J. Internet Comput. Serv.
ISSN 1598-0170 (Print) / ISSN 2287-1136 (Online)
http://www.jics.or.kr
Copyright ⓒ 2026 KSII

로컬 LLM 기반 다중 에이전트 지식 추출 프레임워크의 도메인 적응형 확장 및 추출 결과 비교 분석☆

Domain-Adaptive Extension and Comparative Analysis of Extraction Results in a Local LLM-Based Multi-Agent Knowledge Extraction Framework

김 승 겸[1] 유 동 영[2] 노 기 섭[1*]
Seung-gyeom Kim Dong-Young Yoo Giseop Noh

요 약

본 논문은 비정형 학술 문헌으로부터 지식을 추출·통합하는 다중 에이전트 프레임워크 KARMA(Knowledge-Aware Reasoning with Multi-Agents)의 상용 API 의존성과 생의학 중심의 고정 스키마 구조를 개선하는 방법을 제안한다. 이를 위해 Ollama 기반 Llama 3.1을 활용한 로컬 LLM 환경을 구성하고, 입력 도메인에 따라 엔터티 범주와 관계 술어를 동적으로 유도하는 도메인 적응형 구조로 에이전트 프롬프트를 수정하였다. 또한 비정형 응답 처리를 위한 JSON 파싱 및 예외 처리 로직과 의미론적 갈등 해결 방식을 적용하였다. 생의학 및 컴퓨터 과학 문헌을 대상으로 기존 API 기반 KARMA와 비교한 결과, 제안 구조에서는 더 많은 엔터티와 트리플이 생성되었으며 컴퓨터 과학 도메인에서도 스키마 유도가 가능함을 확인하였다. 다만 엔터티 중복과 관계 술어 파편화가 관찰되었으며, 추출 정확도와 일반화 가능성은 향후 추가 검증이 필요하다.

☞ 주제어 : 지식 그래프, 다중 에이전트 시스템, 로컬 거대언어모델, 도메인 적응, 스키마 유도

ABSTRACT

This paper proposes an approach to reduce dependence on commercial APIs and address the biomedical-domain-specific fixed schema of KARMA (Knowledge-Aware Reasoning with Multi-Agents), a multi-agent framework for knowledge extraction and integration from unstructured academic literature. An Ollama-based Llama 3.1 environment implements a local LLM pipeline, while agent prompts dynamically induce entity categories and relation predicates according to the input domain. JSON parsing, exception handling, and semantic conflict resolution are applied to improve structured processing. Comparative experiments on biomedical and computer science literature show that the proposed structure generates more entities and triples than API-based KARMA and can induce schemas for the computer science domain. However, entity duplication and relation predicate fragmentation were observed, and further evaluation is required to assess extraction accuracy and generalizability.

☞ keyword : Knowledge Graph, Multi-Agent System, Local LLM, Domain Adaptation, Schema Induction

1. 서 론

지식 그래프(Knowledge Graph)는 비정형 데이터로부터 구조화된 지식을 표현하고, 개체 간 관계를 기반으로 추론을 수행하기 위한 핵심 기술로 활용되고 있다[1]. 최근 대규모 언어 모델(Large Language Model, LLM)의 발전에 따라 문헌 기반 지식 추출 자동화에 대한 관심이 증가하고 있으며, 이에 따라 다수의 에이전트가 협업하여 지식 그래프를 구축하는 연구가 활발히 진행되고 있다[2, 3]. KARMA (Knowledge-Aware Reasoning with Multi-Agents)는 문서를 반복적으로 분석하고 추출된 지식을 검증하여 지식 그래프에 통합하는 다중 에이전트 기반 프레임워크이다[2]. [2]는

1 Dept. of Software and Communication Engineering, Hongik University, Sejong, 30016, Korea
2 Dept. of Smart Convergence Security Daejin University, 1007, Hoguk-ro, Pocheon-si, Gyeonggi-do, 11159, Korea
* Corresponding author (kafa46@hongik.ac.kr)
[Received 07 July 2026, Reviewed 17 July 2026(R2 07 September 2026, Accepted 13 September 2026]
☆ AI기반 대화형 정부 웹서비스 실시간 행동 탐지 및 자동 방어 시스템 연구(연구개발과제번호 : RS-2026-2552457) 지원으로 수행
☆ 본 연구는 과학기술정보통신부/정보통신산업진흥원 세종 SW 융합클러스터2.0 사업의 지원으로 수행되었음 (과제번호. PJTS080126041007000100030200).

