지능형 네트워크 구축을 위한 LLM 응용 동향 분석

홍준규, 김희원, 배찬빈, 김하은, 이상훈, 이정원, 구휘모, 백상헌
고려대학교

요 약

네트워크 규모의 확대와 애플리케이션 수요의 급증으로 인해 네트워크 운영의 복잡성이 증가하고 있으며, 6G 등 차세대 네트워크로의 진화에 따라 고도화된 성능 요구사항이 제기되고 있다. 전통적인 규칙 기반 네트워크 제어 방식의 비용 및 유연성 한계를 극복하기 위해 인공지능 (Artificial Intelligence, AI) 기반의 지능형 네트워크 기술이 주목받고 있으나, 이는 특정 작업에 국한된 모델 활용으로 인해 확장성과 일반화 측면에서 제약을 갖는다. 이를 보완하기 위해 다양한 입력과 작업에 유연하게 대응할 수 있는 대규모 언어 모델 (Large Language Model, LLM)을 네트워크 분야에 적용하려는 시도가 활발히 이루어지고 있다. 이에 따라, 본 논문에서는 네트워크 도메인 지식을 학습한 LLM을 기반으로, 네트워크 분석, 구성 및 최적화 등 다양한 측면에서 지능형 네트워크 구축하기 위한 최신 연구 동향을 분석한다.

Ⅰ. 서 론

네트워크 규모의 증가와 다양한 애플리케이션 수요의 급증으로 인해, 네트워크 운영 및 관리 작업의 복잡성이 빠르게 증가하고 있다. 더불어 6G 네트워크와 같은 차세대 네트워크로의 진화에 따라 초저지연성, 대규모 연결성 등과 같이 보다 고도화된 성능 요구사항이 제기되고 있다[1]. 그러나 전통적인 규칙 기반 (Rule-based) 네트워크 관리 방식은 전문가의 경험과 도메인 지식에 대한 높은 의존도로 인해 운용 비용이 증가하며, 복잡하고 동적인 네트워크 환경에 유연하게 대응하기 어렵다는 한계를 지닌다. 이에 따라, 다양한 애플리케이션 수요와 동적인 트래픽 변화에 능동적으로 대응하고, 적은 비용으로 높은 성능 요구사항을 충족하기 위해 지능적이고 자율적인 네트워크 운용 방식의 필요성이 대두되고 있다.

이러한 배경 속에서 인공지능 (Artificial Intelligence, AI) 기반의 지능형 네트워크 기술이 주목받고 있으며, 특히 기계학습 및 강화학습 모델을 활용한 사용자 트래픽 분석, 자원 할당, 성능 최적화 등에 대한 연구가 활발히 진행되고 있다[2]. 이를 통해 네트워크 관리를 자동화하여 관리 비용을 줄이고, 대규모 트래픽 데이터 내의 복잡한 관계를 효과적으로 학습함으로써 네트워크 시스템 성능 향상에 기여할 수 있다. 하지만 기존 AI 기반 지능형 네트워크 기술은 대부분 특정 작업에 특화된 모델을 활용하므로, 확장성과 일반화 능력 측면에서 한계를 갖는다[3]. 특히 네트워크 환경이나 작업 변화에 따라 기존에 학습된 모델의 성능이 저하될 수 있으며, 이를 개선하기 위한 모델 재학습 과정에서 상당한 시간과 비용이 수반된다는 문제가 발생한다.

한편, 최근 자연어 처리 분야를 중심으로 발전한 대규모 언어 모델 (Large Language Model, LLM)은 뛰어난 언어 이해 및 생성 능력을 바탕으로 다양한 분야에서 주목받고 있다. LLM은 트랜스포머 (Transformer) 신경망 구조를 기반으로 하며[4], 어텐션 (Attention) 메커니즘을 통해 문장 내 단어 간의 의미적 관계를 병렬적으로 효과적으로 학습함으로써, 고차원적 의미 추론과 복잡한 관계 표현이 가능하다. 대표적인 LLM으로는 GPT[5], LLaMA[6] 등이 있으며, 이들은 수십억 개 이상의 토큰으로 구성된 대규모 텍스트 데이터를 기반으로 사전 학습되어, 텍스트 생성, 질의 응답, 논리적 추론 등 다양한 작업에서 인간과 유사하거나 더 뛰어난 수준의 성능을 보여주고 있다. 또한, 최근에는 별도의 재학습 없이도 LLM이 새로운 상황에 대해 유연하게 적응할 수 있도록 하는 제로샷 (Zero-shot) 또는 퓨샷 (Few-shot) 학습 방식에 대한 연구가 활발히 이루어지고 있다[7]. 이와 같이 단일 모델로도 다양한 작업과 입력 데이터에 유연하게 적응할 수 있는 LLM은 기존 AI 모델의 확장성 및 일반화 한계를 극복할 수 있는 잠재력을 갖는다.

이에 따라, 뛰어난 확장성과 일반화 능력을 갖춘 LLM을 활용하여 보다 지능적인 네트워크를 구축하기 위한 시도가 활발하게 진행 중이다[3].

〈그림 1〉의 왼쪽은 LLM을 네트워크 분야에 적용한 사례를 보여준다. 예를 들어, 플로우별 발생 시간, 출발지 및 목적지 IP 주소, 플로우 크기로 구성된 최근 트래픽 기록을 입력으로 하여 해당 트래픽이 비정상적인지 판별하도록 LLM에 요청하면, LLM

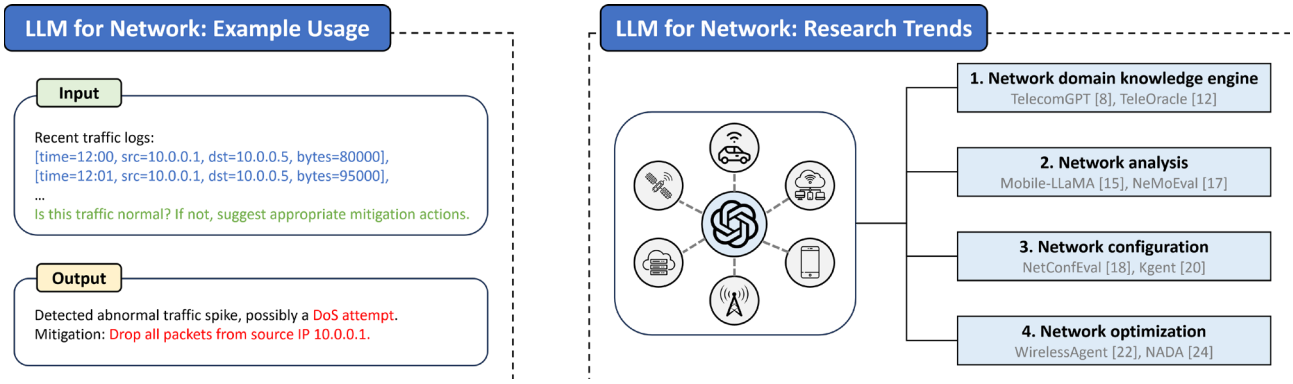


그림 1. 지능형 네트워크 구축을 통한 LLM 응용 사례 (왼쪽)와 연구 동향 (오른쪽).

