거대언어모델을 활용한 그래프 기반 A* 대중교통 경로 계획

김장환*, 신은호*, 전승재*, 전윤호, 박천음†
국립한밭대학교
{kjh, eunho, seungjae5044}@edu.hanbat.ac.kr, {yhjeon, parkce}@hanbat.ac.kr

LLM-A* Path Planning on Public Transportation Graphs

Janghwan Kim*, Eunho Shin*, Seungjae Jeon*, Yunho Jeon, Cheoneum Park†
Hanbat National University

요 약

본 논문에서는 대중교통 경로 탐색에 적용 가능한 새로운 경로 계획 프레임워크를 제안한다. 기존 A* 알고리즘은 국지적 탐색에는 효율적이나, 전역적 경로 추론에는 한계가 있으며, 최근 제안된 LLM-A* 방식 역시 정적이고 규칙적인 환경에 한정된다는 제약이 있다. 본 연구는 실제 대중교통 노선 데이터를 그래프로 구성하고, 이를 기반으로 LLM이 프롬프트를 통해 중간 경유지(waypoint)를 도출한 뒤, 이를 A* 탐색에 반영하는 하이브리드 방식을 제안한다. 실험 결과, 제안한 프레임워크는 기존 A* 대비 탐색 횟수 측면에서 65% 향상되었으며, 복잡한 대중교통 그래프 환경에서도 높은 실용 가능성을 확인하였다.

1. 서론

대중교통 경로 안내 시스템은 출발지부터 목적지까지의 최적 경로를 제공함으로써 사용자의 이동 편의를 향상시킨다. 그러나 대중교통은 노선, 배차 간격, 시간대별 혼잡도 등 다양한 요인에 따라 경로가 유동적으로 변하며, 이러한 변동성은 기존의 휴리스틱 기반 탐색 알고리즘만으로는 최적의 경로 안내를 제공하는 데 한계가 있다. A* 알고리즘 [1]은 거리 또는 소요 시간 최적화에는 효과적이지만, 국지적 탐색 중심 구조 탓에 대규모 그래프나 실시간 변화가 많은 환경에서 계산 효율성과 확장성에 어려움이 있다. 이를 보완하기 위해 거대언어모델(Large Language Model, LLM)의 전역 추론 능력을 경로 계획에 활용하려는 연구가 활발히 진행되고 있다 [2, 3]. 그러나 선행 연구 대부분은 가상 환경이나 로봇 중심의 연속 공간 탐색에 집중되어 있어, 혼잡도와 불확실성이 큰 실제 대중교통 그래프 적용에는 어려움이 있다.

본 논문에서는 실제 대중교통 노선 데이터를 기반으로 그래프를 구성하고, LLM이 추론한 중간 경유지와 A* 알고리즘을 결합한 하이브리드 경로 계획 프레임워크를 제안한다. 세종특별자치시 대중교통 빅데이터를 이용한 실험 결과, 제안한 LLM-A* 방식은 기존 A* 대비 평균 65%의 연산 및 공간 복잡도를 절감하였다. 또한 Qwen3와 Llama3.2를 대상으로 Few-Shot과 Chain-of-Thought 기법을 적용한 결과, Qwen3 Few-Shot은 연산 효율, Llama3.2 Few-Shot은 메모리 사용량과 방문 엣지 수 측면에서 가장 우수한 성능을 보였다.

2. 관련 연구

LLM 기반 경로 계획 연구 및 한계. [2]는 LLM을 활용해 사용자 요구와 교통 정보를 통합하여 대안 경로를 생성하는 구조를 제안하였으나, 시각적 정보가 없는 경우 경로 품질 저하 문제가 나타났다. [3]은 LLM의 실세계 경로 계획에서 구조화된 정보 없이 수행될 경우 정확도와 일관성에 한계가 있음을 분석하고, 그래프 구조나 추가 알고리즘과의 결합 필요성을 지적하였다.

그래프 기반 LLM 활용 연구. [4]는 그래프를 텍스트로 변환하여 LLM이 처리하도록 하는 Graph Agent 구조를 제안하였으나, 복잡한 공간 제약을 가진 환경에 대한 일반화에는 제한이 있다. [5]는 공간 정보를 그래프로 구조화하고 Q-learning을 접목한 프레임워크를 제안하였으며, prompt 기반 접근 대비 향상된 성능을 보였지만, 여전히 실시간 경로 변경 대응에는 한계가 있다.