KARMA를 Ingestion, Reader, Summarizer, Entity Extraction, Relationship Extraction, Schema Alignment, Conflict Resolution, Evaluator Agent 등으로 구성하며, 개체 추출부터 관계 검증, 스키마 정렬 및 갈등 해결에 이르는 다단계 검증 절차를 수행함으로써 기존 단일 에이전트 기반 접근 방식의 한계를 보완하고자 하였다.

그러나 기존 연구는 실제 연구 및 대규모 문헌 처리 환경에서 두 가지 한계를 가진다. 첫째, 모든 에이전트가 OpenAI API에 의존하여 동작하기 때문에 문서 수가 증가할수록 토큰 사용 비용이 급격히 증가하며, 민감한 데이터가 외부 서버로 전송되는 보안상의 제약이 존재한다. 둘째, 에이전트 내부의 엔터티 범주와 관계 술어가 생의학(Biomedical) 도메인을 중심으로 설계되어 있어, 다른 도메인에 적용할 경우 도메인에 맞는 관계 정렬과 스키마 구성이 어려워질 수 있다[4]. 특히 일부 에이전트는 생의학 온톨로지를 전제로 한 고정 스키마에 의존하고 있어, 새로운 도메인의 개념 체계를 효과적으로 반영하기 어렵다[5].

본 연구에서는 이러한 인프라적 제약과 도메인 의존성을 해결하기 위해 오픈소스 기반의 로컬 거대언어모델 엔진인 Ollama(Llama 3.1)를 활용하여 온프레미스 지식 추출 환경을 구축한다[6]. 또한 각 에이전트 프롬프트와 내부 구조를 리팩토링하여 입력 문헌의 도메인에 따라 엔터티 범주와 관계 술어를 동적으로 생성하는 스키마 유도 메커니즘을 도입하였다. 이를 통해 생의학 중심의 고정 스키마 구조를 입력 도메인에 따라 엔터티 범주와 관계 술어를 동적으로 구성하는 도메인 적응형 지식 추출 아키텍처로 확장하였다.

본 연구는 생의학 및 컴퓨터 과학 분야의 학술 문헌을 대상으로 기존 API 기반 KARMA와 로컬 LLM 기반 KARMA 간의 교차 대조 실험을 수행한다. 이를 통해 인프라 전환과 도메인 적응형 구조 적용에 따라 나타나는 추출 결과의 변화를 분석하고[7], 엔터티 수, 트리플 수, 관계 유형 수 및 confidence 등 구조화된 출력 특성을 비교한다.

실험 결과를 통해 로컬 LLM 기반 구조에서 엔터티, 트리플 및 관계 유형의 추출 범위가 변화함을 확인하였으며, 동시에 엔터티 중복 및 관계 술어 파편화와 같은 구조적 한계를 관찰하였다.

2. 도메인 적응형 로컬 LLM 기반 KARMA 아키텍처

본 연구에서는 기존 KARMA의 다중 에이전트 기반 지식 추출 흐름을 유지하면서, 상용 API 의존성과 생의학 중심의 고정 스키마 구조를 완화하기 위해 주요 에이전트의 실행 환경과 내부 처리 방식을 수정하였다. 전체 구조는 기존 KARMA의 지식 추출 및 검증 절차, 로컬 LLM의 비정형 출력을 처리하기 위한 구조화 응답 파싱, 입력 도메인에 따른 엔터티 및 관계 스키마 유도, 그리고 도메인 문맥을 반영한 의미론적 갈등 해결로 구성된다. 이하에서는 기존 KARMA의 지식 추출 구조를 먼저 설명한 뒤, 본 연구에서 수정한 각 구성요소를 순차적으로 기술한다.

2.1 KARMA의 다중 에이전트 기반 지식 추출

기존 KARMA 프레임워크는 상용 GPT 모델을 기반으로 고정밀 지식 추출 모델을 지향한다. 입력된 문서는 Ingestion(IA), Reader(RA), Summarizer(SA) Agent를 거쳐 핵심 정보 위주로 응축된 요약본으로 변환된다. 이후 Entity Extraction(EEA), Relationship Extraction(REA), Schema Alignment(SAA) Agent를 거쳐 표준 개체명과 관계 술어가 트리플(Head-Relation-Tail) 형태로 도출된다. 최종적으로 Conflict Resolution(CRA)과 Evaluator Agent(EA)가 개별 검증 소스의 가중치 합산 점수와 이들의 최종 산술 평균 모델을 통해 지식의 무결성을 검증하고, 임계값 이상의 고품질 지식만을 지식 그래프 스토리지(Knowledge Graph Storage)에 통합한다.

