DMQA Open Seminar

Knowledge Editing: How do LLMs know the President has changed?

2026-02-13

Korea University

Data Mining & Quality Analytics Lab.

심 세 진

발표자 소개

❖ 심세진 (Sejin Sim)

- 고려대학교 산업경영공학과 대학원 재학
- Data Mining & Quality Analytics Lab. (김성범 교수님)
- 석박통합 과정 (2022. 03 ~ Present)

❖ 관심 연구 분야

- Semi-supervised Regression
- Federated Learning
- Machine Unlearning & Knowledge Editing

❖ 연락처

- ssj259@korea.ac.kr

간단한 아이스브레이킹

발표에 앞서!

이번 세미나 Introduction & Related Work의 이미지는

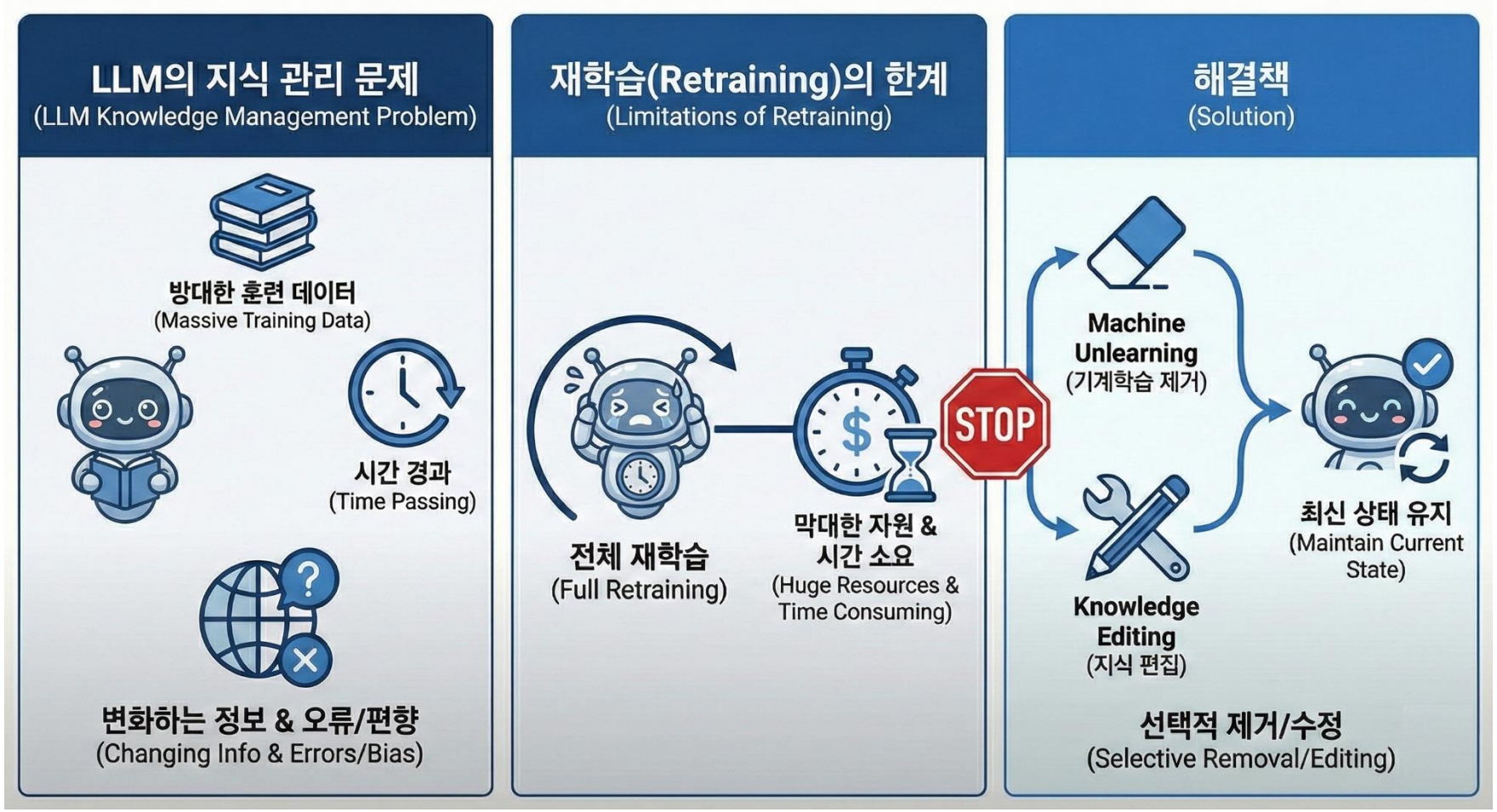
제미나이 나노 바나나의 도움을 받아서 생성했습니다.

퀄리티가 매우 매우 좋으니 시도해 보시길 추천!

Introduction

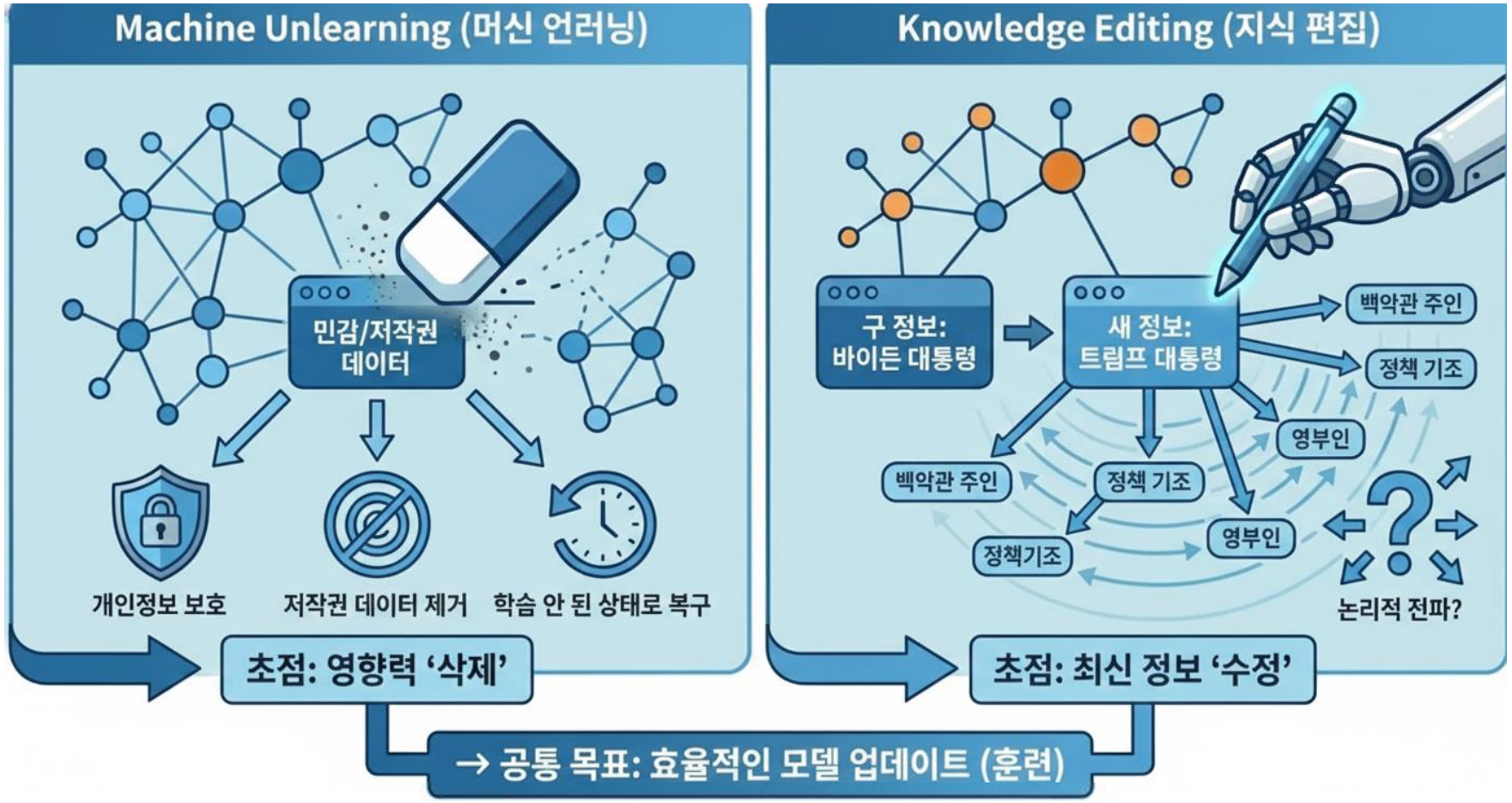
Introduction

❖ Knowledge Editing의 필요성: How do LLMs know the President has changed?



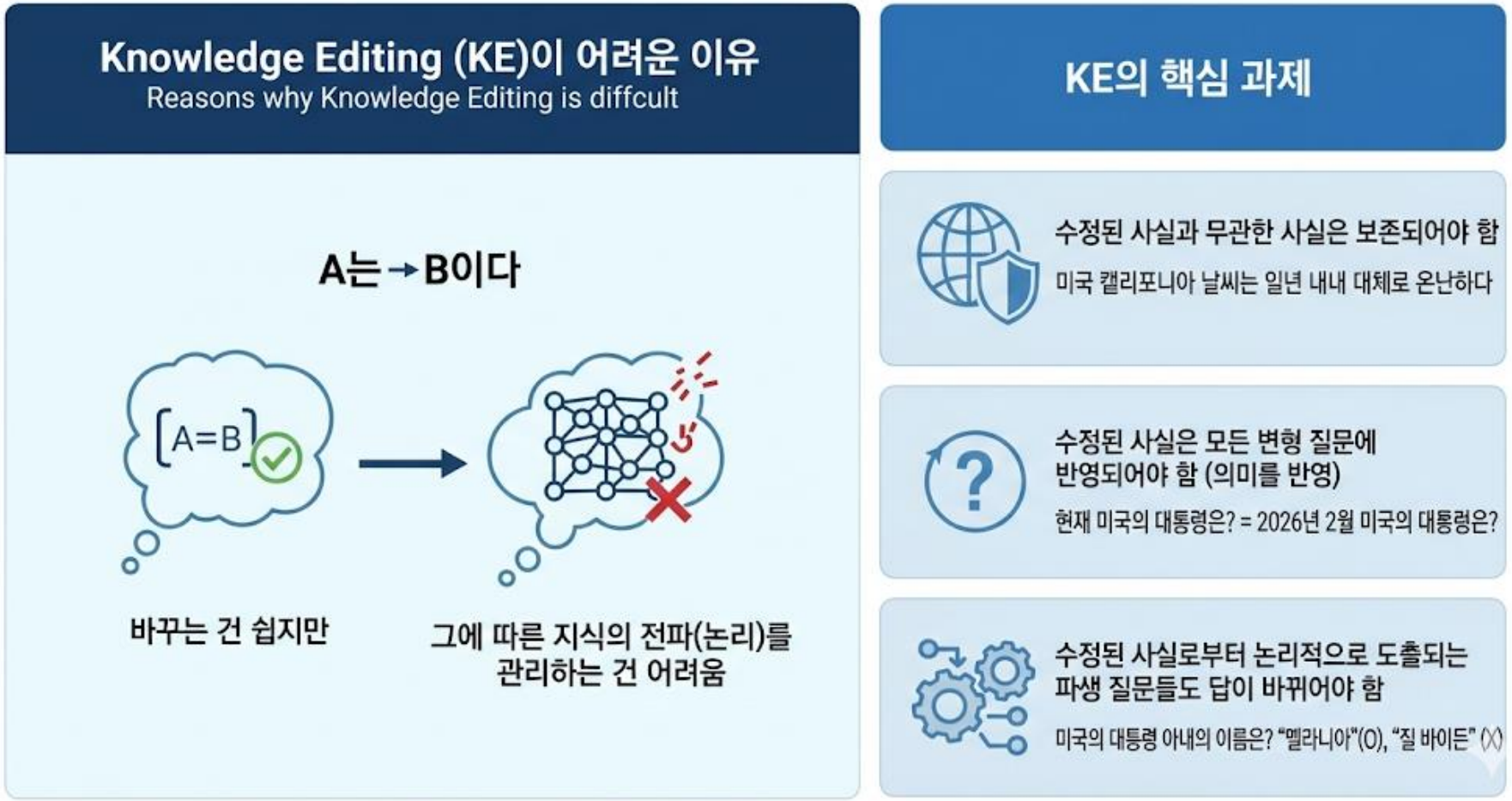
Introduction

❖ Knowledge Editing이란? (언러닝과 뭐가 다를까?)