LLM과 A* 알고리즘의 결합 시도. [6]은 LLM의 추론과 A* 탐색을 결합한 구조를 제안하였으나, 로봇 상호작용 환경에 초점을 맞춰 대중교통 그래프에는 적용 제약이 있다. [7]은 LLM이 생성한 waypoint를 기반으로 A* 탐색을 수행하는 구조를 제안하였으나, 그리드 기반 시뮬레이션에 한정되어 실제 복잡한 교통 그래프 적용에는 한계가 있다.

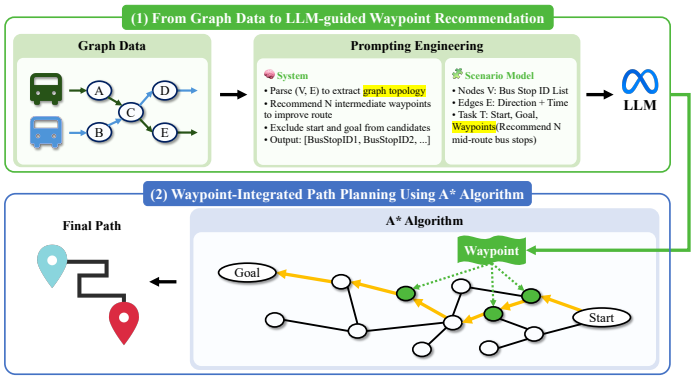

그림 1: 제안한 LLM 기반 경로 계획 프레임워크 구조

*) Equal contribution
†) Corresponding author

3. 제안 방법

본 논문에서 제안한 방법은 [그림 1]과 같으며, 다음의 세 단계로 진행된다: 1) 대중교통 데이터를 기반한 그래프 구성 2) LLM을 이용한 전역적 waypoint 생성 3) waypoint를 참고한 A* 탐색. 제안 방법은 LLM이 제공하는 전역적 탐색 능력과 A* 알고리즘의 국지적 탐색 능력을 동시에 활용해, 방대한 교통 그래프에서도 탐색량을 크게 줄이면서 최적의 경로를 산출한다.

3.1 그래프 데이터 구성

본 논문에서는 대중교통 노선, 정류장 데이터, 버스카드 이용 데이터로부터 하나의 방향성을 가지는 가중 그래프 $G = (V, E)$를 설계하며, 식은 다음과 같다.

$$G = (V, E), V \subseteq \{v_1, v_2, \dots\}, E \subseteq \{(u \to v,\ w_{uv})\} \quad (1)$$

여기서 V는 정류장 집합, E는 노선 순서에 따라 정의된 유향 간선, w는 구간별 평균 이동 시간(분)을 부여하는 가중치이다. 동일 좌표를 공유하는 중복 정류장은 하나의 노드로 병합하고, 구간별 이동 시간이 누락된 경우에는 동일 노선의 평균값으로 보정한다. 구성된 그래프는 노드 수 1995개, 간선 수 3224개로 일반적인 그래프 처리 LLM에 사용되는 그래프에 비해 규모가 크다.

3.2 LLM을 이용한 waypoint 생성

일반적인 LLM은 텍스트를 통해 학습, 평가되기 때문에 그래프가 직접적으로 입력되긴 어렵다 [8, 9]. 본 논문에서는 그래프 데이터를 텍스트 형태의 prompt로 변환하여 LLM에게 입력하며, 그래프의 정보를 노드의 나열과 (엣지, 가중치)의 나열로 서술한다. Prompt는 fewshot과 fewshot+Chain of Thought (CoT) 두 개의 타입으로 진행한다. LLM을 사용하여 크기가 N인 최적의 waypoint 집합 $W = \{w_1, w_2, \dots, w_N\}$을 생성한다. 이때 존재하지 않는 노드가 출력된 경우, 해당 노드를 W에서 제외한다. 생성된 W는 전체 구간에 대한 A* 탐색에 사용된다.

3.3 LLM-A* on Graph

현재 노드를 v_n 이라고 할 때, A* 알고리즘은 식 (2)을 최소화시키는 노드를 우선적으로 탐색한다.

$$f(v_n) = g(v_n) + h(v_n) \quad (2)$$

본 논문은 LLM을 이용하여 생성한 waypoints와 현재 노드 사이의 비용을 $Cost(w_i, v_n)$으로 정의하고 식(2)에 결합하는 방법으로 그래프에서의 LLM-A* 알고리즘을 구현한다.

$$f(v_n) = g(v_n) + h(v_n,) + Cost(w_i, v_n) \quad (3)$$

위 식에서 $g(v_n)$은 시작 노드로부터 현재 노드 v_n까지 도달하는데 소요된 누적 경로 비용이다. $h(v_n)$은 목표 노드까지 휴리스틱 비용이고 $Cost(w_i, v_n)$은 v_n 과 w_i 사이의 위도경도 기반 유클리드 거리이다. A*의 절차에 따라 가장 낮은 $f(v_n)$ 값을 갖는 노드를 선택하고, 이웃 노드들과의 이동 비용을 계산하며 탐색을 수행한다.