2.2 로컬 LLM 구조화 응답 처리

상용 API에서 로컬 LLM(Ollama/Llama.3.1) 환경으로 시스템을 이식하는 과정에서, 로컬 모델의 비정형 출력과 응답 형식 불일치로 인한 파이프라인 오류 문제를 처리하기 위해 코드 수준의 구조화 응답 처리 로직을 수정하였다. 기존 BaseAgent 기반의 통신 구조는 상용 API의 무결한 JSON 출력 및 토큰 사용량 반환을 전제로 작성되었다. 그러나 로컬 LLM은 JSON 구조 전후에 부가적인 자연어 설명을 생성하는 'Chatty Response' 현상을 보이며 이로 인해 JSON 파싱 오류가 빈번하게 발생했다.

이를 해결하기 위해 본 연구에서는 Chatty Response 현상의 발생 양상을 분석하였다. 로컬 LLM이 생성한 응답 중 JSON 파싱에 실패한 원본 응답을 별도의 로그 파일에 기록하는 실험용 파서를 구현하였다. 이로 인해 엔터티 추출 단계에 유효한 구조화 데이터가 전달되지 않거나 후속 파이프라인이 정상적으로 수행되지 않는 문제가 발생한다. 이후 자식 에이전트들의 로직에 정규 표현식(re.sub)

기반의 전처리 과정을 추가하여 순수 구조화된 응답 데이터만 추출할 수 있도록 하였으며, 이를 적용하여 구조화된 응답 데이터를 후속 단계에 전달할 수 있도록 하였다.

표 1에 제시된 사례를 분석한 결과, 로컬 LLM은 JSON 데이터만 반환하도록 지시됨에도 불구하고 출력 전후에 자연어 설명문을 삽입하거나, "PHASE 1: SCHEMA INDUCTION" 및 "PHASE 2: EXTRACTION"과 같은 중간 추론 절차를 함께 생성하는 경향을 보였다. 또한 마크다운 코드 블록과 부가 설명문을 추가하거나 사용자와의 상호작용을 전제로 한 대화형 응답을 생성하는 사례도 다수 관찰하였다. 이러한 비정형 출력은 JSON 파서가 기대하는 구조를 훼손하여 파싱 오류를 발생시키며, 결과적으로 엔터티 추출 파이프라인의 정상적인 데이터 처리를 방해하였다.

이러한 문제는 상용 API와 로컬 LLM 간의 구조화 출력 안정성 차이에서 기인한다. OpenAI의 JSON Mode는 구조화된 출력 형식을 강제하여 구조화된 응답을 안정적으로 생성할 수 있다. 반면, Ollama와 같은 로컬 LLM의 구조화된 출력 기능은 JSON 형식 생성을 유도하는 방식으로 동작하므로, 응답의 형식적 일관성과 스키마 준수 여부가 모델에 따라 달라질 수 있다.

이에 따라 모델이 생성한 응답에 포함될 수 있는 불필요한 설명문, 마크다운 코드 블록, 또는 비정형 텍스트를 제거하고 유효한 JSON 데이터만을 추출하기 위해 후처리 및 필터링 로직을 추가하였다.

그림 1은 Chatty Response 문제를 완화하기 위해 적용한 전처리 및 필터링 로직을 추가한 코드이다. 해당 로직은 먼저 정규 표현식을 이용하여 마크다운 코드 블록(json ...)과 같은 불필요한 출력 요소를 제거한 후, 응답 문자열 내에서 JSON 배열의 시작([ )과 종료( ]) 위치를 탐색하여 유효한 JSON 영역만을 추출한다. 이후 추출된 문자열에 대해 JSON 파싱을 수행하며, 파싱 실패 시 오류를 기록하고 빈 리스트를 반환하도록 구성하였다. 이를 통해 Chatty Response가 포함된 응답에서도 유효한 JSON 영역을 추출하고, 파싱 실패가 발생하더라도 후속 파이프라인이 중단되지 않도록 하였다.

또한, 로컬 LLM의 추론 실패 또는 JSON 파싱 실패로 인해 생성될 수 있는 빈 응답 객체가 후속 처리 과정에서 시스템 예외를 유발하는 문제를 방지하기 위해, 방어적 딕셔너리 접근 기법과 예외 처리 구문을 보완하였다. 이를 통해 비정상 응답이 발생하더라도 예외로 인해 후속 파이프라인 전체가 중단되는 것을 방지하도록 하였다.

