추출, 정의, 정규화: 지식 그래프 구축을 위한 대규모 언어 모델 기반 프레임워크

Bowen Zhang[1] **및 Harold Soh**[1,2]

[1] 싱가포르 국립대학교 컴퓨터과학부, [2] NUS 스마트 시스템 연구소

{bowenzhang, harold}@comp.nus.edu.sg

초록

본 연구에서는 입력 텍스트로부터 지식 그래프 생성(KGC)을 위한 자동화 방법에 주목한다. 대규모 언어 모델(LLM)의 발전은 제로/소량 학습 프롬프팅 등을 통해 이를 KGC에 적용하는 일련의 최근 연구를 촉발시켰다. 소규모 도메인 특화 데이터셋에서는 성과를 거두었으나, 이러한 모델들은 실제 응용 분야에서 흔히 접하는 텍스트로 확장하는 데 어려움을 겪고 있다. 주요 문제점은 기존 방법에서 유효한 삼원조(triplet)를 생성하기 위해 LLM 프롬프트에 KG 스키마를 포함시켜야 한다는 점이다. 더 크고 복잡한 스키마는 LLM의 컨텍스트 윈도우 길이를 쉽게 초과한다. 또한 고정된 사전 정의 스키마가 없는 시나리오에서는 간결한 자체 생성 스키마로 고품질 KG를 구축할 수 있는 방법이 필요하다. 이러한 문제를 해결하기 위해 우리는 '추출-정의-정규화(EDC)'라는 3단계 프레임워크를 제안한다: 공개 정보 추출에 이어 스키마 정의 및 사후 정규화 단계로 구성된다. EDC는 사전 정의된 대상 스키마가 존재하는 환경과 그렇지 않은 환경 모두에 적용 가능하다는 점에서 유연하다. 후자의 경우 자동으로 스키마를 구축하고 자체 정규화를 적용한다. 성능 향상을 위해 입력 텍스트와 관련된 스키마 요소를 검색하는 학습된 구성 요소를 도입합니다. 이는 검색 강화 생성 방식과 유사하게 LLM의 추출 성능을 향상시킵니다. 세 가지 KGC 벤치마크를 통해 EDC가 매개변수 조정 없이도 기존 연구 대비 훨씬 더 큰 스키마로 고품질 삼원조 추출이 가능함을 입증합니다. EDC 코드는 https://github.com/clear-nus/edc에서 확인할 수 있습니다.

1 서론

지식 그래프(KGs)(Ji et al., 2021)는 그래프 구조를 통해 상호 연결된 정보를 조직화하는 지식의 구조화된 표현으로, 엔티티와 관계는 노드와 에지로 표현됩니다. 이들은 광범위하게

그림 1: 지식 그래프 구축을 위한 추출-정의-정규화(EDC)의 고수준 개요.

다양한 하위 작업에 활용되며, 의사 결정(Guo et al., 2021; Lan et al., 2020), 질문응답(Huang et al., 2019; Yasunaga et al., 2021), 추천(Guo et al., 2020; Wang et al., 2019) 등이 포함됩니다. 그러나 지식 그래프 구축(KGC)은 본질적으로 어려운 과제입니다: 일관되고 간결하며 의미 있는 지식 그래프를 생성하기 위해서는 구문과 의미에 대한 이해 능력이 요구됩니다. 따라서 KGC는 주로 집중적인 인적 노동에 의존합니다(Ye et al., 2022). KGC는 광범위한 문제이며, 본 연구에서는 **관계 삼중항 추출 작업에 초점을 맞춥니다**. 이는 KGC에 매우 중요하기 때문입니다.

KGC에 핵심적이기 때문이다. 선행 연구(Ye et al., 2022; Melnyk et al., 2022; Bi et al., 2024)를 따라, 본 연구에서 다루는 과제도 여전히 KGC로 지칭한다.