은 학습된 네트워크 도메인 지식을 통해 이를 자동 분석하고, Denial of Service (DoS) 공격 여부를 판단할 수 있다. 이후 LLM은 해당 출발지 IP 주소를 갖는 모든 패킷에 대해 드롭하는 정책과 같은 대응 방안을 제안할 수 있다. 이와 같이, 최근에는 네트워크 도메인 지식을 학습한 LLM을 기반으로, 동적인 네트워크 환경에서 다양한 네트워크 작업을 효과적으로 처리하기 위한 연구가 활발히 이루어지고 있다. 이에 본 연구에서는 〈그림 1〉의 오른쪽과 같이 네트워크 도메인 지식 엔진 구축, 네트워크 분석, 구성 및 최적화 등 다양한 측면에서 LLM을 활용하여 지능형 네트워크 구축하고 고도화하기 위한 최신 연구 동향을 분석한다.

Ⅱ. 지능형 네트워크 구축을 위한 LLM 응용 동향

본 절에서는 네트워크 도메인 지식 엔진 구축, 네트워크 분석, 구성 및 최적화 등의 측면에서 차세대 지능형 네트워크를 구축하기 위해 LLM을 활용한 최신 연구 동향에 대해 알아본다.

1. 네트워크 도메인 지식 엔진

6G와 지능형 네트워크 시대를 맞이하여, 네트워크 시스템의 구조는 점점 더 복잡해지고 있으며, 그에 따른 다양한 표준들이 제안되고 있다. 특히 3GPP 및 O-RAN Alliance를 포함한 다양한 표준 개발 기구에서 발표되는 방대한 기술 문서는 그 양과 복잡성으로 인해 사람의 수작업만으로는 즉각적인 이해와 활용을 하기에는 한계가 있다. 이러한 복잡한 프로토콜 구조와 빠르게 변화하는 통신 네트워크의 기술 생태계에 대응하기 위해, 도메인 특화 지식의 체계적 해석과 활용을 위한 언어 이해 기반 지식 엔진에 대한 연구가 진행되고 있다. 이러한 요구 속에서, LLM은 복잡한 기술 문서를 자연어 수준에서 해석하여 문맥 기반 문제 처리와 추론을 수행하고, 새로운 문제 상황에도 유연하게 대응할 수 있다는 점에서 네트워크 도메인의 지식 처리에 적합한 도구로 부상하였다. 그러나 범용 LLM은 여전히 통신 도메인에 특화된 지식 부족, 응답 신뢰도 문제, 입력 길이 제한, 그리고 자원 효율성 측면에서 한계를 지니고 있으며, 최근 이를 개선하기 위한 연구가 활발히 제안되고 있다.

가. TelecomGPT[8]

범용 LLM의 한계를 극복하고자 TelecomGPT는 범용 LLM을 네트워크 도메인에 적합하도록 학습하기 위한 파이프라인을 제안하였다. TelecomGPT의 사전학습 파이프라인은 〈그림 2〉와 같으며, 연속 사전학습 (Continual pretraining), 지시 튜닝 (Instruction tuning), 정렬 튜닝 (Alignment tuning)의 세 단계를 통해 LLM이 통신 도메인의 다양한 업무를 수행할 수 있도록 최적화한다. 제안하는 시스템에서는 도메인 연속 사전학습을 위해 3GPP 표준, IEEE 문서, 논문, 특허 등에서 수집한 대규모 텔레콤 전용 데이터셋 (OpenTelecom)을 구축하였으며, 고유 키워드 기반 필터링과 중복 제거로 데이터 품질을 향상시켰다. 또한, 기술 문서 분류, 질의응답 (Question and Answering, QA),

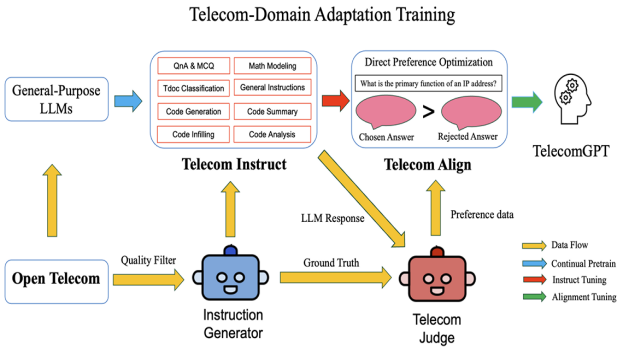


그림 2. TelecomGPT의 사전학습 파이프라인[8].