Introduction

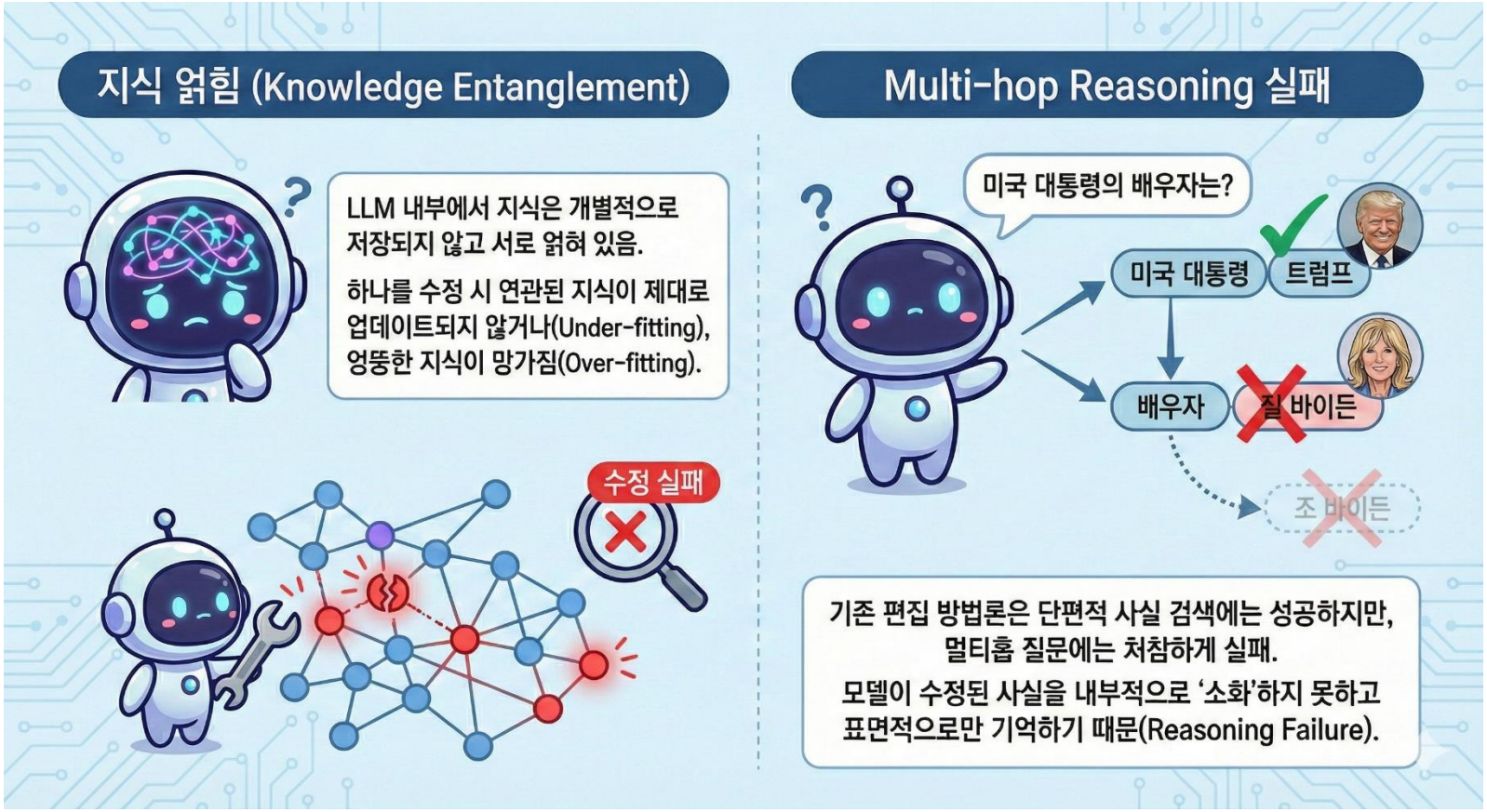
❖ Knowledge Editing(KE)이 어려운 이유



Introduction

❖ Knowledge Entanglement(지식 얽힘) & Multi-hop Reasoning(멀티홉 추론)

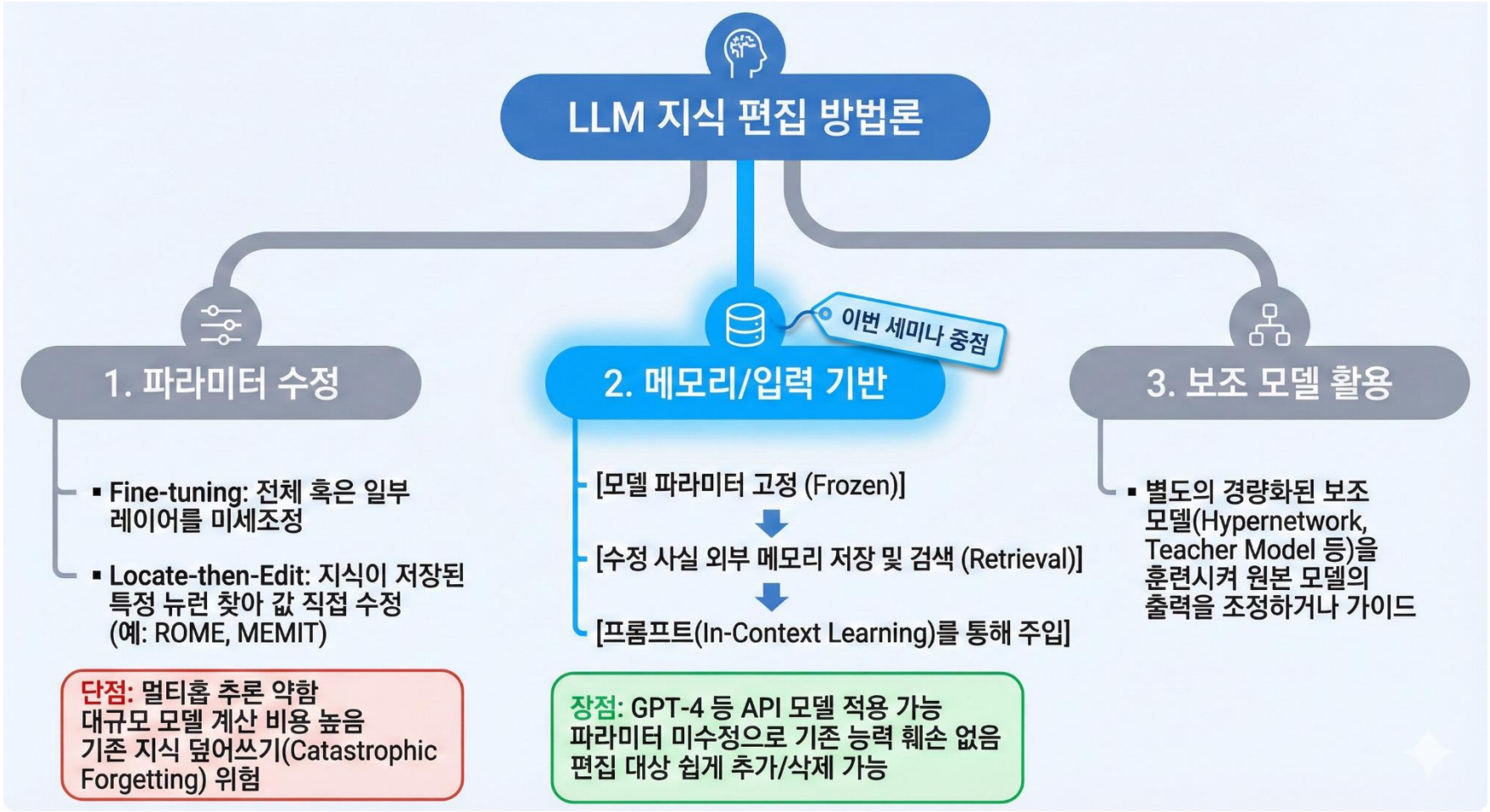
- Multi-hop Reasoning(멀티홉 추론): 여러 개의 유추 단계를 거쳐 결론이나 답을 도출하는 과정



Related Works

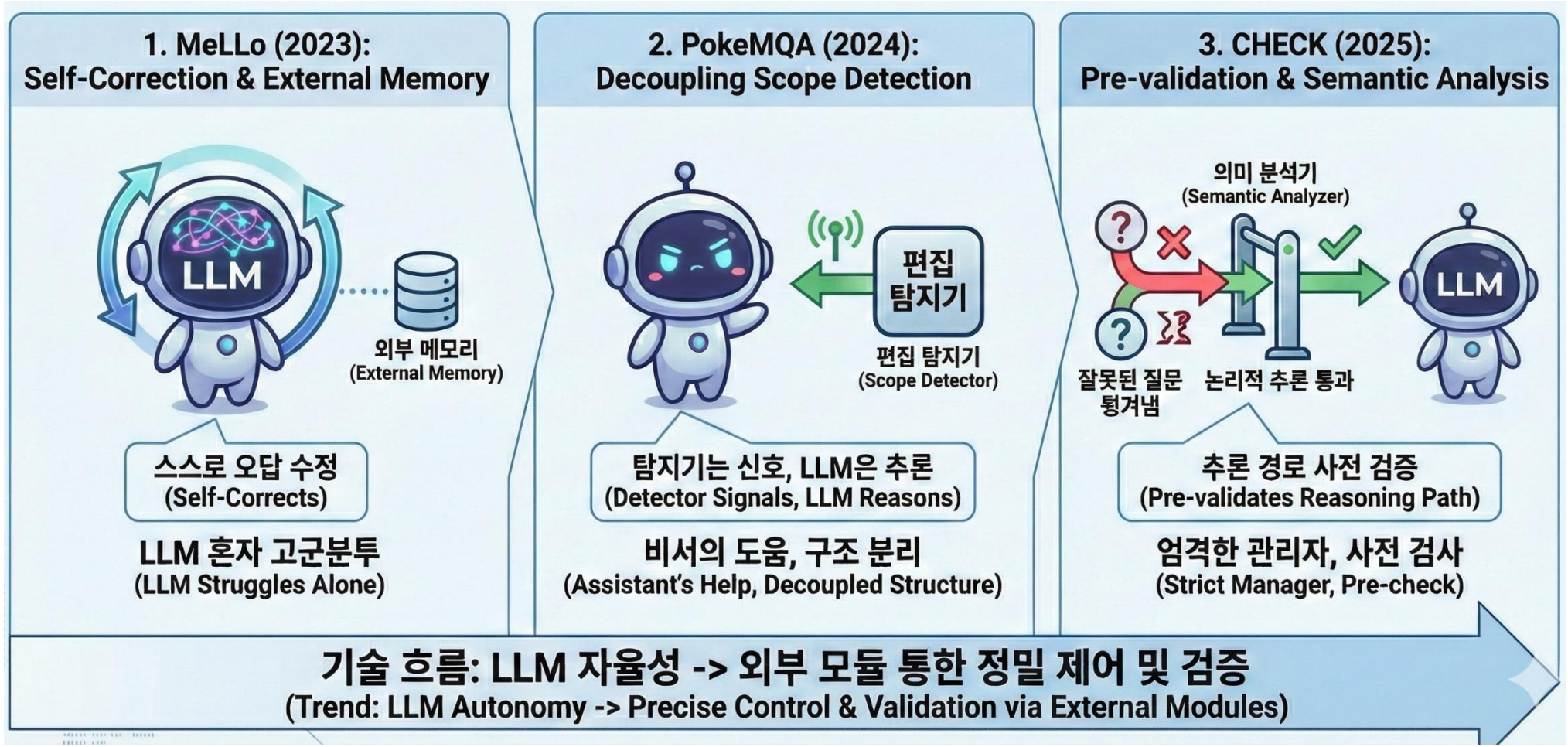
Related Works

❖ 방법론 분류 (Taxonomy)