최근 KGC 자동화 시도(Zhong et al., 2023; Ye et al., 2022)는 뛰어난 자연어 이해 및 생성 능력을 고려하여 대규모 언어 모델(LLMs)을 활용하였다. LLM 기반 KGC 방법은 지식 그래프를 나타내는 엔티티-관계 삼중항을 생성하기 위해 다중 회화(Wei et al., 2023) 및 코드 생성(Bi et al., 2024)과 같은 다양한 혁신적인 프롬프트 기반 기법을 사용합니다. 그러나 이러한 방법들은 현재 소규모 및 특정 도메인 시나리오로 제한되어 있습니다. 생성된 삼중항(triplet)의 유효성을 보장하기 위해 스키마 정보(예: 가능한 엔티티 및 관계 유형)를 프롬프트에 포함시켜야 하기 때문입니다. 복잡한 데이터셋(예: 위키피디아)은 일반적으로 **컨텍스트 창 길이를** 초과하거나 LLM에 의해 무시될 수 있는 **대규모 스키마를** 요구합니다(Wadhwa et al., 2023). 더욱이 **사전 정의된 스키마가 항상 이용 가능한 것은 아닙니다**. 사용자는 사전에 관심 정보에 대한 명확한 의도를 가지고 있지 않을 수 있지만, 여전히 본질적으로 고품질의 지식 그래프를 추출하고자 할 수 있습니다. 기존 방법이 이러한 상황에서 어떻게 작동할지는 불분명합니다.

이러한 문제점을 해결하기 위해, 우리는 KGC를 위한 구조화된 접근법인 **추출-정의-정규화(EDC)**를 제안한다: 핵심 아이디어는 KGC를 세 가지 하위 작업에 해당하는 세 가지 주요 단계로 분해하는 것이다(그림 1):

1. 정보 추출 개방: 입력 텍스트에서 자유롭게 엔티티-관계 삼원조 목록을 추출합니다.

2. 스키마 정의: 추출 단계에서 얻은 삼원조로부터 유도된 스키마의 각 구성 요소(예: 엔티티 유형 및 관계 유형)에 대한 정의를 생성합니다.

3. 스키마 정규화: 스키마 정의를 활용하여 의미적으로 동등한 엔티티/관계 유형이 동일한 명사구/관계 구문을 가지도록 삼중항을 표준화합니다.

각 단계는 대규모 언어 모델(LLM)의 강점을 활용합니다: 추출 하위 작업은 LLM이 효과적인 개방형 정보 추출기라는 최근 연구 결과(Li et al., 2023; Han et al., 2023)를 활용합니다 — LLM은 의미적으로 정확하고 의미 있는 삼원조를 추출할 수 있습니다. 그러나 결과 삼원조는 일반적으로 중복적이고 모호한 정보를 포함합니다. 예를 들어, '직업', '일자리', '직책'과 같은 다중 '직업', '직책', '업종'과 같이 의미적으로 동등한 관계 구문이 포함됩니다(Kamp et al., 2023; Putri et al., 2019; Vashishth et al., 2018).

2단계와 3단계(정의 및 정규화)는 삼중항을 표준화하여 하위 작업에 유용하게 활용할 수 있도록 합니다. EDC는 유연성을 고려하여 설계되었습니다: 기존에 존재하는 대규모 스키마와 일관된 삼중항을 발견하거나(**대상 정렬**) *자체적으로* 스키마를 *생성할* 수 있습니다(**자체 정규화**). 이를 위해 설명 생성 능력을 활용하여 LLM으로 스키마 구성 요소를 정의합니다. LLM은 인간 전문가도 납득할 수 있는 설명을 통해 추출 결과를 정당화할 수 있습니다(Li et al., 2023). 정의된 스키마는 벡터 유사도 검색을 통해 가장 근접한 엔티티/관계 유형 후보를 찾는 데 활용되며, LLM은 이를 참조하여 구성 요소를 정규화합니다. 기존 스키마에 대응하는 항목이 없는 경우, 스키마를 풍부하게 하기 위해 해당 항목을 추가할 수 있습니다.