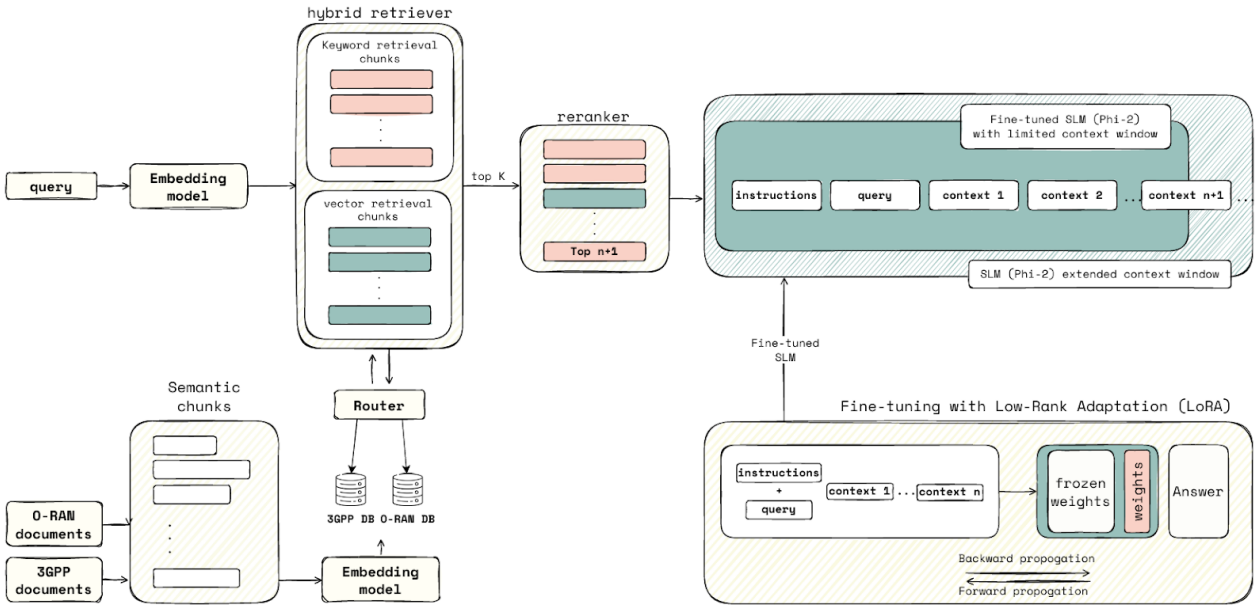


그림 3. TeleOracle 구조[12].

수학 모델링, 코드 생성 등 실제 네트워크 운용 시나리오를 반영한 데이터셋을 구성하고, 경량화된 파인튜닝 (Fine-tuning) 기법인 QLoRA[9] 기반의 지시 튜닝을 통해 모델이 사용자 중심의 지시 수행이 가능하도록 하였다. 마지막으로, 정렬 튜닝 단계에서는 명시적인 보상 모델 없이도 사용자의 선호에 맞는 응답을 선택적으로 학습할 수 있는 Direct Preference Optimization (DPO) 기법을 적용하였다. 이러한 DPO 기법을 활용하여 TelecomGPT는 LLM의 응답이 간결함, 정확성 등의 사용자 선호도와 일치하도록 하였다.

Llama-3-8B[10] 모델을 기반으로 학습한 TelecomGPT의 성능 평가 결과, 3GPP 표준 문서 분류, 수학 모델링, 코드 생성 등 다양한 통신 특화 작업에서 GPT-4[5]와 비슷하거나 그 이상의 성능을 보여주었다. 특히 수학 모델링 벤치마크에서는 수식 간 수학적 구조 유사도를 측정하는 MathBERT[11]를 기준으로, 90% 이상 유사한 수식을 생성한 비율이 GPT-4 대비 약 2.5배 높았다. 또한 표준 문서 분류 작업에서는 GPT-4o에 비해 약 36.4% 높은 정확도를 보여주었으며, 코드 요약 및 분석 작업에서도 정답과 생성 응답 간의 문장 수준 유사도를 나타내는 Rouge-L 점수가 최대 0.39에 도달하며, 도메인 특화 작업에 대한 높은 수행력을 입증하였다. 이를 통해 TelecomGPT는 단순한 질의응답을 넘어, 기술 문서 해석, 문제 상황 분석, 수식 유도, 코드 처리 등 다양한 고차원 작업을 수행할 수 있는 도메인 특화형 지식 엔진으로 기능하는 것을 확인할 수 있다. 또한 경량화된 파인튜닝을 통해 제한된 자원으로도 높은 성능을 달성할 수 있어, 향후 통신망의 실시간 제어, 자율 운용 등에 효과적으로 활용될 수 있는 가능성을 보여주었다.

나. TeleOracle[12]

한편, 실제 네트워크 환경에서 대형언어모델의 직접적인 적용의 어려움을 극복하기 위해 TeleOracle은 소형언어모델 (Small Language Model, SLM)을 기반으로 경량화한 검색 증강 생성 (Retrieval-Augmented Generation, RAG) 시스템을 제안한다. TeleOracle은 자원 제약이 존재하는 실제 네트워크 환경에서도 SLM이 높은 정확도와 문맥 신뢰도를 유지하며 통신 도메인의 질의응답을 처리할 수 있도록 한다.

TeleOracle의 전반적인 구조는 〈그림 3〉과 같으며, 마이크로소프트의 SLM인 Phi-2[13]를 기반으로, 네트워크 도메인 문서의 구조적 특성과 의미 연계를 효과적으로 반영하기 위해, 여러 핵심 기능을 제안하였다. 먼저, Semantic chunking 기법을 통해 문서를 고정 길이 기준이 아닌 의미 단위로 분할함으로써, 문맥 내 논리적 일관성을 유지한 채 정보 검색의 품질을 높였다.

이후, 질의와 관련된 정보를 효율적으로 추출하기 위해 BM25 알고리즘 기반 키워드 검색과 임베딩 기반 벡터 검색을 병행하는 하이브리드 검색기를 구성하였으며, 이를 통해 단순 표면적 유사성과 의미적 유사성을 동시에 반영할 수 있도록 하였다. 검색된 문서 조각들은 문맥과 질의 간 정밀한 상호작용을 반영할 수 있도록 설계된 Cross-encoder 기반 Reranker에 의해 재정렬되어, 가장 관련성 높은 정보가 우선적으로 모델 입력에 포함되도록 한다.