4. 실험

본 논문에서는 NVIDIA A6000 GPU 한 장을 이용하여 실험한다. 실험에 사용한 데이터는 세종특별자치시 빅데이터 개방형 플랫폼[1]에서 제공하는 세종시 버스카드 거래 이력 데이터, 노선 정보, 노선별 정류장 현황, 정류장 정보 데이터를 활용한다. LLM 모델로는 Llama-3.2-3B-Instruct[2]와 Qwen3-8B[3]를 사용한다. 실험에서 사용한 노선 1995개의 노드에서 무작위 복원 추출로 151개의 출발지-도착지 쌍으로 선정하였다.

4.1 평가 지표

실험에서 사용한 평가 지표는 알고리즘의 연산 횟수를 평가하는 Operation, 탐색에 사용된 메모리 크기를 평가하는 Storage, 방문한 엣지의 수를 평가하는 Checking Edge와 같으며, 이에 대하여 A* 알고리즘을 기준으로 상대 지수로 환산하였다. 예를 들어 Checking Edge ratio는 알고리즘이 전체 그래프에서 탐색한 간선 집합을 E라고 했을 때, 다음과 같은 식으로 평가된다.

$$\text{Checking Edge ratio} = \frac{|E^{\text{LLM-A*}}_{\text{checked}}|}{|E^{\text{A*}}_{\text{checked}}|}$$

Valid는 LLM이 올바른 waypoint를 제공한 비율이다. LLM이 제공한 waypoint셋 W마다 A*의 경로에 존재하면 True, 그 외는 False로 설정하여 (전체 True 개수) / (전체 데이터 개수) 로 계산한다.

4.2 실험 결과

표 1: 제안 방법에 대한 접근 방법 별 성능 비교

Method	Model	Prompt	Operation ↓	Storage ↓	Check ↓	Valid ↑
A*	-	-	100	100	100	-
LLM-A*	LLAMA3.2	Few-Shot	35.599	**68.23**	**70.725**	72.185
		CoT	38.602	74.253	76.674	43.046
	Qwen3	Few-Shot	**32.024**	71.453	73.385	71.523
		CoT	39.304	77.491	79.574	48.477

[표 1]은 Operation, Storage, Check, Valid를 통해, 본 논문에서 제안한 LLM-A*와 A* 알고리즘을 비교한 결과를 보인다. Opera-

1) https://www2.sejong.go.kr/bigdata/
2) https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct
3) https://huggingface.co/Qwen/Qwen3-8B

tion, Storage, Check는 값이 낮을수록, Valid는 값이 높을수록 우수함을 의미한다. Valid를 제외한 지표들은 A*를 기준으로 100으로 정규화한다. Llama3.2와 Qwen3 두 모델 모두 few-shot 프롬프트에서 Operation을 32–36 점대로 낮춰 A* 대비 64–68%의 연산 절감을 달성했다. Storage와 Check 역시 68–73점 수준으로 27–32%가량 절감되며, Valid는 71–72%로 유지되었다. 반면 CoT 프롬프트를 적용하면 Operation, Storage, Check이 few-shot 프롬프트에 비해 소폭 증가하는 동시에 Valid가 43–48%로 크게 하락했다. 평가지표별로는 Qwen-3 few-shot이 최저 Operation (32.024)으로 가장 높은 연산 효율을 보이고, Llama-3.2 few-shot에서 최저 Storage (68.23)와 최저 Check (70.725)을 달성했다.

4.3 실험 분석

LLM-A*에서 기존 A*에 비해 탐색 효율이 비약적으로 올라간 이유는 LLM이 전역적 경유지를 제공함으로써, A*가 국지적 탐색에 집중하도록 유도했기 때문이다. 10만 토큰 정도 되는 그래프 정보를 LLM이 처리하지 못하여 올바르지 않은 형식 혹은 잘못된 경로의 waypoint들이 출력되기도 했는데 이는 Valid가 낮은 이유로 직결된다. CoT 프롬프트에서는 추론 과정 중 loop, repetition이 발생해 사이클이나 노드가 밀집된 지역에서 쉽게 벗어나지 못하는 현상이 발생했는데, 이 현상이 특정 지역에 강한 attention을 발생시켜 전역적 추론을 방해하고 최종적으로 품질이 낮은 waypoint를 제공하게 된 것으로 분석된다.