성능을 더욱 향상시키기 위해, 위의 세 단계에 추가적인 **정제** 단계를 수행할 수 있습니다: 초기 추출 과정에서 EDC를 반복하되, 이전에 추출된 삼중항과 스키마의 관련 부분을 프롬프트에 제공합니다. 우리는 입력 텍스트와 관련된 스키마 구성 요소를 검색하는 훈련된 **스키마** 검색기를 제안합니다. 이는 검색 강화 생성(Lewis et al., 2020)과 유사하며, 생성된 삼중항을 개선하는 것으로 나타났습니다.

타겟 정렬 및 자체 정규화 설정에서 세 가지 KGC 데이터셋에 대한 실험 결과, EDC는 자동 및 수동 평가를 통해 최신 방법에 비해 더 높은 품질의 KG를 추출할 수 있음을 보여줍니다. 또한 스키마 리트리버의 사용이 EDC의 성능을 현저하고 일관되게 향상시키는 것으로 나타났습니다.

요약하면, 본 논문은 다음과 같은 기여를 한다:

- EDC는 유연하고 성능이 우수한 LLM 기반 지식 그래프 구축 프레임워크로, 대규모 스키마 또는 사전 정의된 스키마 없이도 고품질 지식 그래프를 추출할 수 있습니다.

- 스키마 리트리버(Schema Retriever): 정보 검색과 유사한 방식으로 입력 텍스트와 관련된 스키마 구성 요소를 추출하도록 훈련된 모델입니다.

- EDC와 스키마 검색기의 효과성을 입증하는 경험적 증거.

2 배경

이 섹션에서는 지식 그래프 구축(KGC), 공개 정보 추출(OIE), 정규화에 관한 관련 배경을 제공합니다.

지식 그래프 구축. 기존 방법들은 일반적으로 엔티티 발견(Žukov-Gregoricˇ 외, 2018; Martins et al., 2019), 엔티티 유형 분류(Choi et al., 2018; Onoe and Durrett, 2020), 관계 분류(Zeng et al., 2014, 2015) 등으로 구성된다. 사전 훈련된 생성형 언어 모델(예: T5 (Raffel et al., 2020) 및 BERT(Lewis et al., 2019) 등)의 발전 덕분에, 최근 연구들은 KGC를 시퀀스-투-시퀀스 문제로 재구성하고 중간 규모의 언어 모델을 미세 조정하여 관계 삼중항을 종단 간 방식으로 생성합니다(Ye et al., 2022). 대규모 언어 모델(LLM)의 성공은 이 패러다임을 한 단계 더 발전시켰습니다: 현재 방법들은 LLM에 직접 프롬프트를 제공하여 제로/소량 샷 방식으로 삼중항을 생성하도록 합니다. 예를 들어, ChatIE(Wei et al., 2023)는 작업을 다중 회전 질문-응답 문제로 구성하여 삼중항을 추출하고, CodeKGC(Bi et al., 2024)는 작업을 코드 생성 문제로 접근합니다. 앞서 언급한 바와 같이, 이러한 모델들은 KG 스키마를 LLM 프롬프트에 포함시켜야 하기 때문에 많은 실제 응용 분야에서 흔히 볼 수 있는 일반 텍스트로 확장하는 데 어려움을 겪습니다. 우리의 EDC 프레임워크는 사후 정규화(post-hoc canonicalization)를 사용하여(기본 LLM의 미세 조정이 필요 없이) 이 문제를 해결합니다.