(표 1) Chatty Response 일부 예시
(Table 1) Examples of Chatty Response

사례	Chatty Response 일부	의미
1	"Here is the extracted metadata in the required JSON format:"	JSON 앞에 자연어 설명문 추가
2	"As a Knowledge Engineer in the COMPUTER SCIENCE domain, I will identify the core entities..."	역할 수행 과정 출력
3	"PHASE 1: SCHEMA INDUCTION," "PHASE 2: EXTRACTION"	중간 추론 과정 출력
4	"Note that I have excluded entities..."	JSON 뒤에 부가 설명 추가
5	"Please let me know if you'd like me to proceed with further analysis!"	대화형 응답 생성
6	json . . .	Markdown 코드 블록 포함

```
def _parse_json_(self, response: str) -> List[Dict]:
    try:
        clean_response = re.sub(r'```json\s*|\s*```', '',
                                response).strip()
        if "[" in clean_response and "]" in clean_response:
            json_str = clean_response[
                clean_response.find("["):
                clean_response.rfind("]")+1
            ]
            return json.loads(json_str)
    except json.JSONDecodeError as e:
            logger.warning(f"Failed to parse JSON response:
                           {e}. Raw: {response[:100]}...")
            return []
```

(그림 1) 순수 JSON 출력 파서 코드
(Figure 1) Parser code for pure JSON output

2.3 도메인 적응형 스키마 유도

기존 프레임워크의 도메인 고착화 문제는 하드 코딩된 온톨로지 정의에서 기인한다. 그림 2는 기존 Entity Extraction Agent(EEA)의 시스템 프롬프트 내에 DRUG, DISEASE, 등 7개의 생의학 엔터티 범주를 직접 선언하고 있음을 나타낸다. 또한 Relationship Extraction Agent(REA)도 TREATS, CAUSES, ASSOCIATED_WITH 등 생의학 문헌에서 주로 사용하는 관계 술어를 5개의 고 등 생의학 문헌에서 주로 사용하는 관계 술어를 5개의 고정된 관계 유형으로 사용한다.

```
system_prompt = """You are a specialized Entity Extraction Agent for biomedical literature.

ENTITY CATEGORIES:
1. DRUG: Pharmaceuticals, therapeutic compounds, medications
2. DISEASE: Medical conditions, disorders, syndromes, pathologies
3. GENE: Genetic elements, chromosomal regions, genetic variants
4. PROTEIN: Enzymes, receptors, antibodies, protein complexes
5. CHEMICAL: Small molecules, metabolites, ions, biomarker
6. PATHWAY: Biological pathways, signaling cascades, metabolic routes
7. ANATOMY: Organs, tissues, anatomical structures, body regions
```

(그림 2) 고정 엔터티 카테고리 기반 EEA
(Figure 2) Fixed Entity Category-Based EEA

```
self.system_prompt = f"""You are a world-class Knowledge Engineer specializing in the {self.domain.upper()}
domain.

[PHASE 1: SCHEMA INDUCTION]
First, identify the 5 - 7 most standard and academically significant
entity types used in {self.domain.upper()}.

[PHASE 2: EXTRACTION]
Extract entities from the provided text that STRICTLY fit into your induced schema.

[STRICT EXCLUSION RULES]
No Bibliographic Noise: No author names, dates, or journal titles.
No Administrative Info: No Page numbers, “Fig 1,” “Table 1”.
```

(그림 3) 도메인 적응형 EEA
(Figure 3) Domain-Adaptive EEA

```
contradiction_pairs = {
                      (“treats,” “causes”),
                      (“inhibits,” “activates”),
                      (“increases,” “decreases”),
                      (“upregulates,” “downregulates”)
                    }
```

(그림 4) 정적 딕셔너리 기반 CRA
(Figure 4) Static Dictionary-Based CRA

```
if (existing.head.lower() == new_triple.head.lower()
           and existing.tail.lower() == new_triple.tail.lower()):
           prompt = f"""
           ARE these two triples semantically contradictory in the {self.domain} domain?

           Triple 0: {existing.head} -> {existing.relation} -> {existing.tail}
           Triple 1: {new_triple.head} -> {new_triple.relation} -> {new_triple.tail}
           """
           response, _, _, _ = self.make_llm_call(prompt)
           decision = self._parse_json_response(response)
```

(그림 5) 의미론적 갈등 토론 기반 CRA
(Figure 5) Semantic Conflict Resolution CRA

본 연구에서는 이를 해결하기 위해 BaseAgent 초기화 단계에서 도메인 변수를 입력받도록 아키텍처를 리팩토링 하였다. 그림 3의 도메인 적응형 EEA와 같이, 수정한 구조에서는 각 에이전트가 입력된 도메인 값을 기반으로 시스템 프롬프트를 동적으로 구성한다. 특히 EEA와 REA는 고정된 온톨로지 매핑 방식을 제거하고, LLM이 먼저 입력 도메인에서 학술적으로 의미 있는 엔터티 범주와 관계 술어를 스스로 유도한 뒤 추출을 수행하도록 설계하였다. 본 논문은 제안한 시스템을 생의학 중심의 폐쇄형 스키마 구조에서 벗어나, 입력 도메인에 따라 스키마를 동적으로 재구성하는 도메인 적응형 지식 추출 시스템으로 확장하였다.