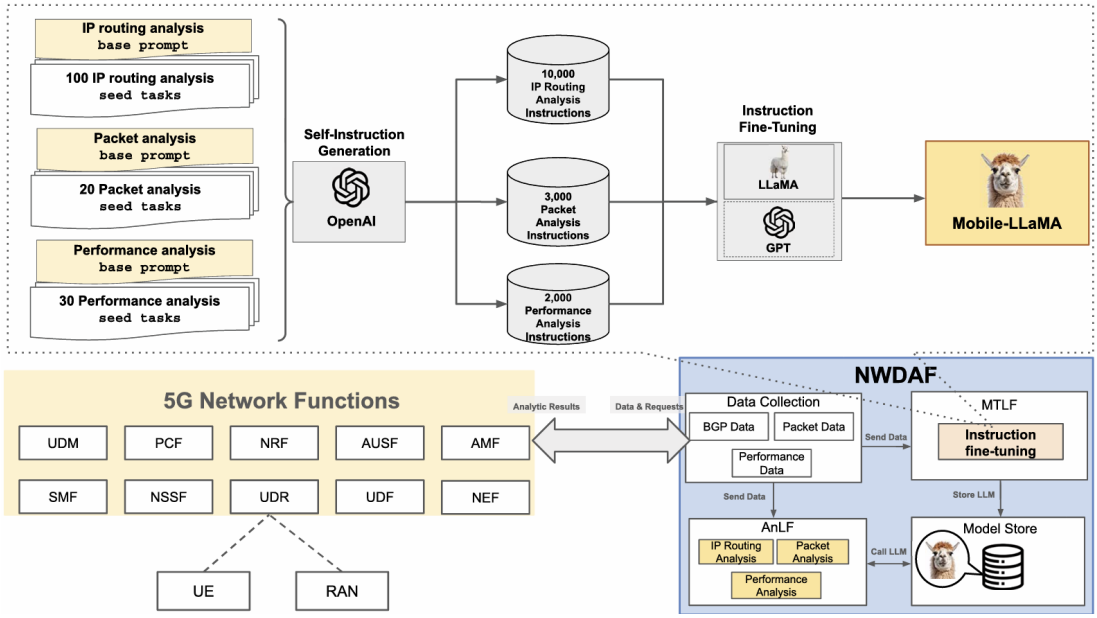


그림 4. Mobile-LLaMA 시스템 모델[15].

또한, 기존 SLM의 한계로 지적되던 짧은 문맥 처리 능력을 보완하기 위해, 다수의 문서 조각을 연결하여 최대 8,000개 이상의 토큰을 처리할 수 있는 SelfExtend 기법을 도입하였다. 이를 통해 긴 표준 문서나 복잡한 설명이 포함된 질의에 대해서도 문맥을 유지한 응답 생성이 가능하게 되었다. 모델 학습은 네트워크 도메인의 질의응답 데이터를 바탕으로 파인튜닝 기법인 LoRA[14] 방식으로 수행되었으며, 질의의 의미에 따라 3GPP, O-RAN Alliance 등의 관련 문서군으로 자동 라우팅하는 Semantic router 기능도 포함하여, 실시간 문맥 탐색과 정확한 정보 응답이 가능하도록 했다.

실험 결과, TeleOracle은 동일한 기본 모델 대비 최대 30% 이상 향상된 정확도를 기록하였으며, 응답의 문맥 신뢰도 또한 78.8%로 높은 수준을 보였다. 특히 학습에 포함되지 않은 도메인에 대해서도 별도 튜닝 없이 강한 전이 성능을 발휘하여, 실용적으로 확장 가능한 지식 엔진을 구축하였다.

2. 네트워크 분석

최근 모바일 네트워크의 구조적 복잡성 및 트래픽 급증으로 인해 네트워크 분석 작업의 자동화 필요성이 크게 증가하고 있다. 특히 에러 감지, 트래픽 분석, 장애 대응 등 정밀한 분석이 요구되는 네트워크 작업에서 자연어 문맥을 이해하고 분석 코드를 자동으로 생성할 수 있는 LLM이 유망한 대안으로 부상하고 있다.

가. Mobile-LLaMA[15]

Mobile-LLaMA는 5G 코어 네트워크에서 데이터 수집, 모니터링, 분석을 수행하는 Network Data Analysis Function (NWDAF) 기능을 LLM으로 구현하는 시스템 모델을 제안한다. 이를 위해서는 사전 학습된 모델을 기반으로 문제 도메인에 특화된 데이터셋으로 파인튜닝 하는 과정이 필요하다. 이때 지시 파인튜닝 (Instruction fine-tuning) 은 모델이 수행할 작업을 설명하는 명령어를 학습에 포함시키는 방식으로, 모델의 응답 품질에 직접적인 영향을 미친다. 하지만, 고품질의 지시 데이터를 확보하기 어려워, 충분한 모델 성능을 보장하기 힘들다는 한계가 존재한다. Mobile-LLaMA는 이를 해결하기 위해 자가-지시 (Self-instruction) 기법을 기반으로 한 시스템 구조를 제안하며, 〈그림 4〉와 같이 데이터 수집 (Data collection), 모델 학습 논리 함수 (Model Training Logical Function, MTLF), 모델 저장소 (Model store), 분석 논리 함수 (Analytics Logic Function, AnLF)와 같은 4개의 논리 구조로 구성된다. 우선, 모바일 코어 네트워크의 네트워크 함수 간의 인터페이스를 통해 패킷, 성능, Border Gateway Protocol (BGP) 데이터 등을 수집한 뒤, 모델 학습 논리 함수를 통해 LLM을 학습시키고, 학습된 모델은 모델 저장소에 보관된다. 이후, 분석 논리 함수에서 NWDAF 기능 수행 시, 학습된 LLM이 호출되어 네트워크 데이터 분석을 수행한다.

모델 학습 논리 함수에서는 지시 데이터 부족 문제를 해결하기 위해 자가-지시 기법을 적용하여, 사전 학습된 OpenAI의 GPT 모델[5]을 활용해 고품질의 시드 데이터 (Seed data)를 기반으로 새로운 지시를 생성하고, 기존 데이터와의 유사도 함수인

Rouge-L이 임계값 이하인 경우에만 추가 학습 데이터로 활용한다. 이는 데이터 다양성을 확보해 모델의 과적합을 방지하고 다양한 데이터를 기반으로 추론 성능을 향상시킨다. 이렇게 학습된 LLM은 라우팅, 패킷 분석, 성능 분석과 관련된 파이썬 스크립트를 자동 생성하며, 문법 오류, 적절한 라이브러리 사용 여부, NWDAF 기능 신뢰도 등의 지표를 기준으로 평가된다.

평가 결과, 자가-지시 방법을 적용한 Mobile-LLaMA는 시드 데이터만으로 학습된 기존 LLaMA 모델[6]에 비해 높은 성능을 보였으며, LLaMa 2 70B[16]보다도 LLaMa 2 13B 기반의 Mobile-LLaMA가 더 우수한 결과를 보여주었다.

나. NeMoEval[17]

NeMoEval은 복잡한 네트워크 토폴로지 및 통신 그래프를 효율적으로 관리하기 위한 접근법으로, LLM이 사용자의 요청 프롬프트로부터 네트워크 분석 코드를 생성하여 실행하는 시스템 구조를 제안한다. 기존의 LLM 기반 네트워크 애플리케이션들은 설명 가능성, 확장성, 개인정보 문제 등에 관한 여러 한계를 지니고 있으며, 특히 네트워크 특성 중 하나인 대규모 토폴로지를 프롬프트로 입력하는 방식은 토큰 및 연산 한계로 인해 실질적인 적용이 어렵다.