개방형 정보 추출과 정규화. 표준(폐쇄형) 정보 추출은 출력 삼중항이 사전 정의된 스키마를 따르도록 요구합니다. 예를 들어 추출 대상 관계 또는 엔티티 유형 목록이 이에 해당합니다. 반면, 개방형 정보 추출(OIE)은 이러한 요구 사항이 없습니다. OIE는 오랜 역사를 가지고 있으며, 포괄적인 내용을 원하는 독자들은 훌륭한 서베이 논문들(Liu et al., 2022; Zhou et al., 2022; Kamp et al., 2023)을 참고하시기 바랍니다. 최근 연구에서는 대규모 언어 모델(LLMs)이 OIE 작업에서 탁월한 성능을 보인다는 사실이 밝혀졌습니다(Li et al., 2023). 그러나 OIE 시스템에서 추출된 관계 삼중항은 정규화되지 않습니다. 의미적으로 동등한 여러 관계가 정규형으로 통합되지 않은 채 공존할 수 있어 유도된 개방형 지식 그래프에 중복성과 모호성을 초래합니다. 삼중항을 표준화하기 위해서는 추가적인 정규화 단계가 필요합니다.

하류 응용 분야에 유용한 KGs.

표준화 방법은 대상 스키마의 유무에 따라 달라집니다. 대상 스키마가 존재하는 경우, 이 작업은 때때로 "정렬(alignment)"이라고도 불립니다(Putri et al., 2019). 예를 들어, Putri 등(2019)은 OIE에서 추출한 관계 구문에 대한 정의를 얻기 위해 WordNet(Miller, 1995)을 부가 정보로 사용하고, OIE 관계 정의와 대상 스키마의 사전 정의된 관계를 비교하기 위해 시아미즈 네트워크를 사용합니다. 대상 스키마를 사용할 수 없는 경우, 최첨단 방법은 일반적으로 클러스터링을 기반으로 합니다(Vashishth 외, 2018; Dash 외, 2020). CESI(Vashishth et al., 2018)는 PPDB(Ganitkevitch et al., 2013) 및 WordNet과 같은 외부 소스의 부가 정보를 사용하여 OIE 관계에 대한 임베딩을 생성합니다. 그러나 클러스터링 기반 방법은 과도한 일반화에 취약합니다(Kamp et al., 2023; Putri et al., 2019)에 취약합니다. 예를 들어, CESI는 "is brother of", "is son of", "is main villain of", "was professor of"를 동일한 관계 클러스터에 포함시킬 수 있습니다.

기존의 정규화 방법과 비교하여 EDC는 더 일반적입니다. 대상 스키마가 제공되든 안 되든 작동합니다. WordNet과 같은 정적 외부 소스를 사용하는 대신, EDC는 대규모 언어 모델(LLM)이 생성한 문맥적이며 의미적으로 풍부한 부가 정보를 활용합니다. 또한, 변환이 수행 가능한지 여부를 LLM이 검증하도록 허용함으로써(단순히 임베딩 유사성에만 의존하지 않음), EDC는 기존 방법이 직면한 과도한 일반화 문제를 완화합니다.

3 방법: KGC를 위한 EDC

본 절에서는 우리의 주요 기여 사항인 대규모 언어 모델(LLM)을 구조화된 방식으로 활용하여 지식 그래프를 구축하는 접근법을 개괄한다. 먼저 EDC 프레임워크를 상세히 설명한 후 정제(**EDC+R**)에 대해 기술한다. 입력 텍스트를 주어졌을 때, 우리의 목표는 결과 지식 그래프(KG)의 모호성과 중복성을 최소화할 수 있도록 표준화된 형태의 관계 삼중항을 추출하는 것이다. 사전 정의된 대상 스키마가 존재할 경우, 생성된 모든 삼중항은 해당 스키마를 준수해야 한다. 스키마가 없는 경우, 시스템은 동적으로 스키마를 생성하고 이를 기준으로 삼아 삼원조를 표준화해야 합니다.