2.4 의미론적 갈등 해결

또한 기존 KARMA는 CRA가 LLM 기반 토론 프롬프트를 사용하여 신규 트리플과 기존 트리플 간의 모순 여부를 판정하는 구조를 제시한다. 그러나 그림 4와 같이, KARMA의 CRA는 (“treats,” “causes”), (“inhibits,” “activates”) 등 정적 술어 쌍을 기준으로 충돌을 탐지하는 방식에 의존한다. 이러한 방식은 사전에 정의한 생의학 관계에 대해서는 동작할 수 있으나, 새로운 도메인에서 발생하는 관계 술어의 의미론적 충돌을 포착하기 어렵다는 한계를 가진다.

이에 본 연구에서는 CRA의 정적 단어 비교 로직을 제거하고, 그림 5와 같이 LLM이 현재 도메인 문맥 속에서 두 트리플의 의미론적 모순 여부를 직접 판단하도록 변경하였다. 수정한 CRA는 동일한 head-tail 쌍을 가진 트리플이 존재할 경우, 두 관계가 입력 도메인에서 논리적으로 충돌하는지 여부를 LLM에게 질의한다. 이후 LLM은 관계 유형, 신뢰도, 구체성, 정량적 근거, 도메인 지식과의 정합성을 종합적으로 고려하여 충돌 여부와 유지할 트리플을 반환한다. 따라서 본 연구에서는 정적 딕셔너리 기반 충돌 탐지를 제거하고, 도메인 문맥을 기반으로 관계 간 의미론적 충돌을 판단하도록 CRA를 수정하였다.

3. 추출 결과 비교 및 분석

본 논문에서는 제안한 도메인 적응형 로컬 LLM 기반 구조에서 나타나는 지식 추출 결과의 특성을 확인하기 위해 기존 API 기반 KARMA와 비교 실험을 수행하였다. 비교 항목으로는 추출된 엔터티 수, 트리플 수, 고유 관계 유형 수 및 confidence를 사용하였으며, 실험 문헌으로 아스피린 기전을 다룬 생의학 문헌과 트랜스포머 아키텍처를 다룬 컴퓨터 과학 문헌을 활용하였다.

(표 2) KARMA의 추출 통계
(Table 2) KARMA Extraction Statistics

항목	값
엔터티 수	8
트리플 수	4
관계 유형 수	3
평균 신뢰도	0.80
주요 관계	inhibits, decreases, associated_with

(표 3) KARMA 트리플 추출 통계 결과
(Table 3) KARMA Triple Extraction Results

Head	Relation	Tail	Confidence
aspirin	inhibits	COX	0.85
aspirin	decreases	PGE2	0.90
aspirin	associated_with	muscle inflammation	0.75
aspirin	associated_with	sarcopenia	0.70

본 연구에서는 동일 문헌에 대한 두 시스템의 추출 결과를 비교하기 위해 단일 예제 논문(아스피린 기전)을 대상으로 기존 상용 API 기반 KARMA 파이프라인을 재현한다. 실험 결과 총 8개의 엔터티와 4개의 트리플이 생성되었고, 평균 신뢰도가 0.80임을 나타낸다(표 2 참조). 표 3은 aspirin의 COX 억제 및 PGE2 감소와 같은 약리 기전에 집중되어 있음을 제시한다. 이러한 결과는 본 실험 조건에서 기존 KARMA가 상대적으로 제한된 수의 엔터티와 트리플을 생성했음을 보여준다.

두 시스템의 추출 범위를 비교하기 위해 *Coverage Gain*의 개념을 참고하였다. 일반적으로 *Coverage Gain*은 기존 지식 그래프에 존재하지 않았던 신규 지식의 증가를 나타내는 데 사용할 수 있다. 그러나 본 연구에서는 별도의 기준 지식 그래프를 구축하지 않았으므로 엄밀한 의미의 *Coverage Gain*을 직접 산출하지 않았다. 따라서 추출된 엔터티 수, 트리플 수 및 고유 관계 유형 수를 두 시스템의 추출 범위와 구조적 차이를 비교하기 위한 통계적 지표로 활용하였다.