이를 해결하기 위해 제안된 NeMoEval 시스템은 자연어 기반 네트워크 분석 및 관리 프레임워크를 구축하며, 입력 데이터를 기반으로 그래프 처리용 코드를 생성한 후, 이를 샌드박스 환경에서 실행함으로써 안정성과 보안성을 확보한다. 코드 결과는 사용자가 검토 가능하도록 제공하며, 이 과정을 통해 LLM 응답의 논리성 및 연산 방식을 검증할 수 있다. 논문에서 제안하는 시스템 모델은 〈그림 5〉와 같으며, 크게 애플리케이션 래퍼 (Application wrapper), 애플리케이션 프롬프트 생성기 (Application prompt generator), 코드 프롬프트 생성기 (Code-Gen prompt generator), 샌드박스 실행 (Execution sandbox)으로 구성된다. 애플리케이션 래퍼는 애플리케이션 데이터로부터 도메인 특화 정보인 애플리케이션 정보와 네트워크 환경 정보를 분리하여 추출한다. 이를 통해 데이터에서 구성 요소 및 관계 그래프를 자연어로 생성하고, 애플리케이션 프롬프트 생성기로 전달한다. 다음으로, 애플리케이션 프롬프트 생성기에서는 애플리케이션 래퍼로부터 전달된 그래프 기반 문맥 정보와 사용자 질의를 결합해 LLM에 입력할 자연어 프롬프트를 생성한다. 이후, 코드 프롬프트 생성기는 프롬프트에 코드 출력 명령을 명시하는 프롬프트를 결합하여, LLM이 코드 생성을 수행할 수 있도록 최종 프롬프트를 구성한다. LLM은 NetworkX, Pandas와 같은 파이썬 라이브러리를 기반으로 한 분석 코드를 생성하고, 질의 의도에 맞는 그래프 연산, 정보 추출 등을 수행하도록 한다. 최종적으로, LLM이 생성한 코드를 안전하게 실행하고 결과를 검증하기 위해 가상화 또는 컨테이너화된 샌드박스 환경에서 코드를 실행하고, 이 과정에서 발생한 오류 여부, 출력 양식의 적합성 등을 검토한다. 이러한 과정을 거쳐 최종 실행 결과는 사용자에게 제공된다.

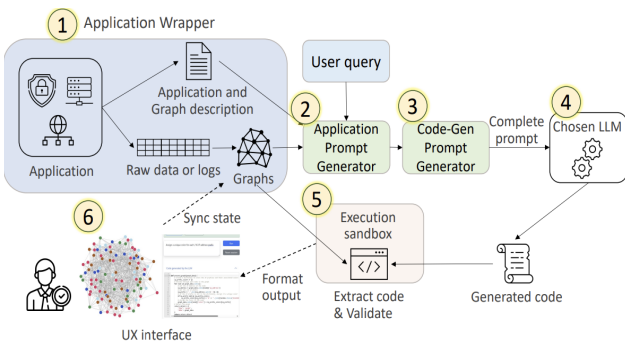


그림 5. NeMoEval 시스템 모델[17].

성능 평가에서는 네트워크 트래픽 분석, 라이프사이클 관리 등의 작업에 대해 LLM이 생성한 코드의 실행 결과를 기준으로 평가가 이루어졌다. 평가 결과, 다양한 네트워크 분석 및 관리 작업에서 LLM 기반 코드 생성이 안정적으로 동작함을 확인할 수 있었다. 특히, 복잡한 그래프 구조에 대한 사용자의 질의를 코드로 변환함으로써, 수작업 없이도 높은 수준의 분석 자동화를 달성할 수 있다는 점에서 실효성이 입증되었다.

3. 네트워크 구성

네트워크 구성은 스위치, 라우터, 서버 등 네트워크 장비들의 동작 방식과 정책을 정의하는 필수 작업으로, 네트워크 서비스 제공을 위한 기반 역할을 수행한다. 그러나 네트워크 인프라의 규모 및 복잡성이 증가하고, 서비스 요구사항이 다양해짐에 따라 구성 작업의 난이도와 부담은 크게 증가하고 있다. 운영자는 각 장비별 프로토콜, API, 구성 문법을 숙지하고, 수백 줄 이상의 Config 파일이나 복잡한 API 호출 코드를 직접 작성해야 하며, 구성 오류나 변경 발생 시 수작업 수정에 의존해야 한다.

이러한 한계를 해결하기 위해 최근 LLM을 활용한 네트워크 구성 자동화 기술이 주목받고 있다. LLM은 자연어 이해와 코드 생성 능력을 바탕으로, 사람이 기술한 고수준 네트워크 요구사항을 기계가 이해 가능한 저수준 구성 정보로 자동 변환할 수 있다. 특히 일부 연구에서는 검증기나 피드백 루프를 활용해 LLM이 생성한 구성 결과의 오류 여부를 자동 검증하고, 필요 시 프롬프트를 보완하거나 결과를 재생성하는 방식까지 제안하고 있어, LLM을 활용한 네트워크 구성 작업을 통해 신뢰성과 유지관리 효율성을 크게 향상시킬 수 있는 가능성을 보여주고 있다.

그림 6. NetConfEval의 Stand-alone 프로토타입 (왼쪽)과 Synthesizer 기반 프로토타입 (오른쪽)[18].

가. NetConfEval[18]

NetConfEval 연구는 LLM 기반 네트워크 구성 자동화의 기술적 가능성과 한계를 검증하기 위해 체계적인 벤치마크 환경을 설계하였다. 또한, 네트워크 구성 과정에서 운영자가 수행하는 주요 작업들을 자연어 입력부터 스펙 변환, API/함수 호출, 라우팅 알고리즘 코드 생성, 저수준 장비 구성 자동화 등 일련의 단계로 구분하여 LLM의 수행 능력을 실험적으로 평가하였다. 이러한 작업별 실험 결과를 바탕으로, NetConfEval은 효과적인 LLM 기반 네트워크 구성 시스템 설계를 위해 총 3가지 설계 원칙을 도출하였다. 첫째, 복잡한 작업을 여러 개의 작은 작업으로 나누어 처리함으로써 수행 정확도를 높이고, 작업별로 적절한 LLM 또는 외부 도구를 선택할 수 있도록 하였다. 둘째로는, 작업 특화 검증기 (Task-specific verifier)를 활용하여 LLM 결과의 오류를 자동으로 검증하고, 오류 발생 시 LLM에 요청을 재입력하여 수정된 결과를 생성하도록 유도하는 방식이 필요함을 제안하였다. 마지막으로, 운영자가 프롬프트를 정교하게 설계하거나 필요시 수동 피드백을 제공하는 Human-in-the-loop 구조를 통해 LLM 성능을 보완하는 설계가 요구됨을 제시하였다.