3.1 EDC: 추출-정의-정규화

EDC는 KGC를 세 가지 연결된 하위 작업으로 분해합니다. 논의를 구체화하기 위해 특정 입력 텍스트 예시를 사용하겠습니다: "*앨런 셰퍼드는 1923년 11월 18일에 태어났으며*

그는 아폴로 14호 승무원 중 한 명이었습니다." 각 단계를 차례로 살펴보겠습니다:

1단계: 공개 정보 추출: 먼저 대규모 언어 모델(LLM)을 활용해 공개 정보 추출을 수행합니다. 소량 데이터 프롬프팅을 통해 LLM은 특정 스키마에 구애받지 않고 입력 텍스트에서 관계 삼원조([주체, 관계, 객체])를 식별 및 추출합니다. 위 예시를 적용하면 프롬프트는 다음과 같습니다:

OIE 프롬프트

주어진 텍스트에서 [주체, 관계, 대상] 형식의 관계 삼원조(triplet)를 추출하라. 예시:
예시 1:
텍스트: 길이 17068.8 밀리미터의 ALCO RS-3는 디젤-전기 변속기를 장착하고 있습니다.
삼원조: [['ALCO RS-3', 'powerType', '디젤-전기 변속기'], ['ALCO RS-3', 'length', '17068.8 (millimetres)']] ...
이제 다음 텍스트에서 삼중항을 추출해 주세요: 앨런 셰퍼드는 1923년 11월 18일에 태어났으며 1959년 NASA에 선발되었습니다. 그는 아폴로 14호 승무원이었습니다.

추출된 삼중항(['Alan Shepard', 'bornOn', 'Nov 18, 1923'], ['Alan Shepard', 'participatedIn', 'Apollo 14'])은 *개방형 지식 그래프(KG)*를 형성하며, 이는 후속 단계로 전달됩니다.

2단계: 스키마 정의: 다음으로, LLM에게 오픈 KG에서 유도된 스키마의 각 구성 요소에 대한 자연어 정의를 제공하도록 요청합니다:

스키마 정의 프롬프트

주어진 텍스트와 그로부터 추출된 관계 삼원조 목록을 바탕으로, 존재하는 각 관계에 대한 정의를 작성하십시오.
예시 1:
텍스트: 길이 17068.8 밀리미터의 ALCO RS-3는 디젤-전기 변속기를 장착하고 있습니다.
트리플릿: [['ALCO RS-3', 'powerType', '디젤-전기 변속기'], ['ALCO RS-3', 'length', '17068.8 (밀리미터)']]
정의:
powerType: 주체 엔티티가 객체 엔티티가 지정한 유형의 동력 또는 에너지원을 사용합니다.
…
이제 다음 텍스트에서 추출한 삼중항마다 존재하는 각 관계에 대한 정의를 작성하세요: 텍스트: 앨런 셰퍼드는 1923년 11월 18일 뉴햄프셔에서 태어난 미국인으로, 1959년 NASA에 선발되었으며, 아폴로 14호 승무원이었고 캘리포니아에서 사망했습니다.
삼중항: [['앨런 셰퍼드', 'bornOn', '1923년 11월 18일'], ['앨런 셰퍼드', 'participatedIn', '아폴로 14']]

이 예시 프롬프트는 (bornOn: 주체 엔티티가 객체 엔티티가 지정한 날짜에 태어남) 및 (participatedIn: 주체 엔티티가 객체 엔티티가 지정한 사건 또는 임무에 참여함)에 대한 정의를 생성하며, 이는 정규화를 위한 *부가 정보로* 다음 단계로 전달됩니다.

3단계: 스키마 정규화: 세 번째 단계는 중복과 모호성을 제거하여 개방형 지식 그래프를 정규화된 형태로 정제하는 것을 목표로 합니다. 먼저 문장 변환기를 사용하여 각 스키마 구성 요소의 정의를 벡터화하여 임베딩을 생성합니다. 이후 대상 스키마의 유무에 따라 두 가지 방식 중 하나로 정규화가 진행됩니다:

- 대상 정렬: 기존 대상 스키마를 바탕으로 각 요소 내에서 가장 밀접하게 관련된 구성 요소를 식별하여 정규화 대상으로 고려합니다. 과도한 일반화 문제를 방지하기 위해 대규모 언어 모델(LLM)은 각 잠재적 변환의 실행 가능성을 평가합니다. 변환이 비합리적이라고 판단될 경우(대상 스키마 내 의미적 동등체가 존재하지 않음을 시사함), 해당 구성 요소와 관련된 삼중항은 제외됩니다.
- 자체 정규화: 대상 스키마가 부재할 경우, 의미적으로 유사한 스키마 구성 요소를 통합하여 단일 표현으로 표준화함으로써 지식 그래프를 간소화하는 것이 목표입니다. 빈 정규화 스키마를 시작으로, 공개 지식 그래프 트리플을 검토하며 벡터 유사도와 LLM 검증을 통해 잠재적 통합 후보를 탐색합니다. 대상 정렬과 달리, 변환 불가능한 것으로 판단된 구성 요소는 정규화 스키마에 추가되어 이를 확장합니다.

예시를 사용한 프롬프트는 다음과 같습니다:

스키마 표준화 프롬프트

주어진 텍스트, 그로부터 추출된 관계 삼중항, 그리고 그 안에 포함된 관계의 정의가 주어졌을 때, 해당 맥락에서 이를 대체할 가장 적절한 관계가 존재한다면 그것을 선택하십시오.
텍스트: 앨런 셰퍼드는 1923년 11월 18일에 태어났으며 1959년 NASA에 선발되었습니다. 그는 아폴로 14호 승무원 중 한 명이었습니다.
삼원조: ['앨런 셰퍼드', '참여했다', '아폴로 14호']
'참여하다(participatedIn)' 정의: 주체 엔티티

는 목적어 개체가 지정한 사건이나 임무에 참여했다.
선택지:
A. 'mission': 주체 개체가 객체 개체가 지정한 사건 또는 작전에 참여했다.
B. '시즌': 주체 개체가 객체 개체가 지정한 시리즈의 시즌에 참여했다.
...
F. 위의 어느 것도 해당되지 않음

위 선택지는 벡터 유사도 검색을 통해 도출된 것임을 유의하십시오. LLM이 선택을 완료한 후 관계는 다음과 같이 변환됩니다:

['Alan Shepard', 'birthDate', 'Nov 18, 1923'],

['Alan Shepard', 'mission', 'Apollo 14'], 이는 우리의 정규화된 지식 그래프를 형성합니다.

3.2 EDC+R: 스키마 검색기를 통해 EDC를 반복적으로 정제합니다.

정제 과정은 추출된 삼중항(triplet)의 품질을 향상시키기 위해 EDC가 생성한 데이터를 활용합니다. 검색 강화 생성(retrieval-augmented generation) 및 선행 연구(Bi et al., 2024)에서 영감을 받아, 추출 단계에 대한 "힌트"를 구성합니다(자세한 내용은 부록 A.4 참조). 이 힌트는 두 가지 주요 요소로 구성됩니다:

- 후보 엔티티: 이전 반복에서 EDC가 추출한 엔티티와 LLM을 사용하여 텍스트에서 추출한 엔티티;
- 후보 관계: 이전 사이클에서 EDC가 추출한 관계와 훈련된 스키마 검색기를 사용하여 사전 정의/정규화된 스키마에서 검색된 관계.

LLM과 스키마 검색기에서 추출된 엔티티 및 관계를 모두 포함함으로써 LLM에 더 풍부한 후보 풀을 제공하며, 이는 엔티티나 관계의 부재로 인해 LLM의 효율성이 저하되는 문제를 해결합니다. 이전 단계에서 추출된 엔티티와 관계를 엔티티 추출 및 스키마 검색의 새로운 결과와 병합함으로써, 이 힌트는 이전 라운드의 결과를 기반으로 부트스트랩하여 OIE를 지원합니다.