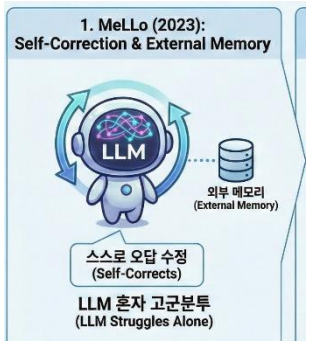
Related Works

❖ 메모리/입력 기반 (총 3가지 소개 예정)



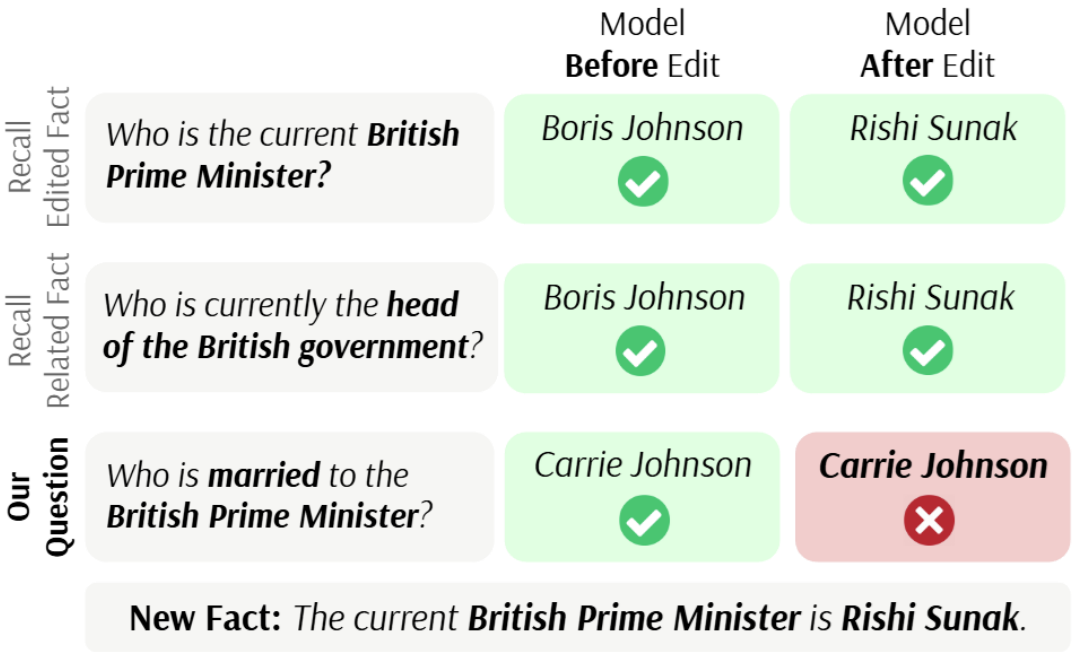
MeLLo
(Memory-based Editing for Large Language Models)

MeLLo (Memory-based Editing for Large Language Models)



❖ 논문 간단 설명

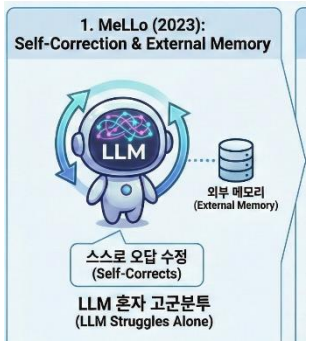
- 기존 방법론들(파라미터 업데이트: ROME, MEMIT)은 단편적 사실에 대해서는 수정이 성공적이지만, Multi-hop(다단계) 추론 질문에는 실패하는 경우가 많음
- Why? 특정 질문에 특정 답이 나오도록 가중치에 하드코딩(Hard-coding)하는 경향 → 연쇄적인 추론 전파 X



기여점

1) 외부 메모리를 활용하는 MeLLo 제안
2) Multi-hop 추론 능력 평가하는 MQuAKE 데이터셋 제안
   ex) Single-hop: 현재 미국 대통령은?
   Multi-hop: 현재 미국 대통령의 배우자는 누구?
   ㄴ 연쇄적 추론 필요

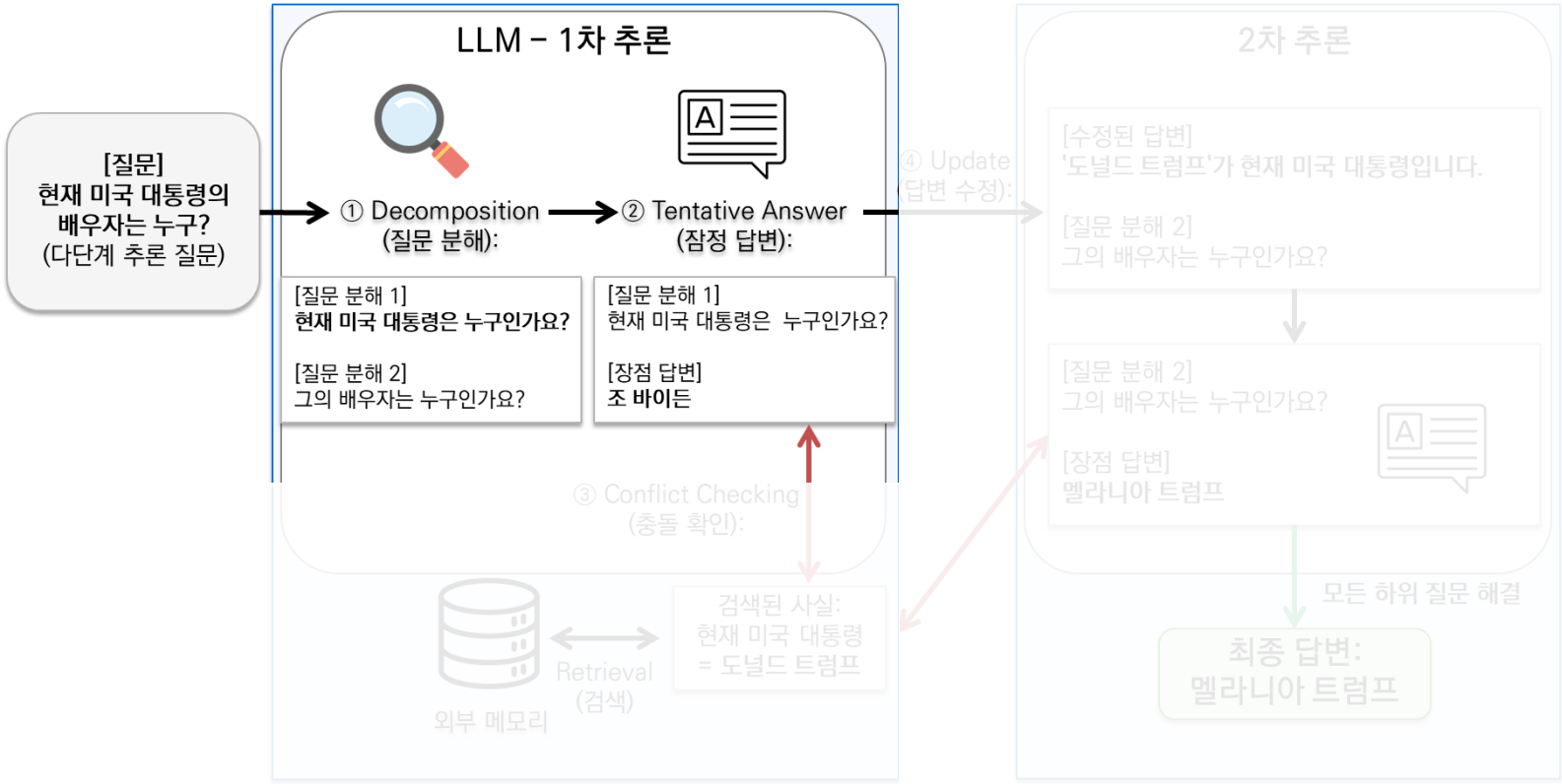
MeLLo (Memory-based Editing for Large Language Models)



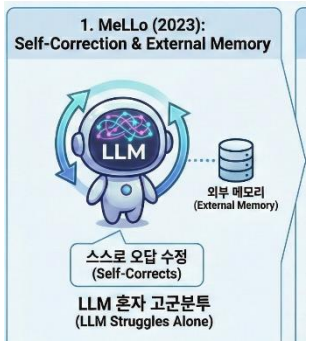
❖ 방법론 구조

① Decomposition (질문 분해): 복잡한 멀티 홉 질문을 LLM이 하위 질문으로 분해

② Tentative Answer (잠정 답변): LLM 내부 지식 답 생성



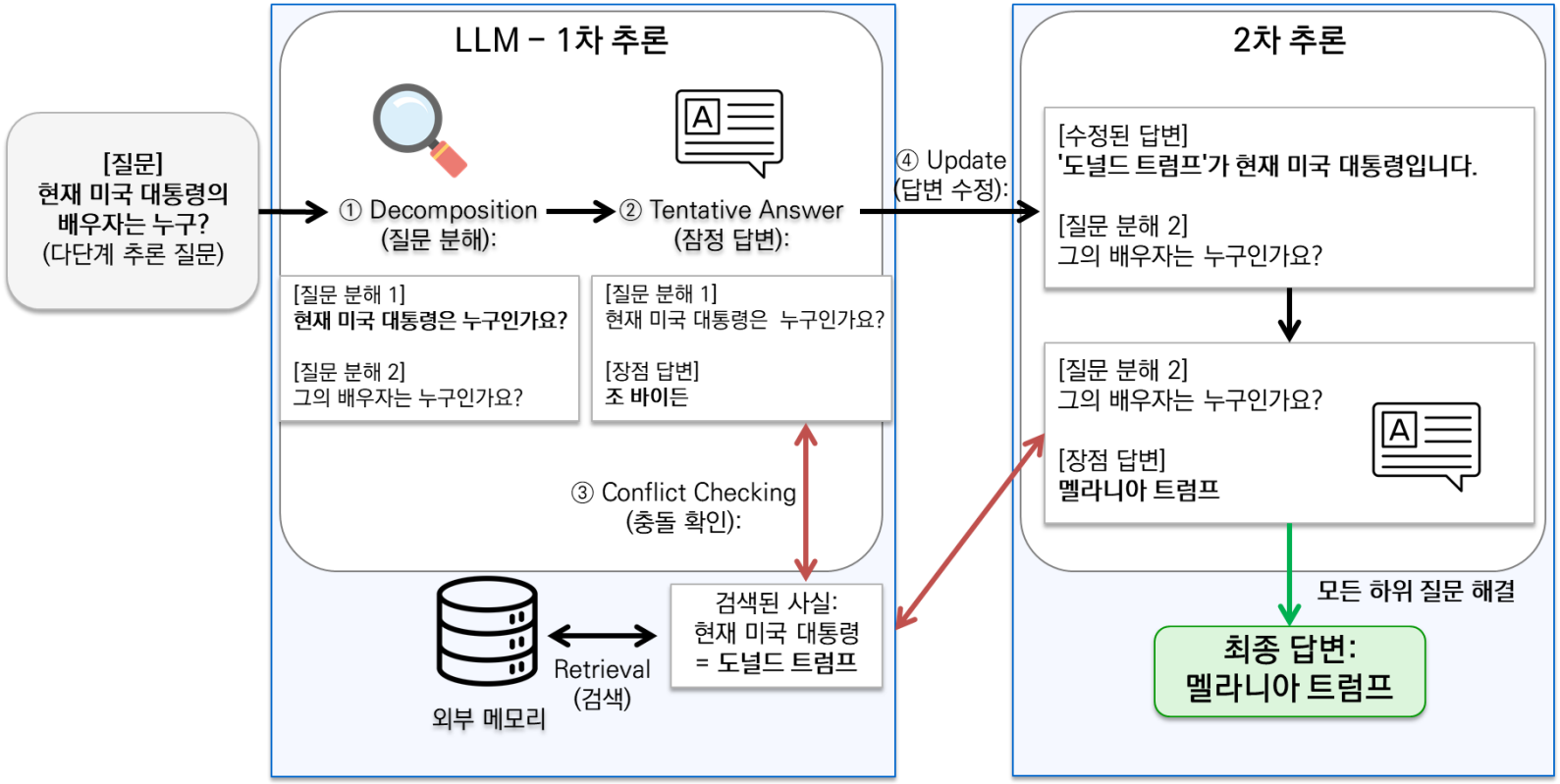
MeLLo (Memory-based Editing for Large Language Models)



