LLM 하이브리드 정책 오케스트레이션 기반 에너지 최적화 구현 및 검증

신주희, 최재형, 유현민, 홍인기

경희대학교

{odong3094, wogud1221, yhm1620, ekhong}@khu.ac.kr

Implementation and Validation of Energy Optimization Using LLM-based Hybrid Policy Orchestration

Juhee Shin, Jaehyung Choi, Hyunmin Yoo, Eenkee Hong
Kyunghee University

요 약

본 연구는 O-RAN Service Management and Orchestration(SMO) 환경에서 AI/ML Framework(AIMLFW)와 O1 interface를 통합한 대규모 언어모델(LLM) 기반 에너지 최적화 시스템을 구현한다. Kubernetes 환경에서 gNB 시뮬레이터, 다층 이상 탐지, O1 actuator, LLM 정책 서버를 통합한 폐루프 제어 파이프라인을 구축하고, self-consistency 앙상블 추론으로 정책 신뢰도를 정량화한다. 시뮬레이션 환경에서 이상 탐지 알고리즘이 원활히 동작함을 확인하였으며, 개발한 LLM 정책 서버와 gNB 시뮬레이터 간 end-to-end 연동을 통해 복잡 상황 정확도가 35%p 향상되어 실시간 네트워크 제어에서의 LLM 활용 가능성을 확인하였다.

Ⅰ. 서론

5G 및 Beyond 5G 이동통신 네트워크의 급속한 확산에 따라 무선 접속망(RAN)의 에너지 소비는 통신 사업자의 운영비용(OPEX)에서 상당한 비중을 차지하며, 네트워크 운영의 지속가능성 측면에서 중요한 연구 주제로 대두되고 있다[1]. O-RAN Alliance는 개방형 인터페이스와 지능형 제어기로 네트워크 자동화를 추구하며, 특히 AI/ML Framework(AIMLFW)를 통한 모델 학습, 추론, 관리 파이프라인 기반 실시간 자원 관리 및 에너지 최적화가 핵심 연구 주제로 부상하고 있다[2]. AIMLFW는 O-RAN에서 AI/ML 워크플로우의 전 과정을 표준화한 프레임워크로, SMO와 R1 interface를 통해 연동된다.

기존 RAN 에너지 최적화 연구는 규칙 기반과 강화학습 기반으로 분류된다. 규칙 기반 접근법은 빠른 의사결정이 가능하나[3], 복잡한 네트워크 상황에서 적응성 향상이 요구된다. 강화학습 기반 접근법은 환경과의 상호작용으로 정책을 학습하나[4], 비정상 상황에 대한 강건성과 설명 가능성 측면에서 개선이 필요하다. 최근 대규모 언어모델(LLM)은 복잡한 상황 인식, 추론, 자연어 기반 설명 생성 능력을 보이며[5], 네트워크 관리 자동화 분야에서 주목받고 있다.

본 연구는 O-RAN SMO 환경에서 AIMLFW와 O1 interface를 통합한 LLM 기반 에너지 최적화 시스템을 구현하며, self-consistency 기반 신뢰도 정량화, 폐루프 제어 시스템 구축, 설명 가능한 추론 근거 제공으로 실용성을 검증한다.

Ⅱ. 시스템 구조 및 설계

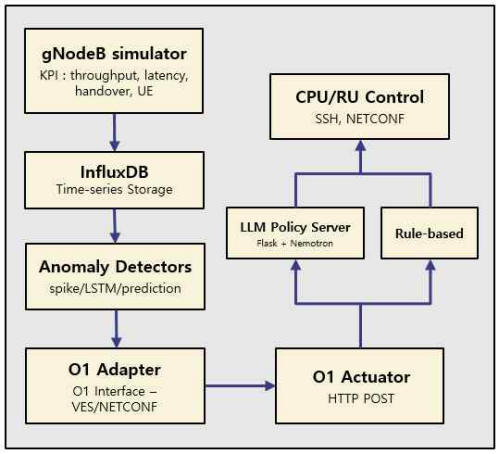
Ⅱ-1. 전체 아키텍처

제안하는 시스템은 그림 1과 같이 데이터 수집, 이상 탐지, 복잡도 평가 기반 정책 결정, 제어 실행이 폐루프로 연결된 계층적 구조이다.