기존 KARMA는 생의학 예제 문헌에서 엔터티 8개, 트리플 4개, 고유 관계 3종을 추출하였다. 이는 본 실험 조건에서 기존 KARMA가 상대적으로 제한된 수의 엔터티와 트리플을 생성했음을 보여준다. 반면 본 연구에서 제시한 로컬 LLM 기반 KARMA를 동일한 생의학 도메인에 적용한 결과, 엔터티 342개와 트리플 283개가 도출되었으며, 컴퓨터 과학 도메인 확장 실험에서는 엔터티 119개와 트리플 131개를 생성하였다. 제안한 로컬 LLM 기반 구조에서는

동일한 생의학 문헌에서 기존 API 기반 KARMA보다 많은 엔터티와 트리플이 생성되었으며, 컴퓨터 과학 문헌에서도 다수의 엔터티와 관계가 추출되었다. 이는 스키마 유도 방식 적용 이후 추출되는 지식의 양과 관계 유형의 범위가 확대되는 경향을 보여준다. 다만 지식 그래프의 완전성 또는 추출 범위와 정확성은 서로 구분되는 품질 차원이며, 더 많은 트리플의 확보가 반드시 높은 정확성을 의미하지는 않는다[8]. 따라서 본 연구에서 관찰된 엔터티 및 트리플 추출량의 증가는 실제 지식 추출 정확도의 향상으로 직접 해석할 수 없으며, 이에 대한 정확성은 별도의 검증이 필요하다.

신뢰도 점수 측면에서도 두 시스템 간 차이가 존재한다. [2]는 수식 1과 같이 confidence $C(t)$, clarity $Cl(t)$, relevance $R(t)$를 각각 산출한 뒤 이들의 평균값이 임계값 Θ이상일 경우 트리플을 통합하는 구조를 제시한다. 반면 실제 배포 코드에서는 수식 2와 같이 confidence와 clarity, relevance에 각각의 가중치를 부여하여 통합 점수를 산출한다. 이에 따라 관계 추출 단계에서 부여하는 confidence 점수는 후속 통합 여부에 가장 큰 영향을 미치는 핵심 지표로 작용한다. 수식 1의 임계값과 수식 2의 가중치는 본 연구에서 새롭게 최적화한 값이 아니라 기존 KARMA에서 공개한 코드의 설정을 적용하였다. 공개한 코드에서는 confidence, clarity, relevance에 각각 0.5, 0.25, 0.25의 가중치를 적용하고 임계값을 0.7로 설정한다. 본 연구에서는 기존 시스템과의 비교 조건을 유지하기 위해 동일한 설정을 사용하였으며, 가중치 및 임계값에 대한 별도의 최적화는 수행하지 않았다.

$$integrate(t) = \begin{cases} 1, & \text{if } \dfrac{C(t) + Cl(t) + R(t)}{3} \geq \Theta \\ 0, & otherwise \end{cases} \quad (1)$$

$$integration\,Score = \begin{array}{l} (0.5 \times Confidence) \\ + (0.25 \times Clarity) \\ + (0.25 \times Relvance) \end{array} \quad (2)$$

기존 KARMA는 관계 추출 단계에서 프롬프트에 정의된 추출 기준을 기반으로 각 트리플의 confidence 점수를 산출한다. 자세한 내용은 [2]를 참조한다. 이 기준에 따르면, associated_with와 같이 단순 연관성을 나타내는 관계는 직접적인 실험 근거나 정량적 수치가 제시되지 않는 경우 중간 수준의 신뢰도 점수(0.5-0.7)를 부여받아야 한다. 그러나 기존 KARMA는 아스피린과 근육 염증 간의 associated_with 트리플에 대해 0.75의 confidence 점수를 부여하였고, 이는 해당 프롬프트에서 단순 연관성 관계에 권장한 점수 범위보다 다소 높은 값이다. 다만 해당 점수 범위는 정답 데이터에 기반하여 최적화된 절대적인 기준이 아니라, 관계의 증거 수준을 LLM의 confidence 생성에 반영하기 위해 프롬프트에 설정한 점수 부여 기준이다.

본 논문은 관계 추출 프롬프트 내에 증거 수준별 confidence scoring 기준을 명시한다. 정량적 근거 또는 통계적 유의성이 있는 관계에 0.9 – 1.0, 직접적인 실험 관찰에 기반한 관계에 0.7 – 0.8, 단순 연관성 또는 배경 이론 수준의 관계에 0.5 – 0.6 범위의 confidence를 부여하도록 명시하였다. 해당 scoring 기준 적용 이후 생의학 및 컴퓨터 과학 실험에서 평균 confidence는 각각 0.77과 0.76으로 나타났다. 다만 본 연구에서는 confidence 점수의 보정이나 반복 실험에 따른 점수 안정성을 별도로 평가하지 않았으므로, 이러한 값만으로 confidence의 일관성 또는 변별력 향상을 단정하지 않는다.