NetConfEval은 이러한 설계 원칙을 기반으로, 〈그림 6〉과 같이 자연어 요구사항 입력으로부터 네트워크 장비를 설정하는 전체 과정을 자동화하는 두 가지 프로토타입을 설계하고 구현하였다. 첫 번째 프로토타입은 LLM만을 활용해 별도의 파인튜닝 없이 자연어 요구사항으로부터 데이터 평면 구성 언어인 P4[19] 코드를 자동 생성하는 Stand-alone 구조이며, 두 번째 프로토타입은 기존 네트워크 구성 도구 (Synthesizer)와 LLM을 연계하여 자연어 기반 사용자 인터페이스와 기존 시스템의 호환성을 함께 제공하는 구조로 설계되었다. 실험 결과, 각 프로토타입은 처리 과정 내에서 작업 단위 처리, 검증기 기반 LLM 결과 검증, 오류 발생 시 재생성, 운영자 개입 기반 Human-in-the-loop 설계 등 연구에서 제시한 설계 원칙을 반영하여 실제 네트워크 구성 자동화 환경에서 효과적으로 동작함을 입증하였다.

나. Kgent[20]

extended Berkeley Packet Filter (eBPF)[21]는 운영체제 커널 내부에서 네트워크 트래픽 모니터링, 성능 분석, 보안 정책 구현 등을 지원하는 기술로 널리 활용되고 있다. 그러나 커널 내부 구조에 대한 높은 이해와 루프, 메모리 접근 제한 등 eBPF 검증기의 프로그래밍 제약으로 인해 일반 개발자나 운영자가 쉽게 활용하기 어려운 한계가 존재한다. Kgent 연구는 이러한 문제를 해결하기 위해, 운영자가 자연어로 네트워크 구성 의도를 입력하면 자동으로 eBPF 프로그램을 생성하고 검증해주는 시스템을 제안하였다.

Kgent의 동작 절차는 〈그림 7〉과 같다. 먼저, 사용자가 "TCP 연결 시 IP 주소와 포트 번호를 기록하라"와 같은 문장을 입력하면, LLM은 이를 기반으로 eBPF 코드 후보를 자동 생성한다. 이후, 프로그램 동작 의미를 보다 명확히 하고 검증 가능하도록 Program comprehension 기법을 활용해 각 코드 부분에 호어 논리 기반의 동작 전·후 조건을 자동으로 정의한다. 이러한 조건이 부여된 프로그램이 모든 상황에서 올바르게 동작하는지를 확인하기 위해, Symbolic execution 기법을 적용해 프로그램의 모든 실행 경로를 논리적으로 분석한다. Symbolic execution은 변수들을 기호로 처리해 가능한 실행 경로를 추론하고, 조건 위반 여부는 Satisfiability Modulo Theories (SMT) solver를 통해 검증한다. 검증 과정에서 오류가 발생하면, 해당 오류 메

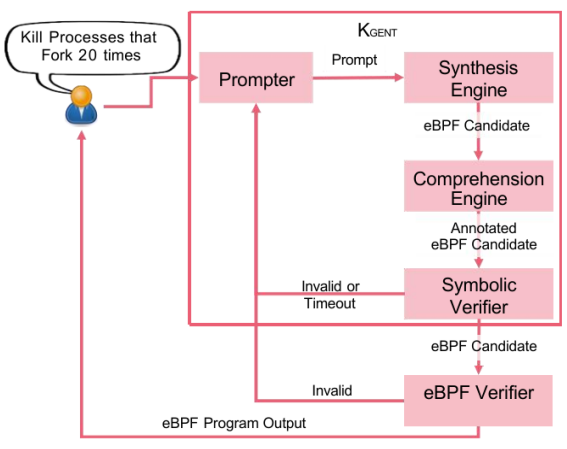


그림 7. Kgent의 동작 절차[20].

시지를 LLM의 입력에 추가하여 새로운 프로그램을 재생성하는 피드백 루프 방식으로 동작한다. 이러한 과정을 거쳐 Symbolic execution 검증을 통과한 프로그램은 eBPF 검증기를 통해 커널 수준의 안전성 검사를 최종 수행한 후 실행 가능한 eBPF 프로그램으로 출력된다.

Kgent는 이러한 구조적 설계를 통해 기존 GPT-4[5] 대비 약 2.67배 높은 네트워크 구성 코드 생성 정확도를 기록하였으며, False positive 발생률을 최소화하는 데 성공하였다. 해당 기법은 LLM 기반 코드 생성, Program comprehension, Symbolic execution 검증, 피드백 루프 설계를 유기적으로 결합하였으며, LLM을 통한 eBPF 기반 커널 구성 자동화의 실용성과 안정성을 보여주었다.

4. 네트워크 최적화

무선 통신 네트워크는 높은 복잡도, 동적 환경, 다양한 트래픽 요구사항으로 인해 기존의 기계학습 기반 최적화 솔루션만으로는 실시간 의사결정의 한계에 직면하고 있다. 이에 따라 높은 일반화 능력을 갖는 LLM을 활용해 다양한 네트워크 문제를 해결하고 성능을 최적화하기 위한 연구들이 활발하게 진행되고 있다.

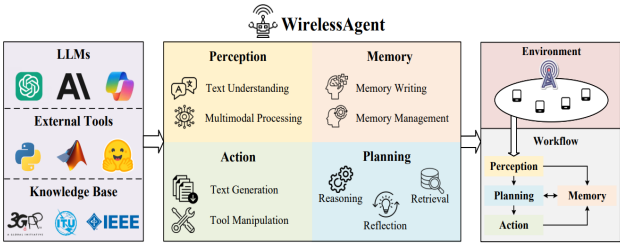


그림 8. WirelessAgent 프레임워크[22].