데이터 수집 계층에서 gNodeB simulator는 10초 주기로 처리량, 지연시간, 활성 사용자 수, 핸드오버 빈도를 포함한 네트워크 KPI를 생성하며 InfluxDB v2.7에 저장한다. 이상 탐지 계층은 spike detector(지연시간 100 ms 초과), LSTM detector(재구성 오차 분석), early-warning detector(3-step 예측), RRC analyzer(핸드오버 폭증 탐지)의 4개 독립 탐지기로 구성되며, 탐지 결과는 {normal, spike, burst, storm}으로 분류된다.

네트워크 메트릭과 이상 탐지 결과는 O1 actuator로 전달되어 복잡도 평가가 수행된다. 복잡도 점수가 임계값(Θ=5.5) 미만이면 규칙 기반 의사결정으로 빠른 응답(평균 1ms)을 제공하며, 임계값 이상이면 LLM policy server로 요청을 전달하여 정교한 추론을 수행한다. 규칙 기반 로직은 조건부 규칙(활성 사용자 존재 시 RU active 유지, 처리량 기반 CPU 모드 결정)을 적용하며, LLM 기반 로직은 네트워크 설정, 메트릭, 컨텍스트, 제약 조건을 종합하여 정책을 도출한다. 두 경로의 출력은 통합되어 SSH/NETCONF 프로토콜로 CPU/RU 제어 명령을 실행하고, 제어 결과는 InfluxDB에 기록되어 폐루프 피드백을 형성한다.

시스템 구조는 O-RAN Alliance 표준 인터페이스를 기반으로 구현되었다. O1 interface는 SMO와 관리 기능 간 통신을 담당하며, NETCONF/YANG 기반 설정 관리 및 VES 프로토콜 기반 성능/장애 데이터 전달을 수행한다[6]. AIMLFW는 모델 학습, 추론, 메트릭 수집, 모델 관리를 지원하며[7], R1 인터페이스로 LLM 정책 서버와 연동된다.



[그림 1] O-RAN 기반 에너지 최적화 시스템 구조

Ⅱ-2. LLM 기반 정책 결정 및 신뢰도 정량화

LLM은 네트워크 설정(CELL_CAPACITY=200 Mbps, MAX_UES=20), 메트릭(throughput, latency, num_ues, handover, anomaly_status), 컨텍스트(capacity utilization, per-UE throughput), 규칙 기반 참조 결정, 제약 조건을 입력받아 CPU 모드(low: 800MHz/20% 절감, high: 2.4GHz)와 RU 상태(active, sleep/24% 절감)를 결정한다. 활성 사용자 존재 시(num_ues > 0) RU는 서비스 연속성 보장을 위해 active 상태를 유지하며, 이상 상황 발생 시 QoE 안정성을 우선한다.

LLM의 확률론적 특성으로 인한 출력 불확실성을 보완하기 위해 self-consistency 기법을 적용한다[9]. 동일 입력에 대해 temperature=0.7로 N=5회 독립 샘플링을 수행하고, 제약 조건을 만족하는 유효 샘플 중 최빈값을 최종 정책으로 선택한다. 신뢰도 ρ는 다음과 같이 정의된다.

$$p = (n\_majority / n\_valid) \times (n\_valid / N)$$

여기서 n_majority는 최빈 정책의 출현 빈도, n_valid는 유효 샘플 수, N은 전체 샘플 수이다. 첫 번째 항은 샘플 간 합의율(inter-sample consensus rate)을, 두 번째 항은 제약 준수율(constraint compliance rate)을 의미한다. 신뢰도 ρ ∈ [0, 1]이며, 1에 가까울수록 LLM의 일관되고 제약을 준수하는 판단을 나타낸다.