그러나 도메인 적응형 스키마 유도 구조를 적용하는 과정에서 다음과 같은 구조적 한계도 관찰되었다. 원인은 다음과 같다.

첫째, 엔터티 중복 현상이다. 컴퓨터 과학 실험 결과에서 동일 개념인 'transformer-based architectures'와 'transformer'가 서로 다른 노드로 분리되어 그래프에 저장되었다. 이는 중복 제거 메서드가 단순 소문자화 및 공백 제거라는 구문적 일치 매칭에만 의존하기 때문에 발생하는 한계로, LLM의 다양한 동의어 및 변형 표현을 통제하지 못한다는 점을 보여준다.

둘째, 관계 술어의 과잉 파편화이다. 자유도가 높은 스키마 유도 메커니즘을 적용함에 따라 로컬 모델은 'uses', 'utilizes', 'employs'와 같이 실질적으로 유사한 의미를 갖는 술어들을 서로 다른 관계 라벨로 생성하였다. 그 결과 컴퓨터 과학 도메인에서 58종의 독립 관계 라벨이 생성되었으며, 이는 개방형 스키마 유도 방식에서는 다양한 관계 라벨이 생성될 수 있으나, 관계 정규화가 부족할 경우 그래프 스키마의 일관성을 저해할 수 있음을 보여준다. 이에 따라 동의어 사전 기반 정규화와 같은 관계 라벨의 과잉 파편화 완화의 필요성을 확인하였다.

4. 결 론

본 논문은 상용 API 중심의 기존 KARMA 프레임워크를 로컬 LLM 환경으로 전환하고, 생의학 중심의 고정 스키마 구조를 입력 도메인에 따라 동적으로 구성할 수 있는 도메인 적응형 구조로 확장하였다. 이를 위해 Ollama 기반 Llama 3.1 실행 환경을 구성하고, 로컬 LLM의 비정형 응답

을 처리하기 위한 구조화 응답 파싱 및 예외 처리 로직, 도메인 변수 기반 스키마 유도, 그리고 도메인 문맥을 고려한 의미론적 갈등 해결 방식을 적용하였다.

생의학 문헌을 대상으로 기존 API 기반 KARMA와 제안 구조의 추출 결과를 비교한 결과, 기존 시스템에서는 8개의 엔터티와 4개의 트리플이 생성된 반면 제안 구조에서는 342개의 엔터티와 283개의 트리플이 생성되었다. 또한 컴퓨터 과학 문헌에서는 119개의 엔터티와 131개의 트리플이 생성되어, 생의학 중심의 고정 스키마를 사용하지 않고도 입력 도메인에 따른 엔터티 및 관계 유형을 유도하여 지식을 추출할 수 있음을 확인하였다. 이러한 결과는 제안 구조에서 추출되는 엔터티, 트리플 및 관계 유형의 범위가 기존 시스템과 다른 특성을 나타냄을 보여준다. 다만 추출량의 증가는 개별 지식의 정확도 향상을 직접 의미하지 않는다.

구조적 측면에서는 엔터티 중복과 관계 술어의 과잉 파편화도 관찰되었다. 동일하거나 유사한 개념이 서로 다른 엔터티로 생성되거나 의미적으로 유사한 관계가 서로 다른 술어로 분리되는 사례가 확인되었으며, 이는 개방형 스키마 유도 방식에서 엔터티 및 관계 정규화가 필요함을 보여준다. 또한 관계의 증거 수준에 따른 confidence scoring 기준을 적용하였으나, 본 연구에서는 confidence, clarity, relevance의 교차 비교나 confidence 점수 보정을 별도로 수행하지 않았으므로 평가 지표의 일관성이나 변별력 향상을 직접적으로 판단하지 않았다. 본 연구에서는 관계 추출 단계의 confidence를 중심으로 분석하였으며, 최종 지식 통합에 함께 활용되는 clarity와 relevance에 대해서는 두 시스템 간 동일 조건의 교차 비교를 수행하지 않았다. 세 평가 지표와 통합 점수의 교차 분석은 향후 연구에서 추가로 검증할 필요가 있다.