가. WirelessAgent[22]

WirelessAgent는 LLM을 기반으로 무선 네트워크 환경에서 복잡한 상황을 이해하고, 자율적으로 네트워크 최적화 결정을 수행할 수 있는 지능형 에이전트의 역할을 수행한다. 〈그림 8〉과 같이 WirelessAgent 프레임워크는 관측 (Observation), 추론 (Reasoning), 실행 (Action), 피드백 (Feedback)의 4단계 모듈로 구성된다. 먼저 관측 단계에서 실시간으로 수집된 네트워크 상태 정보를 Structured-to-text 변환기를 통해 자연어 프롬프트로 변환하여, LLM이 이해할 수 있는 입력으로 재구성한다. 다음으로 추론 단계에서는 변환된 입력을 LLM이 Chain-of-Thought (CoT)[23] 방식으로 문제를 분석하고, 과거 사례 및 도메인 지식을 활용해 네트워크 최적화 전략을 도출한다. 이후 실행 단계에서 LLM의 추론 결과가 자연어 기반 제어 명령으로 생성되어 실제 네트워크에 적용되며, 별도의 파라미터 튜닝 없이도 프롬프트 조정만으로 다양한 상황에 대응할 수 있다. 마지막으로 피드백 단계에서는 제어 결과에 대한 성능 지표를 LLM에 피드백함으로써 모델을 개선하고 의사 결정 성능을 향상시킨다. 이때 검색 증강 생성 기법을 활용하여 과거의 유사 경험을 검색함으로써 보다 정교하고 일관된 추론이 가능하도록 지원하였다.

실험 결과, WirelessAgent는 별도의 도메인 지식 없이 LLM 기반의 추론만으로도 안정적인 네트워크 최적화 성능을 나타냈다. 예를 들어, 네트워크 슬라이스 환경에서 WirelessAgent는 다양한 수의 사용자를 지원하기 위해 할당되는 Resource block의 수를 기존의 전통적 방식에 대비하여 평균 3.2% 감소시킴으로써, 동적으로 변화하는 네트워크 상황에서도 우수한 적응력을 보였다. 또한, 강화학습 기반 모델과 비교하여 학습에 필요한 비용이 현저히 낮으며, 특정 환경에서 학습된 모델을 다른 시나리오에 대한 전이 학습 없이 즉시 적용할 수 있다는 강점이 확인되었다. 이를 통해 WirelessAgent는 LLM이 단순한 텍스트 생성 모델이 아닌, 실제 네트워크 운영 환경에서 상황 인지, 추론, 최적화, 학습의 전 과정을 자율적으로 수행할 수 있는 지능형 에이전트로 확장할 수 있음을 입증하였다.

나. NADA[24]

NADA는 기존 네트워크 문제 해결 알고리즘들의 코드를 학습한 LLM을 활용하여 네트워크 최적화 알고리즘을 자동으로 설계하는 프레임워크를 제안하며, 그 동작 절차는 〈그림 9〉와 같다. 먼저 패킷 라우팅, 로드 밸런싱, 스케줄링과 같이 해결하고자 하는 네트워크 문제를 명확하게 정의한 후, 이를 LLM이 이해할 수 있도록 자연어 프롬프트 형태로 변환한다. 이와 같이 구조화된 프롬프트를 LLM에 입력하면, 모델은 학습을 통해 습득한 일반화된 알고리즘 지식을 바탕으로 문제 해결을 위한 강화학습 알고리즘 후보군을 자연어 혹은 의사코드 형태로 생성한다. 이때, 계산 비용의 절감을 위해 후보군 내 알고리즘들의 초기 학습 성능을 바탕으로 낮은 성능을 보이는 알고리즘을 제외하는 조기 중단 기법을 도입하였다. 이후 선택된 강화학습 알고리즘은 실

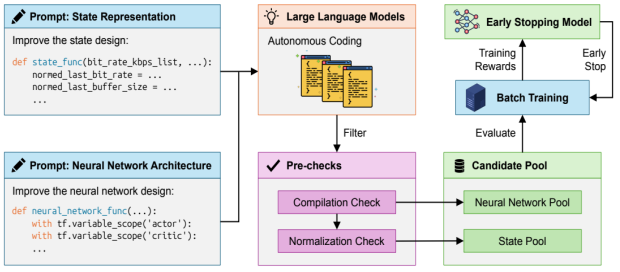


그림 9. NADA 동작 절차[24].

제 시뮬레이션 환경에서 평가되며, 그 결과를 기반으로 LLM의 출력 프롬프트를 수정하거나, 강화학습이나 파인튜닝을 통해 모델 성능을 개선한다.

미국 전역의 실제 인터넷 인프라 데이터에 해당하는 FCC[25], Starlink 위성, 4G, 5G 등 현실적인 네트워크 환경의 데이터셋을 활용하여 실험한 결과, 각 환경에서 NADA가 생성한 알고리즘은 Pensieve[26] 기법 대비 일관되게 우수한 영상 스트리밍 최적화 성능을 보였다. 5G 환경에서는 NADA가 생성한 알고리즘이 평균 QoE에서 Pensieve를 8% 이상 상회하였으며, 조기 중단 메커니즘을 통해 전체 알고리즘 후보의 약 80%를 조기에 제거하여 계산 비용을 약 60% 절감하는 결과를 나타냈다. NADA는 수작업 설계나 강화학습의 한계를 넘어, 다양한 네트워크 환경에 적응 가능한 제어 알고리즘을 효율적으로 생성할 수 있음을 입증했다. 이러한 방식을 통해 향후 네트워크 최적화 알고리즘의 자동화와 고도화에 크게 기여할 것으로 기대된다.

Ⅲ. 결 론

본 논문에서는 LLM의 뛰어난 확장성 및 일반화 능력을 바탕으로 지능형 네트워크를 구축하기 위한 최신 연구 동향에 관해 분석하였다. 특히, 네트워크 도메인 지식 엔진 구축, 네트워크 분석, 구성 및 최적화 등의 다양한 측면에서 LLM을 네트워크 분야에 적용하려는 연구가 활발히 진행되고 있음을 확인하였다. 향후에는 관련 연구가 보다 다양한 네트워크 작업과 응용 분야로 확장되고 고도화됨으로써, LLM의 네트워크 분야에 대한 기여도 및 활용 가능성이 증대될 것으로 기대된다.

Acknowledgement

본 논문은 2025년도 정부(과학기술정보통신부)의 재원으로 한국연구재단 중견연구 (No. RS-2024-00341965) 및 IITP 네트워크 연구센터 (NRC) (No. RS-2024-00398948)의 지원을 받아 수행된 연구임

참 고 문 헌

[1] C. -X. Wang et al., "On the Road to 6G: Visions, Requirements, Key Technologies, and Testbeds," IEEE Communications Surveys & Tutorials, vol. 25, no. 2, pp. 905-974, February 2023.