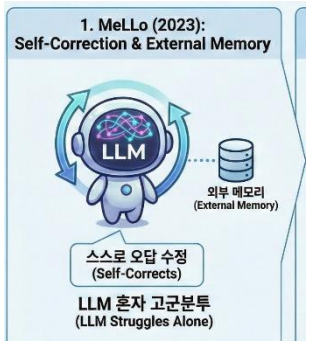
❖ 방법론 구조

③ Conflict Checking (충돌 확인): ②가 외부 메모리에 있는 수정된 사실과 충돌하는지 확인

④ Update (답변 수정): 충돌 시, 메모리 최신 사실로 교체 후 다음 하위 질문으로



MeLLo (Memory-based Editing for Large Language Models)



❖ 장단점 정리

[장점]

1. 기존 파라미터 업데이트 방법론(MEMIT) 보다 높은 성능
2. API 를 사용하는 모델들(ex. GPT 4.0)을 사용할 수 있음 (확장 가능성)

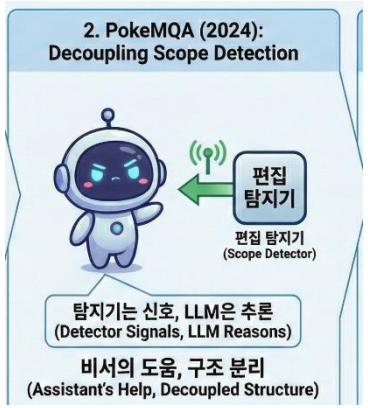
[단점]

1. 질문 분해, 답변 생성, 충돌 감지를 하나의 프롬프트 안에서 LLM 혼자 처리함 → 과부화
2. 불안정한 추론: 충돌 검색 판단하는 데 실패하거나, 논리적 단계를 건너뛰는 문제 발생 가능

→ 추론과 편집을 분리하는 PokeMQA가 등장!

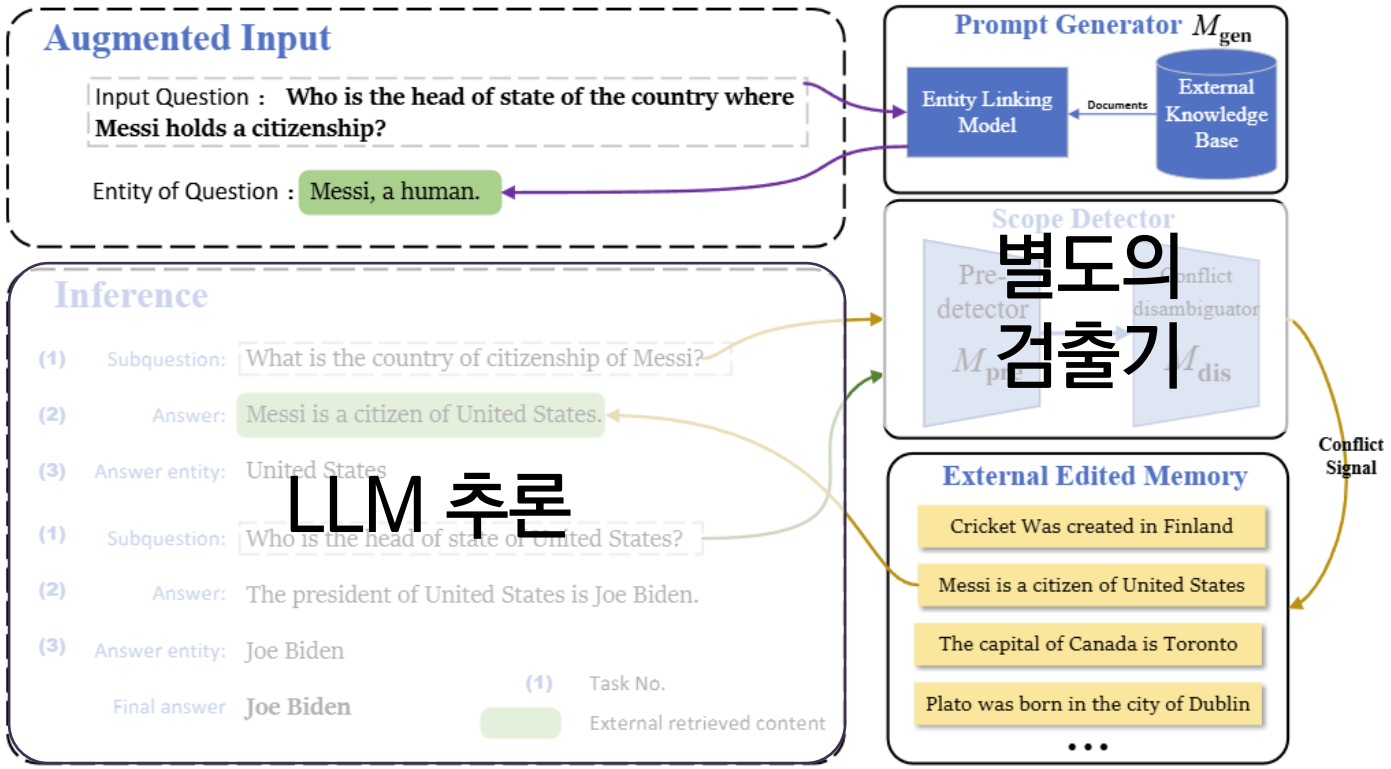
PokeMQA
(Programmable knowledge editing for Multi-hop Question Answering)

PokeMQA (Programmable knowledge editing for Multi-hop Question Answering)

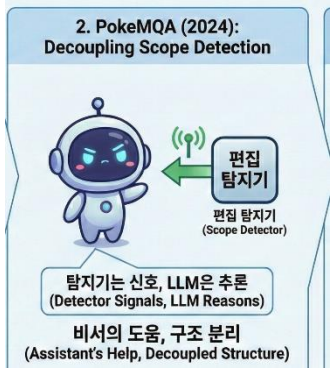


❖ 방법론 구조

- 주요 포인트: LLM은 '추론'에만 집중하고, 편집된 지식의 유효성 판단은 별도의 '검출기'로!



PokeMQA (Programmable knowledge editing for Multi-hop Question Answering)

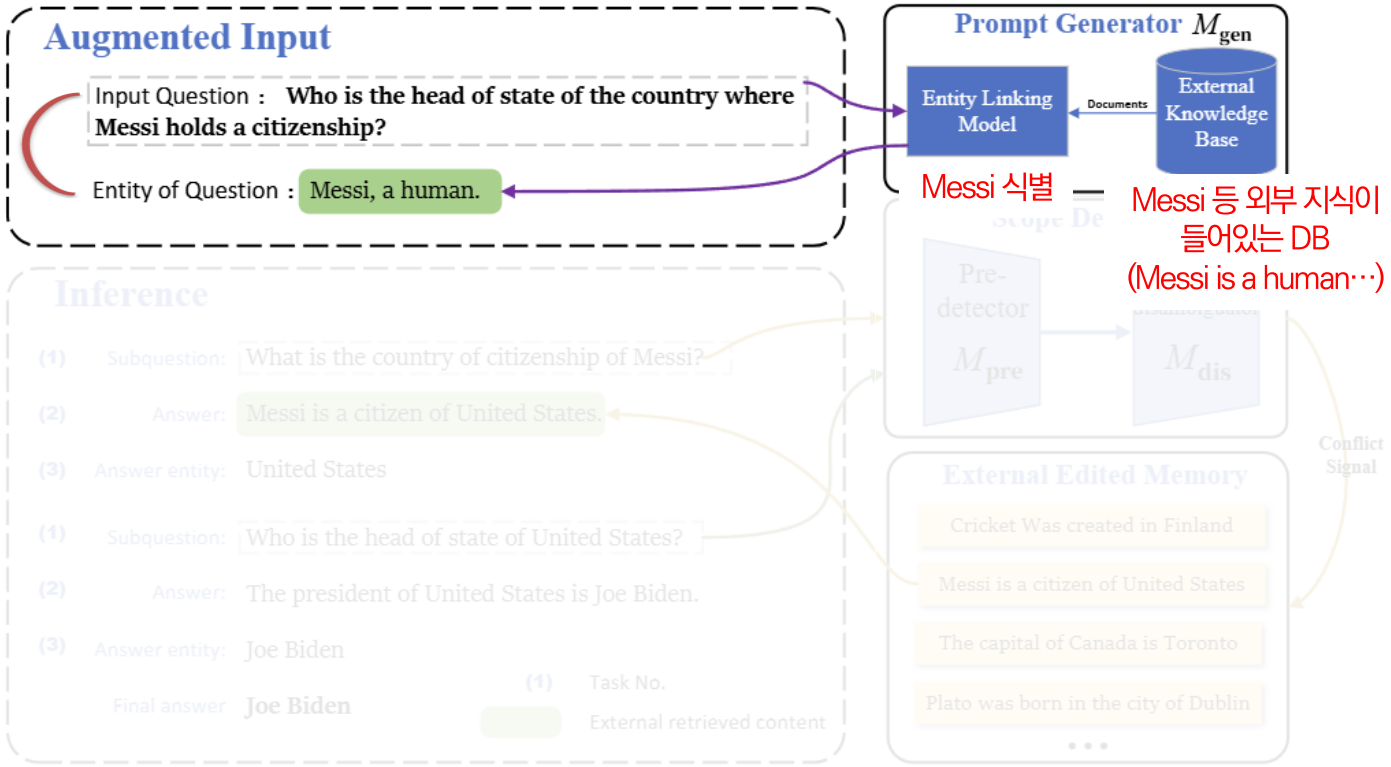


❖ 방법론 구조

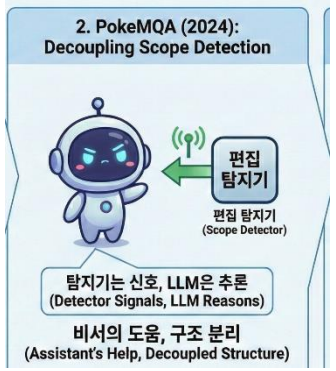
① Knowledge Prompt Generator('질문 분해'의 정확도를 높이기 위함)