EDC를 대규모 스키마로 확장하기 위해, 우리는 훈련된 스키마 검색기를 활용하여 스키마를 효율적으로 검색합니다. 스키마 검색기는 벡터 공간 기반 정보 검색 방법(Ganguly et al., 2015; Lewis et al., 2020)과 유사한 방식으로 작동합니다. 스키마 구성 요소와 입력 텍스트를 벡터 공간으로 투영하여 코사인 유사도가 두 요소 간의 관련성, 즉 스키마 구성 요소가 입력 텍스트에 존재할 가능성을 포착하도록 합니다.

우리의 설정에서 유사도 공간은 벡터 공간 내 코사인 유사도로 의미적 동등성을 포착하는 표준 문장 임베딩 모델과 다릅니다. 우리의 스키마 검색기는 문장 임베딩 모델 E5-mistral-7b-instruct(Wang et al., 2023)의 미세 조정된 변형입니다. 본 논문에서 상세히 설명된 원본 훈련 방법론을 따릅니다. 이는 텍스트 쌍과 그에 대응하는 정의된 관계 쌍을 활용하는 것을 포함합니다. 자세한 내용은 부록 A.3을 참조하십시오. 주어진 긍정적 텍스트-관계 쌍 (t^+, r^+)에 대해, t^+에 대한 지시문 템플릿을 사용하여 새로운 텍스트를 생성합니다

t^+_{inst} = "지시: 존재하는 관계를 검색하라
주어진 텍스트 \n 쿼리: $\{t^+\}$".

그런 다음 InfoNCE 손실을 사용하여 임베딩 모델을 미세 조정하여 주어진 텍스트와 관련된 올바른 관계와 다른 관련 없는 관계를 구별합니다.

예시로 돌아가면, 스키마 검색기를 통한 정제 과정은 기존 관계 집합에 다음과 같은 관계를 추가합니다: ['Alan Shepard', 'selectedByNasa', '1959']. 'selectedByNasa' 관계는 다소 생소하지만 대상 스키마에 명시된 내용입니다.

4 실험

이 섹션에서는 EDC와 EDC+R의 성능을 평가하기 위해 설계된 실험을 설명합니다. 간단히 말해, 우리의 결과는 EDC가 타겟 정렬 및 자체 정규화 설정 모두에서 최신 기법을 크게 능가함을 보여줍니다. 정교화는 EDC를 더욱 향상시킵니다. EDC의 소스 코드와 실험 재현을 위한 자료는 부록 C의 전체 표와 함께 보충 자료에서 확인할 수 있습니다.

4.1 실험 설정

데이터셋. 우리는 세 가지 KGC 데이터셋을 사용하여 EDC를 평가합니다:

- WebNLG (Ferreira et al., 2020): WebNLG+2020 (v3.0)의 의미 분석 작업에서 테스트 분할을 사용합니다. 여기에는 텍스트와 삼중항 쌍 1165개가 포함됩니다. 이러한 참조 삼중항에서 도출된 스키마는 159개의 고유한 관계 유형을 포괄합니다.
- REBEL (Cabot and Navigli, 2021): REBEL의 원래 테스트 분할은 105,516개의 항목으로 구성됩니다. 비용 관리를 위해 1000개의 텍스트-트리플렛 쌍을 무작위로 선택합니다. 이 하위 집합은 200개의 고유한 관계 유형을 가진 스키마를 유도합니다.

- Wiki-NRE (Distiawan et al., 2019): Wiki-NRE의 테스트 분할(29,619개 항목)에서 1000개의 텍스트-트리플렛 쌍을 샘플링하여 45개의 고유한 관계 유형을 가진 스키마를 유도합니다.

이 데이터셋들은 ADE(Gurulingappa et al., 2012)(관계 유형 1종), SciERC(Luan et al., 2018) (7가지 관계 유형), CoNLL04(Roth and Yih, 2004)(4가지 관계 유형) 등 기존 LLM 기반 방법 평가에 사용된 대안들보다 관계 유형의 다양성이 풍부하기 때문입니다. 이러한 다양성은 실제 세계의 복잡성을 더 잘 모방합니다. 본 실험에서는 모든 데이터셋에 걸쳐 유일하게 활용 가능한 스키마 구성 요소로서 관계 추출에 집중한다. 관계는 지식 그래프의 기초 요소이므로, 엔티티나 이벤트 유형 같은 다른 구성 요소보다 우선시된다. 다만 EDC는 다른 스키마 구성 요소로도 쉽게 확장 가능하다는 점을 유의해야 한다.