[2] A. Banchs et al., “Network intelligence in 6G: Challenges and opportunities,” in Proc. ACM Workshop on Mobility in the Evolving Internet Architecture (MobiArch), New Orleans, LA, USA, April 2022.

[3] H. Zhou et al., "Large Language Model (LLM) for Telecommunications: A Comprehensive Survey on Principles, Key Techniques, and Opportunities," IEEE Communications Surveys & Tutorials, to appear.

[4] A. Vaswani et al., “Attention is all you need,” in Proc. Advances in Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, December 2017.

[5] OpenAI, “GPT-4 Technical Report,” arXiv preprint arXiv:2303.08774, March 2023.

[6] H. Touvron et al., “Llama: Open and Efficient Foundation Language Models,” arXiv preprint arXiv:2302.13971, February 2023.

[7] T. Brown et al., “Language Models are few-shot learners,” in Proc. Advances in Neural Information Processing Systems (NeurIPS), Virtual Event, December 2020.

[8] H. Zou et al., "TelecomGPT: A framework to build telecom-specific large language models," arXiv preprint arXiv:2407.09424, July 2024.

[9] T. Dettmers et al., "Qlora: Efficient finetuning of quantized llms," in Proc. Advances in Neural Information Processing Systems (NeurIPS), New Orleans, LA, USA, December 2023.

[10] A. Grattafiori et al., “The Llama 3 Herd of Models,” arXiv preprint arXiv:2307.21783, July 2024.

[11] J. T. Shen et al., "Mathbert: A pre-trained language model for general nlp tasks in mathematics education," arXiv preprint arXiv:2106.07340, June 2021.

[12] N. Alabbasi et al., "TeleOracle: Fine-Tuned Retrieval-Augmented Generation With Long-Context Support for Networks," IEEE Internet of Things Journal, to appear.

[13] Phi-2, Accessed: April 15, 2025.[Online]. Available: https://www.researchgate.net/profile/Gustavo-

De-Rosa/publication/385654002_Phi-2_The_surprising_power_of_small_language_models/links/672e175eecbbde716b61db34/Phi-2-The-surprising-power-of-small-language-models.pdf

[14] E. J. Hu et al., “LoRA: Low-rank adaptation of large language models,” in Proc. International Conference on Learning Representations (ICLR), Virtual Event, April 2022.

[15] K. B. Kan et al., “Mobile-LLaMA: Instruction Fine-Tuning Open-Source LLM for Network Analysis in 5G Networks,” IEEE Network, vol. 38, no. 5, pp. 76-83, September 2024.

[16] H. Touvron et al., “Llama 2: Open Foundation and Fine-Tuned Chat Models,” arXiv preprint arXiv:2307.09288, July 2023.

[17] S. K. Mani et al., "Enhancing Network Management Using Code Generated by Large Language Models," in Proc. ACM Workshop on Hot Topics in Networks (HotNets), Cambridge, MA, USA, November 2023.

[18] C. Wang et al., "NetConfEval: Can LLMs facilitate network configuration?," in Proc. ACM International Conference on emerging Networking EXperiments and Technologies (CoNEXT), Los Angeles, CA, USA, December 2024.

[19] P. Bosshart, et al., "P4: Programming protocol-independent packet processors," ACM SIGCOMM Computer Communication Review, vol. 44, no. 3, pp. 87-95, July 2014.

[20] Y. Zheng et al., "Kgent: Kernel extensions large language model agent," in Proc. ACM SIGCOMM Workshop on eBPF and Kernel Extensions, Sydney, Australia, August 2024.

[21] eBPF, Accessed: April 15, 2025.[Online]. Available: https://ebpf.io/

[22] J. Tong et al., “WirelessAgent: Large Language Model Agents for Intelligent Wireless Networks,” arXiv preprint arXiv:2409.07964, September 2024.

[23] J. Wei et al., "Chain of thought prompting elicits reasoning in large language models," in Proc. Advances in Neural Information Processing Systems (NeurIPS), New Orleans, LA, USA, November 2022.

[24] Z. He et al., “Designing Network Algorithms via Large Language Models,” in Proc. ACM Workshop on Hot Topics in Networks (HotNets), Irvine, CA, USA, November 2024.

[25] FCC, Accessed: April 15, 2025.[Online]. Available: https://www.fcc.gov/general/measuring-broadband-america

[26] H. Mao et al., “Neural adaptive video streaming with pensieve,” in Proc. ACM Special Interest Group on Data Communication (SIGCOMM), Los Angeles, CA, USA, August 2017.

약 력

홍준규

2024년 고려대학교 공학사
2024년~현재 고려대학교 석박통합과정
관심분야: 프로그래머블 데이터 평면, 인-네트워크 컴퓨팅, LLM

김희원

2021년 고려대학교 공학사
2021년~현재 고려대학교 석박통합과정
관심분야: 프로그래머블 데이터 평면, 인-네트워크 컴퓨팅, LLM

배찬빈

2022년 고려대학교 공학사
2022년~현재 고려대학교 석박통합과정
관심분야: 프로그래머블 데이터 평면, 인-네트워크 컴퓨팅, 6G, LLM

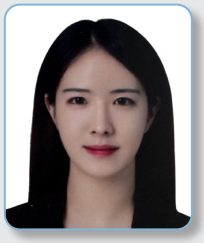
김하은

2024년 고려대학교 공학사
2024년~현재 고려대학교 석사과정
관심분야: 프로그래머블 데이터 평면, 인-네트워크 컴퓨팅, LLM

약 력

이상훈

2024년 아주대학교 공학사
2024년~현재 고려대학교 석사과정
관심분야: 프로그래머블 데이터 평면, 인-네트워크 컴퓨팅, LLM



이정원

2017년 경북대학교 공학사
2024년~현재 고려대학교 석사과정
관심분야: 인-네트워크 컴퓨팅, LLM

구휘모

2025년 동국대학교 공학사
2025년~현재 고려대학교 석사과정
관심분야: 프로그래머블 데이터 평면, 인-네트워크 컴퓨팅, LLM

백상헌

2002년 서울대학교 공학사
2005년 서울대학교 공학박사
2007년~현재 고려대학교 전기전자공학부 교수
관심분야: 네트워크 자동화, AI 기반 네트워크, 네트워크 소프트웨어화, 프로그래머블 네트워크, 5G/6G 모바일 네트워크, 차량네트워크, 이동성 관리