: 질문에 포함된 핵심 개체(ex. Messi는 사람) 정보를 외부 DB(WikiData 등)에 검색해서 프롬프트에 입력

결합해서
질문 분해(LLM 추론)에 도움이 되게



PokeMQA (Programmable knowledge editing for Multi-hop Question Answering)



❖ 방법론 구조

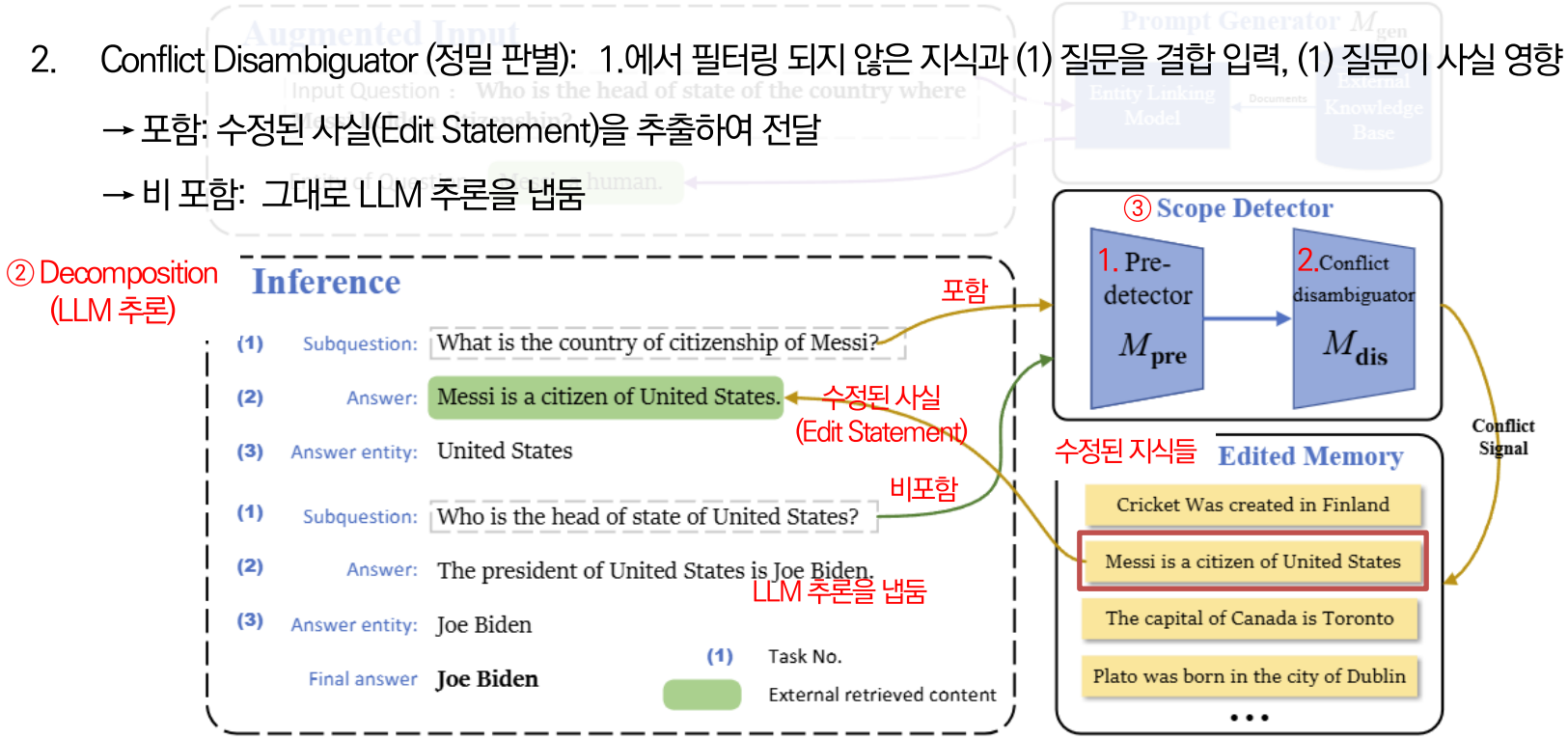
② Decomposition: 멀티 홉 해결하기 위한 하위질문으로 분해

③ Scope Detector(Retrieval): (1) 질문이 수정된 지식들(Edited Memory)과 관련이 있는지 판단

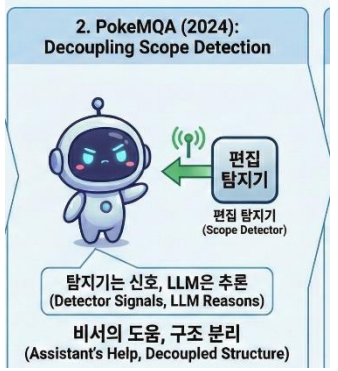
1. Pre-detector (필터링): 1) 질문과 수정된 지식들을 비교하여 <u>의미적으로 거리가 먼 것들을</u> <u>빠르게 필터링</u>
2. Conflict Disambiguator (정밀 판별): 1.에서 필터링 되지 않은 지식과 (1) 질문을 결합 입력, (1) 질문이 사실 영향 범위(Scope, 0~1)에 포함되는지 판단

   → 포함: 수정된 사실(Edit Statement)을 추출하여 전달

   → 비 포함: 그대로 LLM 추론을 냅둠



PokeMQA (Programmable knowledge editing for Multi-hop Question Answering)



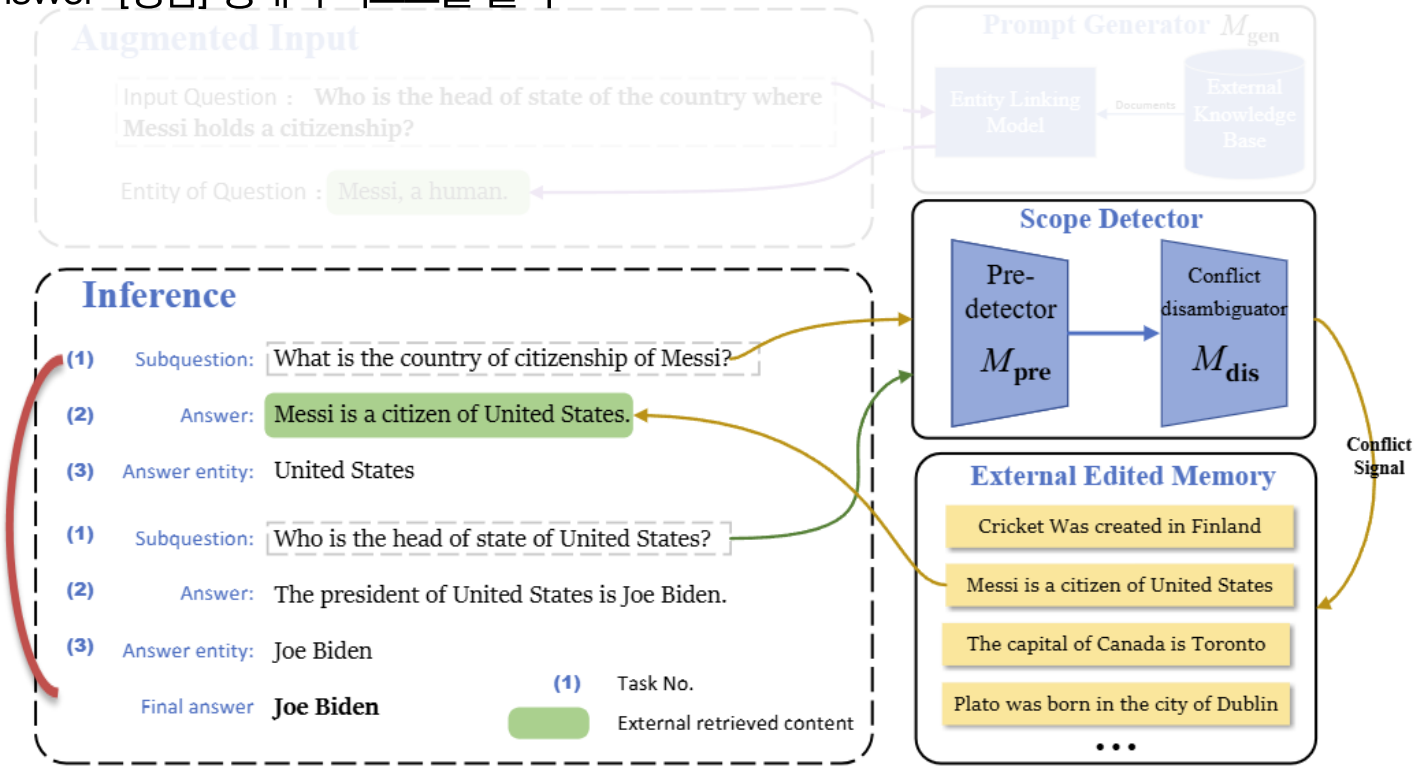
❖ 방법론 구조

- Answer Generation & Iteration: 답변 생성 및 반복

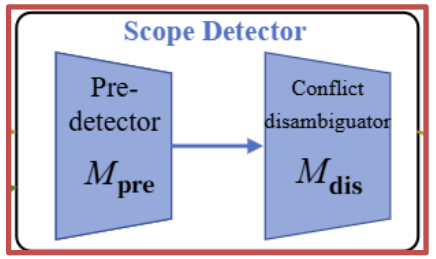
  : LLM이 판단하기에 더 이상 분해할 질문이 없고 원래 질문에 대한 답이 나왔다고 생각되면, 다음 Subquestion 생성 대신

  Final answer: [정답] 형태의 텍스트를 출력

Final answer 까지
답변 생성을 반복

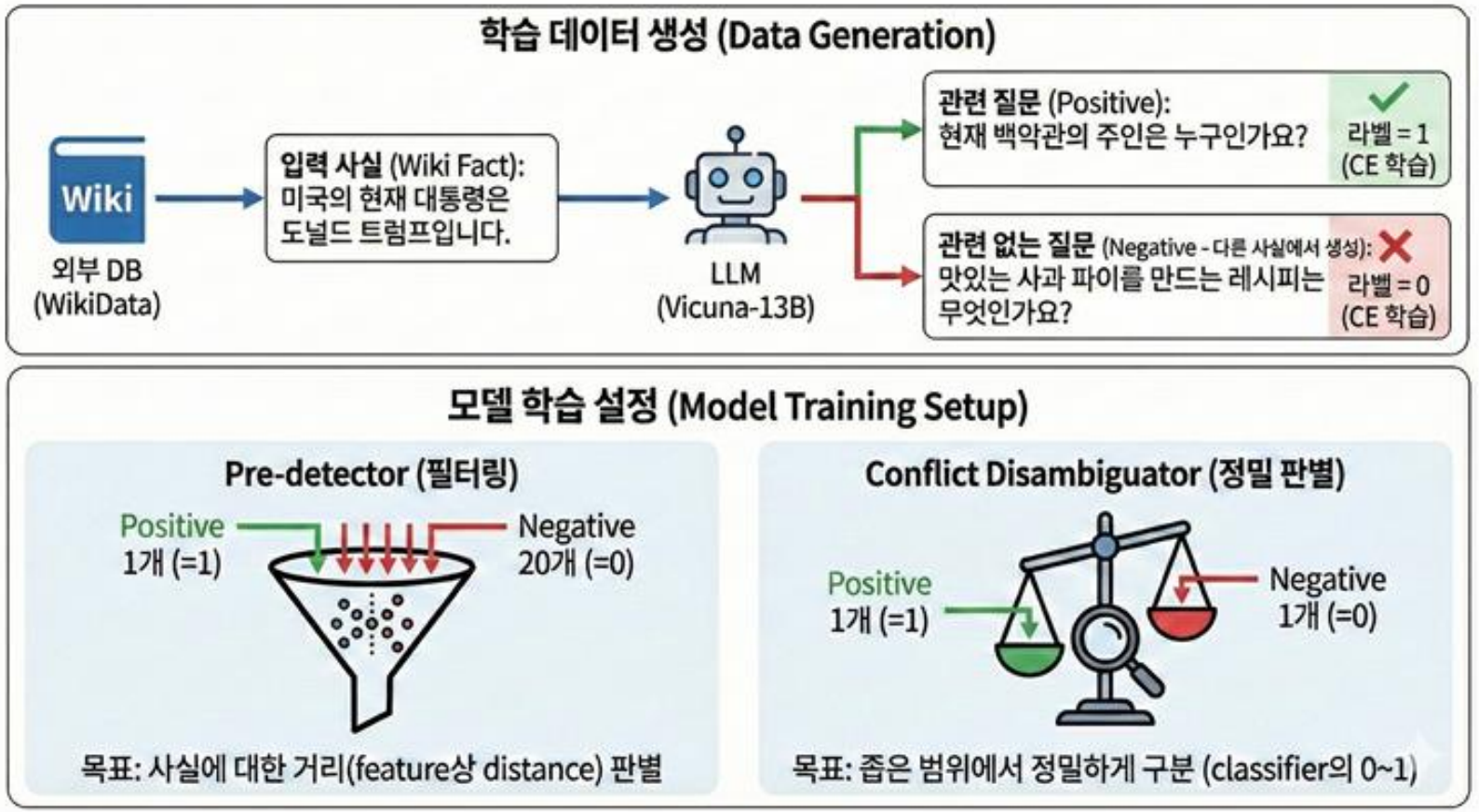


PokeMQA (Programmable knowledge editing for Multi-hop Question Answering)

