한 결과가 아니라 자원 상태를 무시한 결과 한계를 가진다. 반면 TOPSIS를 GPU load를 cost 항목으로 반영함으로써 threshold 감소 상황에서도 가장 완만한 성능 저하를 보였으며, 특히 TX2와 같이 자원이 제한된 환경에서 월등히 안정적인 성능을 유지하였다. 이는 자원 제약적인 엣지 환경에서 resource-aware scheduling이 필수적임을 보여준다. 이러한 결과는 엣지 DNN 추론 환경에서 deadline 하나만을 기준으로 한 스케줄링은 더 이상 충분하지 않으며, 다양한 QoS 요소를 동시에 고려할 수 있는 multi-criteria decision 기반 스케줄링 시스템이 필수적임을 의미하고 있다.

Ⅲ. Deadline-Aware TOPSIS-based 스케줄링 시스템

본 연구에서는 이기종 엣지 디바잇 환경에서 실시간 DNN 추론 작업을 효율적으로 스케줄링하기 위해, Deadline-Aware TOPSIS(DA-TOPSIS) 기반 스케줄링 시스템을 제안한다. YAFS(Yet Another Fog Simulator)를 기반으로 구현되었으며, 실시간으로 유입되는 DNN 추론 작업을 Xavier 및 TX2와 같은 서로 다른 성능의 엣지 디바이스에 동적으로 할당한다.

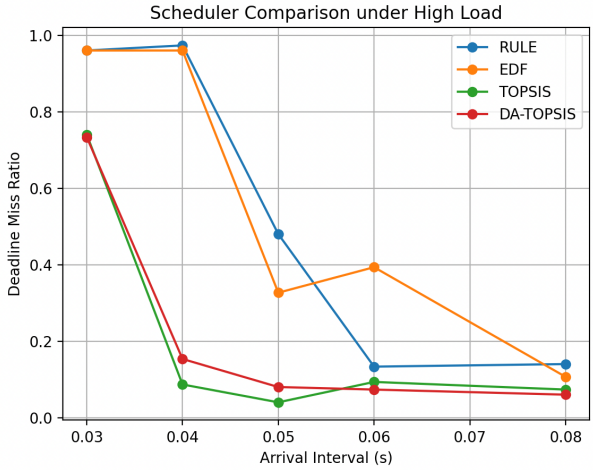


[그림 3. 시스템 아키텍처 구성도]

그림3은 제안하는 시스템의 전체 아키텍처를 나타낸다. 시스템은 크게 Job Generator, Scheduler, Edge Devices의 세 부분으로 구성된다. Job Generator는 일정한 주기로 추론 작업을 생성하며 각 작업은 모델종류, batch size, arrival time, deadline 정보를 포함한다. Scheduler는 대기큐에 존재하는 작업을을 대상으로 스케줄링 알고리즘을 수행하며, 각 작업을 어느 디바이스에 할당할지 결정한다. Edge Devices는 할당된 작업을 실제 추론 시간 모델에 따라 처리하며, 완료 시간과 deadline 만족 여부를 기록한다.

Ⅳ. 실험

제안하는 시스템의 성능을 검증하기 위해, 동일한 환경에서 RULE-based Scheduler, EDF, TOPSIS, DA-TOPSIS 네 가지 스케줄링 기법을 비교하였다. 실험 환경은 Xavier와 TX2로 구성된 이기종 엣지 디바이스에 DNN 추론 작업을 YAST 시뮬레이터를 통해 요청하고 장비에 할당하도록 하였고, batch size에 따라 실험을 진행하여 모든 작업의 deadline은 300ms로 동일하게 부여하였다. 평가지표로는 deadline 내 완료된 작업 수를 기준으로 전체 작업 대비 deadline 초과 비율을 확인하여 요청 도착 간격 (arrival interval)을 점진적으로 감소시켜 시스템 부하를 증가시는 시나리오를 구성하였다. Arrival interval은 0.30초에서 0.03초까지 감소하며, 저부하 환경에서 극단적인 고부하 환경까지 평가하였다.



[그림 4. Arrival Interval 변화 및 Deadline Miss Ratio]

그림4와 같이 도착 간격이 0.15초 이상인 구간에서는 모든 스케줄링 기법이 deadline miss 없이 정상 동작하였다. 그러나 0.08초 이하의 고부하 영역부터 스케줄러 간 성능 차이가 명확히 나타났다. RULE 및 EDF 스케줄러는 deadline miss가 급격히 증가하며 성능이 붕괴되는 양상을 보였다. TOPSIS는 상대적으로 안정적인 성능을 유지했으나, 극단적인 고부하 환경에서는 miss ratio가 증가하였다. 반면 DA-TOPSIS는 중,고부하 환경에서도 실시간 제약과 자원 활용 간의 균형을 가장 안정적으로 유지하였다.

Ⅴ. 결과

본 실험은 단일 기준 스케줄링이 이기종 엣지 환경과 고부하 조건에서 구조적 한계를 실험적으로 확인하였다. 이를 해결하기 위해 본 논문은 deadline 민감도를 통합한 DA-TOPSIS 스케줄링 기법을 제안하였으며, 다양한 워크로드 조건에서 deadline miss를 효과적으로 감소시켰다. 실험 결과는 DA-TOPSIS가 자원 활용과 실시간 제약 간의 균형을 안정적으로 유지하는 실용적인 엣지 추론 스케줄링 기법임을 보여준다.

ACKNOWLEDGMENT

본 연구는 2022년 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학사업의 연구결과로 수행되었음 (2022-0-00964)

참 고 문 헌

[1] Al-Masri, E., et al. “A Multi-Criteria Decision Making Framework for Ranking IoT Services Using TOPSIS,” *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5423-5435, June 2019.

[2] Chen, J., Ran, X., and Glenn, J. “Deep Learning with Edge Computing: A Resource-Aware Scheduling Study on Jetson TX2,” *Proceedings of the IEEE International Conference on Distributed Computing Systems Workshops (ICDCS Workshops)*, July 2019.

[3] Chung, S., Kim, H., and Yoo, Y. “Predictive Model for Edge AI Inference Latency on Embedded GPUs,” *IEEE Access*, vol. 9, pp.

[4] Kang, D., Emmons, J., Abuzaid, F., and Bailis, P. “Multi-Model Serving System for GPU Sharing and Isolation,” *Proceedings of the Conference on Machine Learning and Systems (MLSys)*, March 2021.

[5] Zhang, C., et al. “DNN Scheduling for Multi-Tenant Edge Inference,” *Proceedings of the ACM/IEEE Symposium on Edge Computing (SEC)*, October 2020.