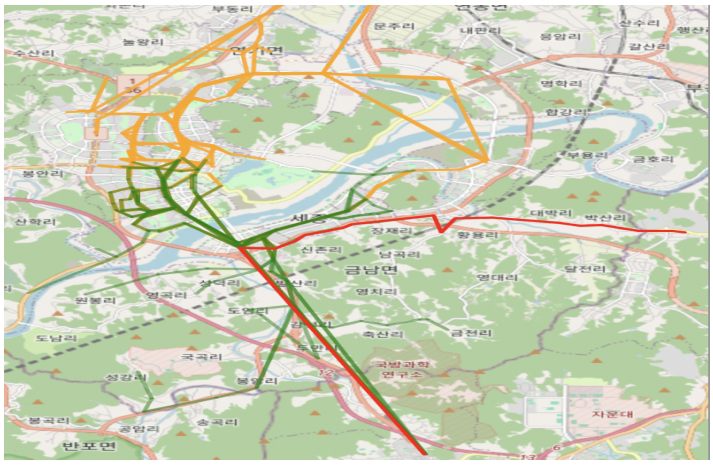

그림 2: LLM-A* 기반 대중교통 경로 탐색 결과

4.4 정성 평가

제안 방법을 실제 대중교통 시스템에 적용한 결과는 [그림 2]와 같다. 여기서 주황선은 A*가 탐색한 경로, 초록선은 LLM-A*가 탐색한 경로, 빨간선은 실제 이동 경로이다. 실험 결과, 실제 이동 경로와 멀리 떨어져 있는 영역도 탐색하는 A* 알고리즘과 달리, LLM-A*는 버스 이동 경로 주변에 탐색 공간이 형성된 것을 볼 수 있다.

5. 결론

본 논문에서는 LLM의 전역적 추론 능력과 A* 알고리즘의 국지적 탐색 효율을 결합한 하이브리드 경로 탐색 프레임워크를 제안하였다. 제안한 프레임워크는 실제 대중교통 노선 데이터를 기반으로 유향 가중 그래프를 구성하고, prompt 기반의 LLM 추론을 통해 중간 경유지를 도출한 후, 경유지를 적용하는 A* 알고리즘으로 전역성과 연산 효율성을 동시에 달성하도록 설계되었다. 실험 결과, 본 프레임워크는 기존 A* 에 비해 평균 약 65%의 연산 및 메모리 사용량을 절감하면서도, 70% 이상의 경로 유효성(Valid)을 유지했다. 실제 이동 경로와 비교하였을 때도 A* 에 비해 더 효과적인 탐색 영역을 생성하는 것을 알 수 있다.

다만, 본 논문은 정적 그래프 기반 환경에서 수행되어 시간대별 혼잡도 같은 동적 요인을 반영하지 못한 한계가 존재한다. 또한 LLM이 그래프를 제대로 이해하지 못하여 waypoint의 신뢰도가 떨어지는 문제도 발생했다. 향후 연구에서는 이러한 한계를 보완하기 위해 동적 그래프를 사용하여 실시간 변화에 대응 가능한 경로 탐색 시스템으로의 확장하고, 더 나은 그래프 이해를 위해 전체 그래프를 여러 서브그래프로 LLM에 입력하는 프레임워크 구성을 계획하고 있다.

참고 문헌

[1] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.

[2] B. Fang, Z. Yang, S. Wang, and X. Di, "Travellm: Could you plan my new public transit route in face of a network disruption?," *arXiv preprint arXiv:2407.14926*, 2024.

[3] W. Chen, M.-W. Su, N. Mehjabin, and M. L. Cummings, "Can llms plan paths in the real world?," *arXiv preprint arXiv:2411.17912*, 2024.

[4] Q. Wang, Z. Gao, and R. Xu, "Graph agent: Explicit reasoning agent for graphs," *arXiv preprint arXiv:2310.16421*, 2023.

[5] H. Deng, H. Zhang, J. Ou, and C. Feng, "Can llm be a good path planner based on prompt engineering? mitigating the hallucination for path planning," *arXiv preprint arXiv:2408.13184*, 2024.

[6] H. Xiao and P. Wang, "Llm a*: Human in the loop large language models enabled a* search for robotics," *arXiv preprint arXiv:2312.01797*, 2023.

[7] S. Meng, Y. Wang, C.-F. Yang, N. Peng, and K.-W. Chang, "Llm-a*: Large language model enhanced incremental heuristic search on path planning," in *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 1087–1102, 2024.

[8] H. Wang, S. Feng, T. He, Z. Tan, X. Han, and Y. Tsvetkov, "Can language models solve graph problems in natural language?," 2024.

[9] Z. Chai, T. Zhang, L. Wu, K. Han, X. Hu, X. Huang, and Y. Yang, "Graphllm: Boosting graph reasoning ability of large language model," 2023.