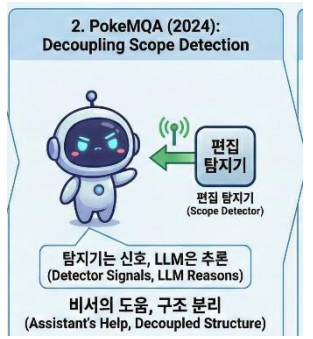


❖ 참고!

- Scope Detector(Retrieval)의 경우 <u>별도 학습 후 사용</u>



PokeMQA (Programmable knowledge editing for Multi-hop Question Answering)



❖ 장단점 정리

[장점]

1. 질문 분해의 정확도를 높이기 위해 외부 DB(WIKI)와 개체 탐지 모델을 활용함
2. 추론을 하는 LLM과 편집을 판단하는 탐지기를 사용하여 추론의 안정성을 높임

[단점]

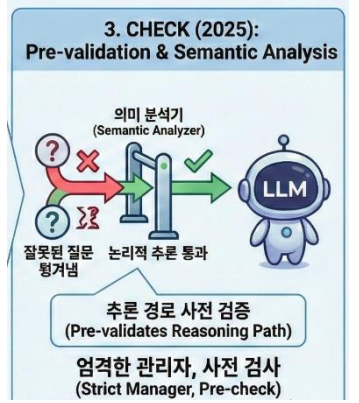
1. 여전히 질문 분해 및 추론의 Final Answer 탐지는 LLM 추론에 맡김
2. LLM은 종종 논리적 순서가 뒤바뀐 질문 분해를 하거나 (예: A의 아내를 찾기 전에 A의 국적을 먼저 묻는 등), 문맥상 말이 안 되는 하위 질문을 생성하는 문제가 발생함

→ LLM 추론을 검증하는 CHECK가 등장!

CHECK

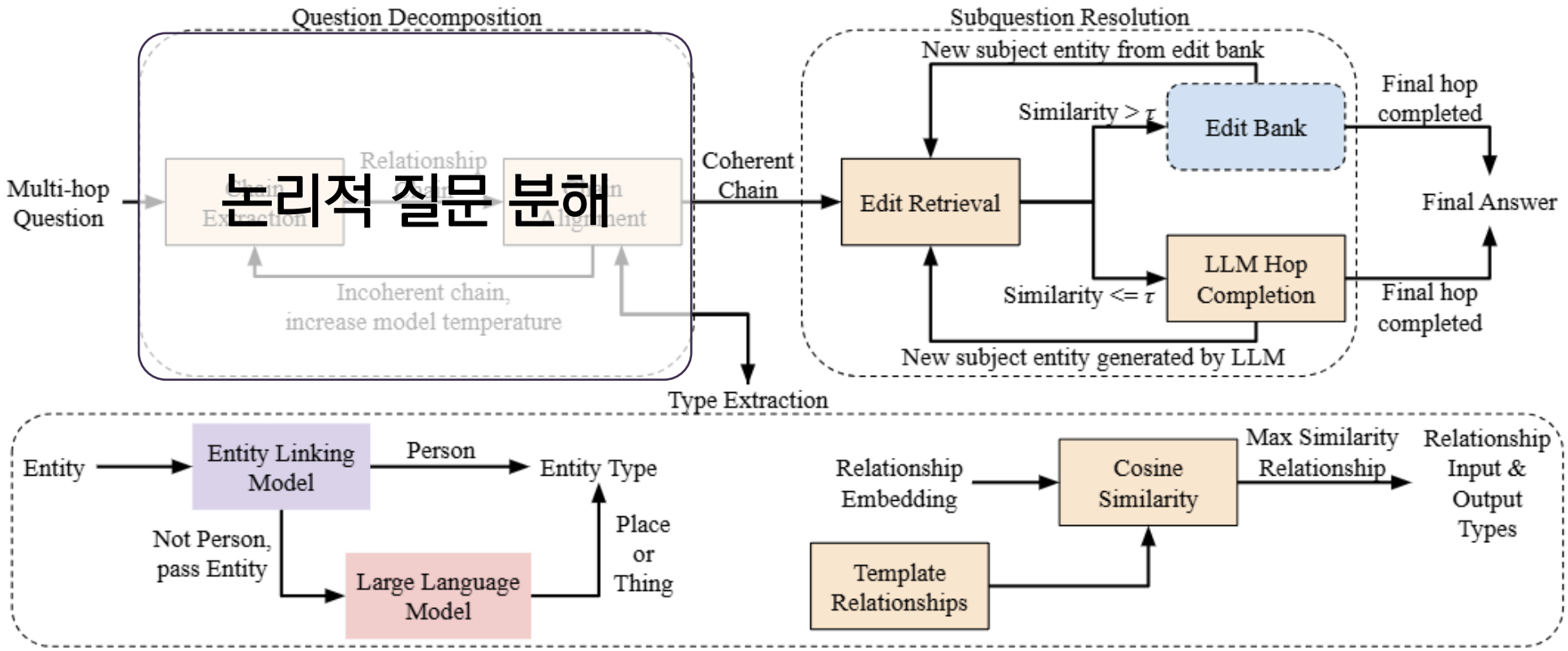
(Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis)

CHECK (Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis)

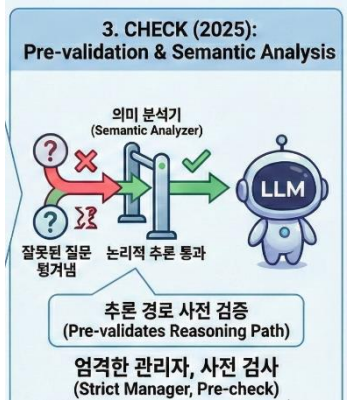


❖ 방법론 구조

- 주요 포인트: Multi-hop의 **비논리적 질문 분해(Decomposition)을 바로 잡자!**



CHECK (Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis)



❖ 방법론 구조

① Question Decomposition – 1) Chain Extraction

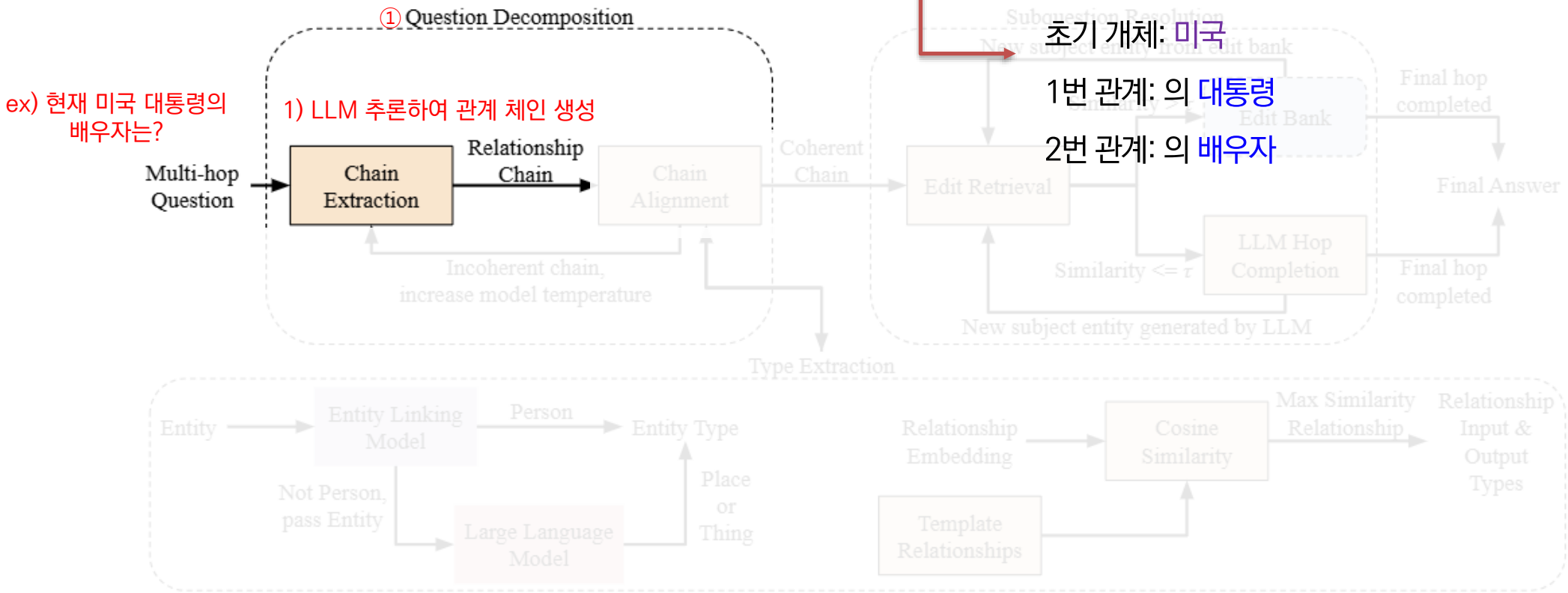
개체와 관계를 추론(LLM) 및 분해: 현재 미국의 대통령의 배우자

ex) 현재 미국 대통령의 배우자는?