본 연구는 생의학 및 컴퓨터 과학 분야의 제한된 문헌을 대상으로 구조화 출력의 양적·구조적 특성을 비교하였다는 한계를 가진다. 따라서 본 결과를 모든 학문 분야에 일반화하거나 실제 지식 추출 정확도의 향상으로 해석하기에는 추가적인 검증이 필요하다. 향후 연구에서는 정답 데이터 또는 전문가 검증을 기반으로 엔터티와 트리플의 Precision, Recall, F1-score를 평가하고, 복수 문헌과 다양한 도메인을 대상으로 반복 실험을 수행할 예정이다. 또한 confidence, clarity, relevance 및 통합 점수에 대한 교차 비교와 임계값 민감도 분석을 수행하고, 엔터티 정규화와 관계 술어 정규화 기법을 적용하여 도메인 적응형 지식 추출 구조의 정확성과 일반화 가능성을 검증할 필요가 있다.

참고문헌(Reference)

[1] Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S. Yu, “A Survey on Knowledge Graphs: Representation, Acquisition and Applications,” arXiv preprint arXiv:2002.00388, pp. 1-34, 2020. https://doi.org/10.48550/arXiv.2002.00388

[2] Yuxing Lu, Wei Wu, Xukai Zhao, Rui Peng, and Jinzhuo Wang, “KARMA: Leveraging Multi-Agent LLMs for Automated Knowledge Graph Enrichment,” in Proc. of NeurIPS 2025, pp. 1-24, 2025. https://doi.org/10.48550/arXiv.2502.06472

[3] Bowen Zhang and Harold Soh, “Extract, Define, Canonicalize: An LLM-based Framework for Knowledge Graph Construction,” in Proc. of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP), Vol. 2024, No. 1, pp. 9820-9836, 2024. https://doi.org/10.18653/v1/2024.emnlp-main.548

[4] Xiaohan Feng, Xixin Wu, and Helen Meng, “Ontology-grounded Automatic Knowledge Graph Construction by LLM under Wikidata schema,” arXiv preprint arXiv:2412.20942, pp. 1-10, 2024. https://doi.org/10.48550/arXiv.2412.20942

[5] Weidong Bao, Yilin Wang, Ruyu Gao, Fangling Leng, Yubin Bao, and Ge Yu, “DIAL-KG: Schema-Free Incremental Knowledge Graph Construction via Dynamic Schema Induction and Evolution-Intent Assessment,” arXiv preprint arXiv:2603.20059, pp. 1-18, 2026. https://doi.org/10.48550/arXiv.2603.20059

[6] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, et al., “The Llama 3 Herd of Models,” arXiv preprint arXiv:2407.21783, pp. 1-92, 2024. https://doi.org/10.48550/arXiv.2407.21783

[7] Jialin Yang, Dongfu Jiang, Lipeng He, et al., “StructEval: Benchmarking LLMs’ Capabilities to Generate Structural Outputs,” arXiv preprint arXiv:2505.20139, pp. 1-20, 2025. https://doi.org/10.48550/arXiv.2505.20139

[8] Bingcong Xue and Lei Zou, “Knowledge Graph Quality Management: A Comprehensive Survey,” IEEE Transactions on Knowledge and Data Engineering, Vol. 35, No. 5, pp. 4969-4988, 2023. https://doi.org/10.1109/TKDE.2022.3150080

◐ 저 자 소 개 ◑



김 승 겸(Seung-gyeom Kim)
2022년～현재 홍익대학교 소프트웨어융합학과(학사)
관심분야 : 지식 그래프, 정보 추출, 대규모 언어 모델
E-mail : ksg072896@naver.com



유 동 영(Dong-Young Yoo)
2011년 고려대학교 컴퓨터학과(박사)
2014년～2018년 숭실대학교 정보과학대학원 겸임교수
2000년～2019년 한국인터넷진흥원 단장
2021년～2025년 홍익대학교 소프트웨어융합학과 부교수
2026년～현재 대진대학교 스마트융합보안학과 부교수
관심분야 : 자연어처리, 신경망 이론, 강화학습
E-mail : ydy7@daejin.ac.kr



노 기 섭(Giseop Noh)
1998년 공군사관학교 산업공학과(공학사)
2009년 콜로라도대학교 대학원 컴퓨터학과(공학석사)
2014년 서울대학교 대학원 컴퓨터공학과(공학박사)
2016. 12～2018. 02 공군사관학교 전산학과 조교수
2018. 03～2025. 02 청주대학교 인공지능SW학과 조교수
2025. 03～현재 홍익대학교 SW융합학과 조교수
관심분야 : 자연어처리, 신경망 이론, 강화학습
E-mail : kafa46@hongik.ac.kr