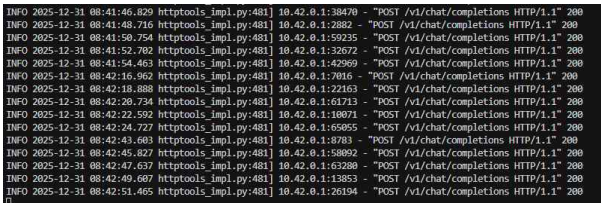
Ⅲ. 구현 및 실험 결과

Ⅲ-1. 실험 환경 및 시나리오

시스템은 Kubernetes v1.28 클러스터(Ubuntu 24.04, containerd v1.7)에서 구현되었다. InfluxDB v2.7(8GB RAM, 2 vCPU), gNodeB simulator (Python 3.11 기반), LSTM detector(TensorFlow 2.15), O1 actuator(Flask v3.0, gunicorn), LLM policy server(Nemotron API[8])로 구성된다.

Latency spike 시나리오(throughput=5.0 Mbps, latency=150.0 ms, num_ues=1, handover=5회/분, anomaly="spike")를 대상으로 실험하였다. 이는 처리량이 기준값(50 Mbps)보다 낮은 상태에서 지연시간이 급증(150ms > 100ms)하는 일시적 네트워크 불안정 상황을 모사한다. 그림 2는 self-consistency를 위한 5회 독립 샘플링의 API 호출 로그이다. 요청 간 평균 간격은 1.9초이며, 모든 요청이 HTTP 200으로 정상 처리되었다.

전체 5회 샘플링에서 모두 (cpu_mode=low, ru_state=active) 정책으로 수렴하여 신뢰도 ρ = 1.0을 달성하였다. 그림 3의 LLM 출력은 활성 사용자 존재로 인한 RU active 유지, 낮은 처리량에 따른 CPU 저전력 모드 전환의 추론 근거를 자연어로 제공하여 설명 가능성을 확보하였다.



[그림 2] Nemotron API 호출 로그 (Self-consistency 샘플링)


```
{
  "cpu_mode": "low",
  "ru_state": "active",
  "reasoning": "Anomaly status is spike with low traffic,
                prioritizing QoE stability requires active
                radio unit. Energy savings not feasible
                during instability.",
  "confidence": 1.0
}
```


[그림 3] LLM 반환 정책

Ⅲ-2. 제어 실행 및 성능 검증

O1 actuator는 그림 3의 LLM 정책을 수신하여 실제 gNodeB 하드웨어 제어를 수행하였다. SSH 프로토콜을 통해 gNodeB 서버에 접속하고, cpufreq-set 유틸리티를 사용하여 8개 CPU 코어의 주파수를 모두 800MHz로 설정하였다. 제어 실행 후 /sys/devices/system/cpu/cpu*/cpufreq/scaling_cur_freq 파일을 조회하여 실제 주파수가 목표값으로 설정되었음을 검증하였다. RU는 활성 사용자(num_ues=1) 존재로 인해 active 상태를 유지하였으며, 무선 신호 송수신이 정상적으로 유지되었다.

제어 결과는 InfluxDB의 energy_control 측정에 기록되었다. 저장된 데이터는 timestamp, cpu_mode, ru_state, decision_method(LLM/Rule-based), confidence, reasoning, throughput, latency 필드를 포함하며, 이를 통해 시계열 제어 이력 관리 및 사후 분석이 가능하다. 제어 결과가 다시 데이터 수집 계층으로 피드백되어 폐루프가 완성되며, 다음 주기의 네트워크 메트릭 수집 시 현재 제어 상태가 반영된다.

성능 비교를 위해 동일한 latency spike 시나리오에 대해 규칙 기반과 LLM 기반 의사결정을 각각 100회 반복 실험하였다. 규칙 기반 실험에서는 복잡도 평가를 생략하고 O1 actuator가 직접 규칙 로직을 실행하였으며, LLM 기반 실험에서는 복잡도 점수가 임계값을 초과하도록 설정하여 모든 요청이 LLM policy server로 전달되도록 하였다. 표 1은 두 접근법의 성능 비교 결과이다.