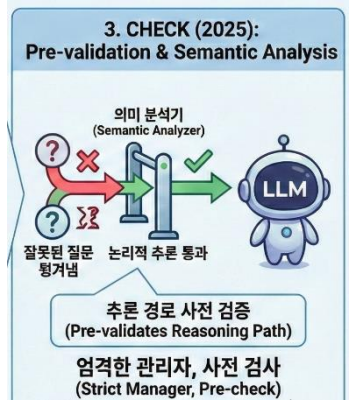
초기 개체: 미국

1번 관계: 의 대통령

2번 관계: 의 배우자



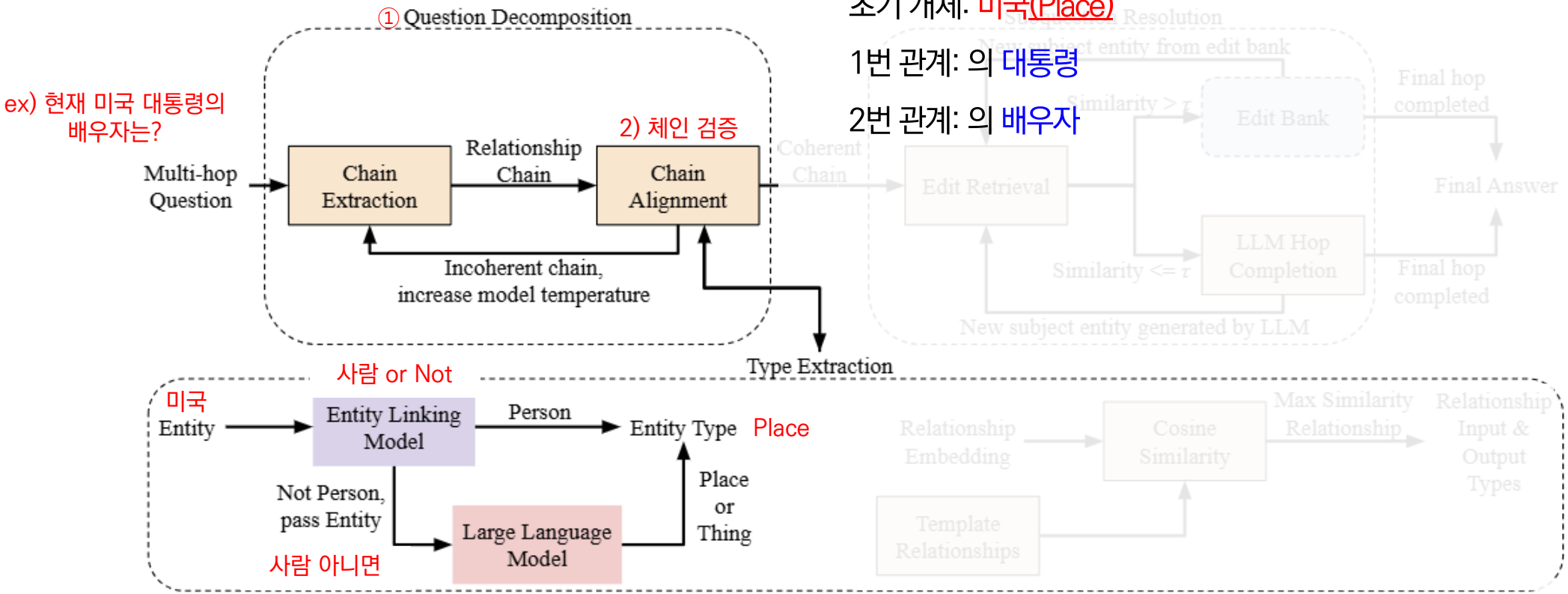
CHECK (Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis)



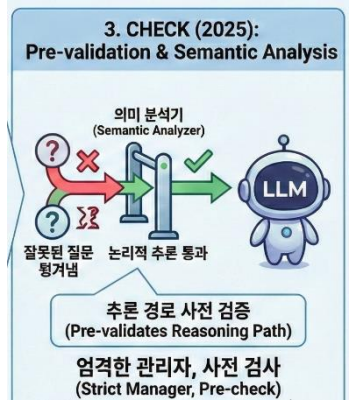
❖ 방법론 구조

① Question Decomposition – 2) Chain Alignment

- ➢ Type Extraction (타입 추출) – Entity Type



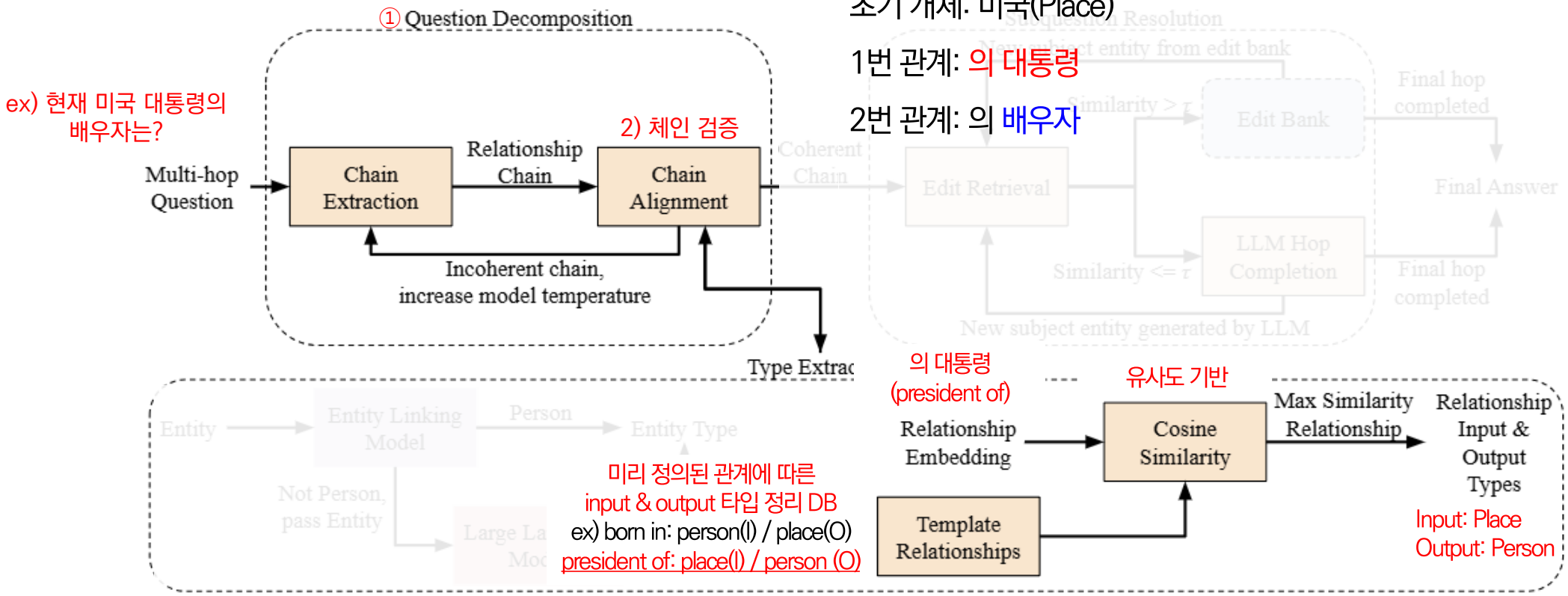
CHECK (Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis)



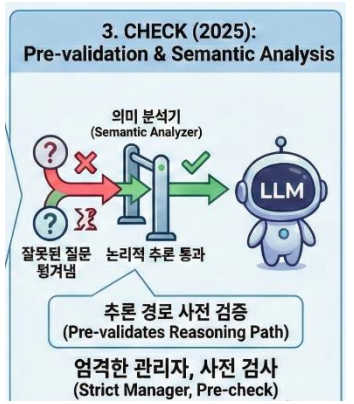
❖ 방법론 구조

① Question Decomposition – 2) Chain Alignment

- ➢ Type Extraction (타입 추출) – Relationship input & output type



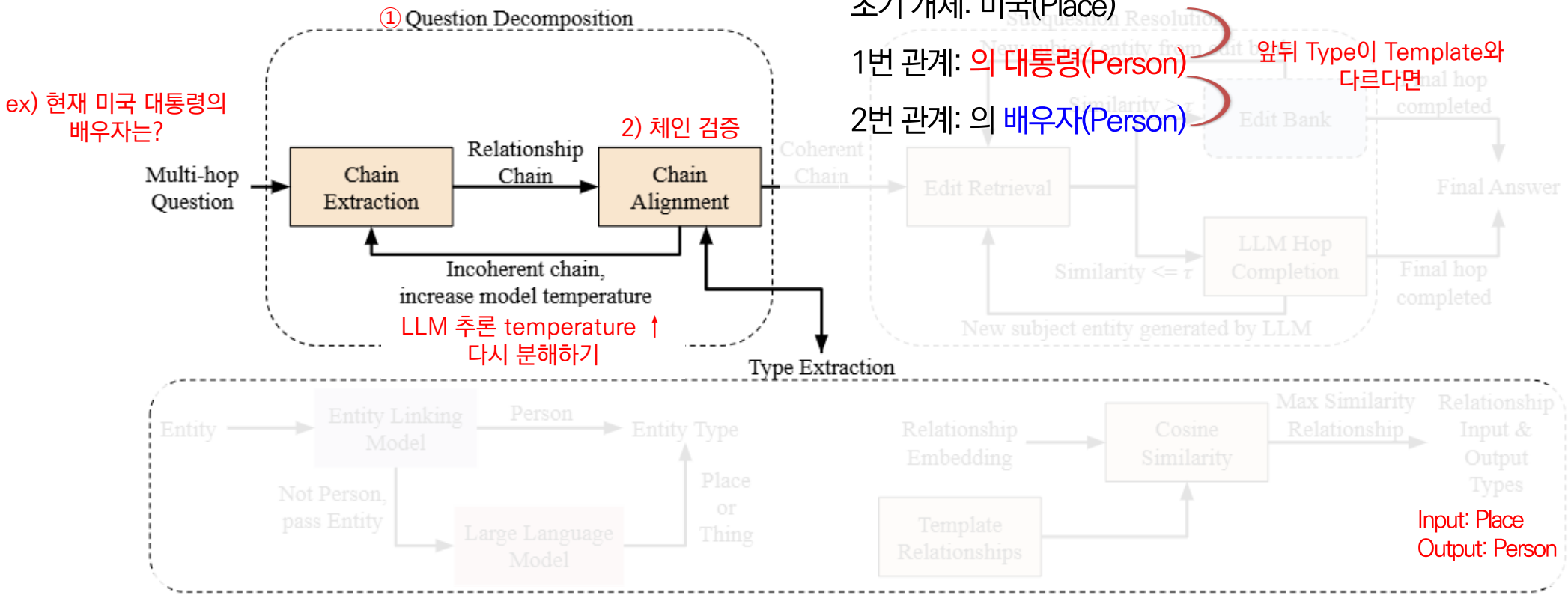
CHECK (Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis)



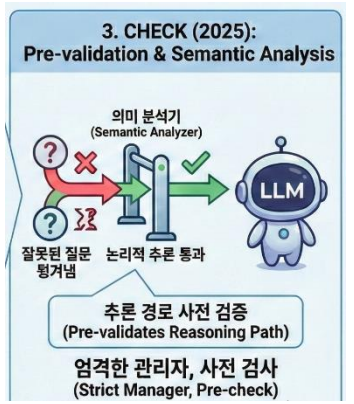
❖ 방법론 구조

① Question Decomposition – 2) Chain Alignment

- ➢ Type Extraction (타입 추출) – Relationship input & output type



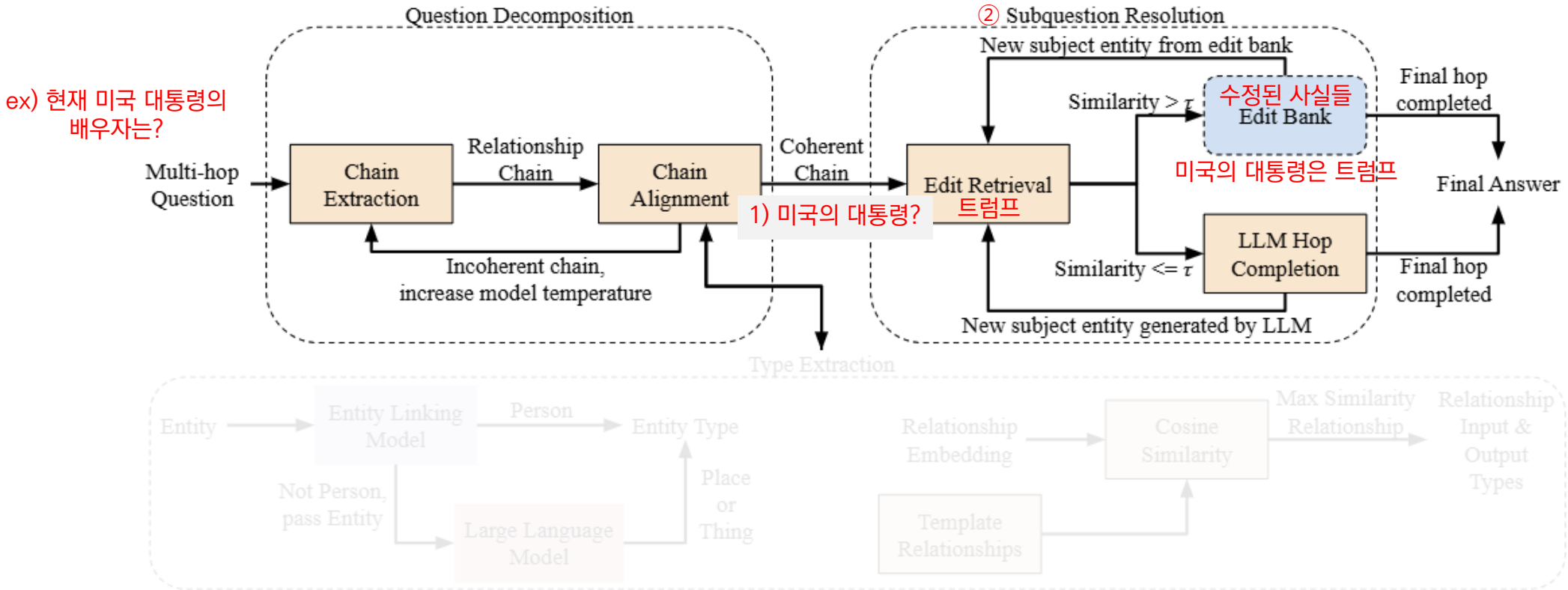
CHECK (Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis)



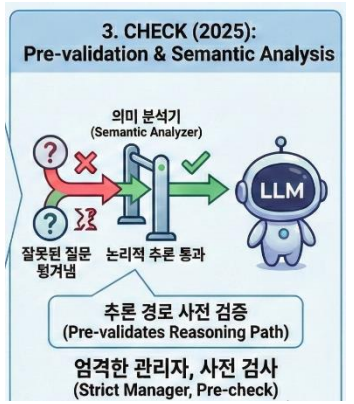
❖ 방법론 구조

② Subquestion Resolution

: 수정된 사실들(Edit Bank)와의 유사도 차이를 기반으로 유사도 높으면 사실 가져오기, 낮으면 LLM 추론 답



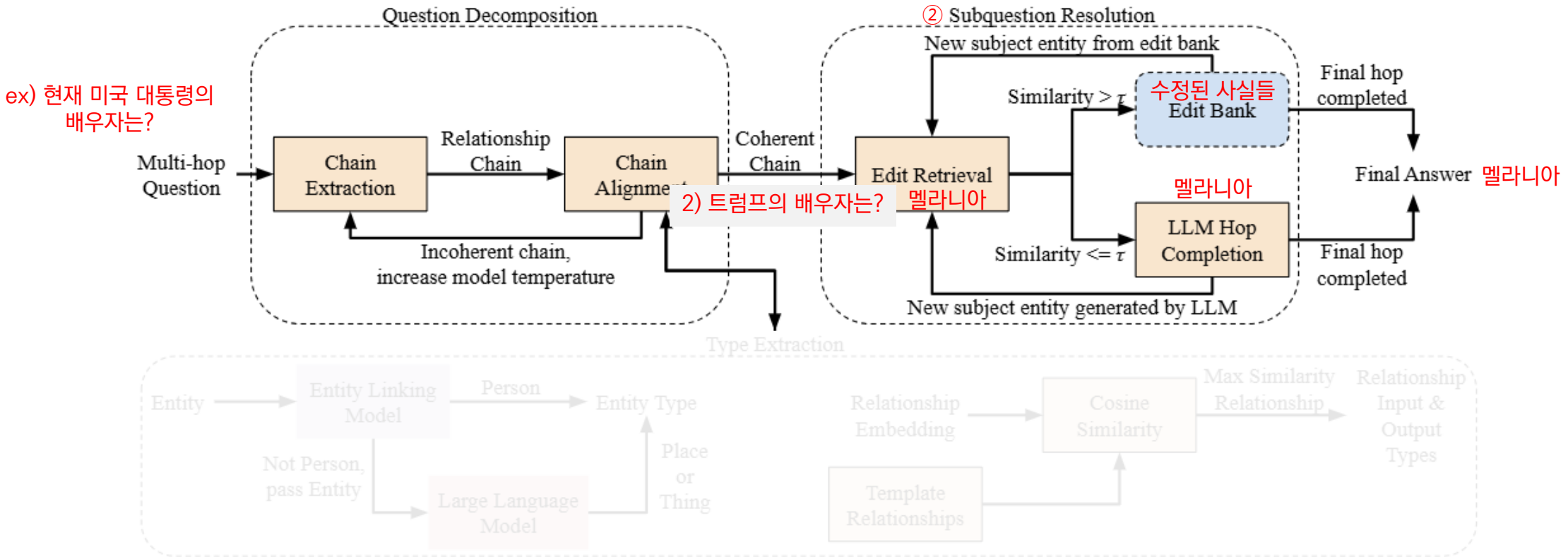
CHECK (Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis)



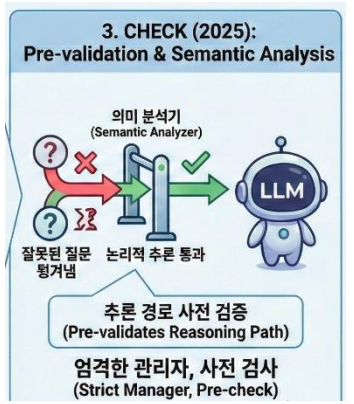
❖ 방법론 구조

② Subquestion Resolution

: 수정된 사실들(Edit Bank)와의 유사도 차이를 기반으로 유사도 높으면 사실 가져오기, 낮으면 LLM 추론 답



CHECK (Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis)



❖ 장단점 정리

② Subquestion Resolution

[장점]

: 수정된 사실들(Edit Bank)와의 유사도 차이를 기반으로 유사도 높으면 사실 가져오기, 낮으면 LLM 추론 답

1. 개체 Type 정의와 관계에 따른 In/output Type 검사 수행을 통해 논리적 연쇄 추론 ↑
2. 복잡한 Multi-hop(분해 시 Single-hop=4개)에서도 SOTA

[단점]

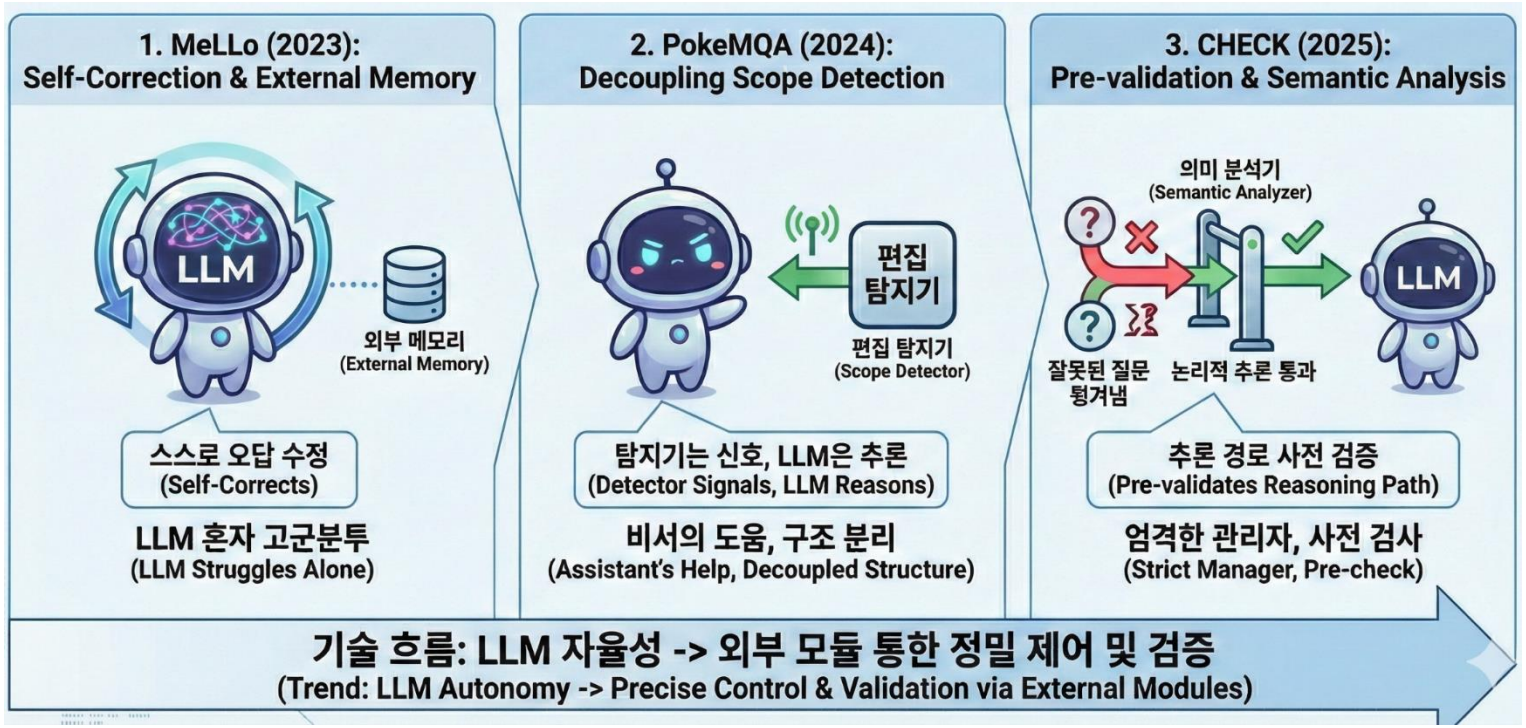
1. Input & output 타입 정리 DB (수동 레이블링 비용)
2. 다른 방법론 대비 실패에 대한 체인 재 분해 추론 시간 ↑ (소요시간에 대한 언급 논문에 x)

Conclusion

Conclusion

❖ 결론 및 느낀 점

- LLM 지식 편집을 위한 방법론 3가지를 소개함 (프롬프트 엔지니어링이 주가 됨)
  - ➢ 변경된 지식을 어떻게 기억시킬까?(MeLLo) → 변경된 지식을 활용하는 추론 과정이 논리적인가?(CHECK) = 초점 변경
- 느낀 점: 현재 유니모달(텍스트)을 넘어 이미지+텍스트를 같이 잊거나(Unlearning), 지식 편집하는 방법론에 대한 궁금증이 생겼음!



References

1. Liu, S., et al. (2025). Rethinking machine unlearning for large language models. Nature Machine Intelligence.
2. Li, Q., et al. (2025). A survey of machine unlearning in large language models: Methods, challenges and future directions. arXiv preprint arXiv:2410.00624.
3. Zhong, Z., Wu, Z., Manning, C. D., Potts, C., & Chen, D. (2023). MQUAKE: Assessing Knowledge Editing in Language Models via Multi-Hop Questions. In EMNLP.
4. Gu, H., Zhou, K., Han, X., Liu, N., Wang, R., & Wang, X. (2024). PokeMQA: Programmable knowledge editing for Multi-hop Question Answering. In ACL.
5. Simon, D., & Ewetz, R. (2025). Knowledge Editing for Multi-Hop Question Answering Using Semantic Analysis. In IJCAI.

Thank you