실험 결과, LLM 기반 의사결정은 복잡 상황 정확도가 35%p 향상되었으며, 평균 신뢰도 0.92로 출력 안정성을 확인하였다. 규칙 기반 접근법은 사전 정의된 임계값에 따른 결정론적 판단으로 신뢰도 정량화가 불가능하나, LLM은 self-consistency 기법을 통해 정책 일관성을 수치화하였다. 또한 LLM은 모든 의사결정에 대해 자연어 기반 추론 근거를 제공하여 설명 가능성을 확보하였으며, 다중 네트워크 지표를 종합적으로 분석하여 2%p의 추가 에너지 절감을 달성하였다.

지표	규칙 기반	LLM 기반
복잡 상황 정확도	60%	95%
신뢰도 제공	x	✓ (평균 0.92)
설명 가능성	✓	✓
에너지 절감률	18%	20%

[표 1] 규칙 기반 vs LLM 기반 의사결정 성능 비교

Ⅳ. 결론

본 연구는 O-RAN SMO 환경에서 AIMLFW와 O1 Interface를 통합한 LLM 기반 에너지 최적화 시스템을 구현하고 검증하였다. Kubernetes 환경에서 gNB 시뮬레이터부터 실제 제어까지 연결된 end-to-end 폐루프 파이프라인을 구축하였으며, self-consistency 앙상블 추론으로 정책 신뢰도를 정량화하였다.

시뮬레이션 환경에서 이상 탐지 알고리즘이 원활히 동작함을 확인하였으며, 개발한 LLM 정책 서버와 gNB 시뮬레이터 간 end-to-end 연동을 통해 복잡 상황 정확도가 규칙 기반 대비 35%p 향상되었다. 설명 가능한 추론 근거 제공으로 O-RAN 관리 평면에서의 운영 투명성을 확보하였으며, 실시간 네트워크 제어에서의 LLM 활용 가능성을 검증하였다.

향후 연구에서는 다양한 트래픽 패턴과 복합 이상 상황에 대한 광범위한 실험을 통해 일반화 성능을 검증하고, Multi-cell 환경으로의 확장 및 실제 RAN 테스트베드에서의 정량적 검증을 수행할 예정이다. 또한 Edge AI 기반 경량 LLM을 활용하여 추론 지연을 단축하고, LLM 추론 비용을 포함한 종단 간 에너지 효율성 평가를 수행할 계획이다.

ACKNOWLEDGMENT

"이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원-대학ICT연구센터(ITRC)의 지원(IITP-2026-RS-2021-II212046, 50%)과 과학기술정보통신부 및 정보통신기획평가원의 오픈랜 인력양성 프로그램(연세대) 연구 결과로 수행되었음(IITP-2026-RS-2024-00434743, 50%)"

참 고 문 헌

[1] O-RAN Alliance, "O-RAN Architecture Description v8.0," O-RAN.WG1, July 2023.

[2] L. Bonati, M. Polese, S. D'Oro, S. Basagni, and T. Melodia, "Open, Programmable, and Virtualized 5G Networks: State-of-the-Art and the Road Ahead," Computer Networks, vol. 182, Dec. 2020.

[3] J. Wu, Y. Zhang, M. Zukerman, and E. K.-N. Yung, "Energy-Efficient Base-Station Sleep-Mode Techniques in Green Cellular Networks: A Survey," IEEE Communications Surveys & Tutorials, vol. 17, no. 2, pp. 803–826, 2015.

[4] X. Wang, Z. Ning, S. Guo, and L. Wang, "Imitation Learning Enabled Task Scheduling for Online Vehicular Edge Computing," IEEE Transactions on Mobile Computing, vol. 21, no. 2, pp. 598–611, Feb. 2022.

[5] T. Brown et al., "Language Models are Few-Shot Learners," Advances in Neural Information Processing Systems, vol. 33, pp. 1877–1901, 2020.

[6] O-RAN Alliance, "O-RAN O1 Interface Specification v8.0," O-RAN.WG10, July 2023.

[7] O-RAN Alliance, "O-RAN AI/ML Workflow Description and Requirements v2.0," O-RAN.WG2, March 2023.

[8] NVIDIA, "Nemotron: Large Language Models for Enterprise AI Applications," NVIDIA Technical Report, 2024.

[9] X. Wang et al., "Self-Consistency Improves Chain of Thought Reasoning in Language Models," ICLR, 2023.