EDC 모델. EDC는 대규모 언어 모델(LLM)로 구동되는 여러 모듈을 포함합니다. OIE 모듈은 KG에 포착되는 의미적 내용을 결정하는 핵심 상류 모듈이므로, GPT-4(Achiam et al., 2023), GPT-3.5-turbo(Brown et al., 2020), Mistral-7b(Jiang et al., 2023) 등 다양한 크기의 LLM을 테스트했습니다. Mistral-7b는 로컬 워크스테이션에 배포된 반면, GPT 모델들은 OpenAI API를 통해 접근했습니다. 프롬프트가 필요한 프레임워크의 나머지 구성 요소에는 GPT-3.5-turbo를 사용했습니다. 정규화 단계에서는 수정 없이 벡터 유사도 검색을 위해 E5-Mistral-7b 모델을 활용했습니다.

4.1.1 평가 기준 및 기준선

타깃 정렬(스키마 제공 시)과 자체 정규화(스키마 없음) 환경에서 본 방법론을 별도로 평가합니다. 이는 *본질적으로 다른 목표를* 지향하기 때문입니다: 전자는 타깃 스키마와 일치하는 정답 주석 삼중항을 복원하는 반면, 후자는 사전 정의된 비교 대상 없이 간결하고 중복 없는 지식 그래프를 유도하는 의미론적으로 정확하고 의미 있는 삼중항을 추출하는 데 목적이 있습니다. 상기 데이터셋의 경우, 기존 LLM 기반 KGC 방법(ChatIE 및 CodeKGC)은 스키마 규모로 인해 적용할 수 없었습니다. EDC는 소규모 도메인 특화 데이터셋을 대상으로 하지 않지만, 평가의 포괄성을 위해 SciERC 및 CoNLL04에 대한 결과를 부록 E에 포함합니다.

목표 정렬. 각 데이터셋에 대해 EDC와 EDC+R을 해당 데이터셋 전용으로 훈련된 모델과 비교합니다.

각 데이터셋별로:

- **REGEN**(Dognin et al., 2021)은 웹 자연어 생성(WebNLG) 분야의 최첨단 모델(SOTA)입니다. 이 모델은 사전 훈련된 T5(Raffel et al., 2020)와 강화 학습(RL)을 활용하여 양방향 텍스트-그래프 및 그래프-텍스트 생성을 수행하는 시퀀스-투-시퀀스 모델입니다.

- **GenIE**(Josifoski et al., 2022)는 사전 훈련된 BART(Lewis et al., 2019)와 제약 생성 전략을 활용하여 출력 삼중항이 사전 정의된 스키마와 일관되도록 제한하는 시퀀스-투-시퀀스 모델입니다. GenIE는 REBEL 및 Wiki-NRE 분야의 최신 기술 수준 모델입니다.

기존 연구(Dognin et al., 2021; Melnyk et al., 2022)를 따라, 토큰 기반 방식으로 출력 삼중항과 정답 간의 정밀도(Precision), 재현율(Recall), F1 점수를 계산하는 WEBNLG 평가 스크립트(Ferreira et al., 2020)를 사용합니다. 명명된 엔티티 평가를 기반으로 한 메트릭을 사용하여 세 가지 다른 방식으로 정밀도, 재현율 및 F1 점수를 측정했습니다.

- *정확:* 후보와 참조 트리플이 유형(주어, 관계, 목적어)을 무시하고 완전히 일치해야 합니다.

- *부분 일치:* 유형을 무시하고 후보 트리플과 참조 트리플 간의 최소한 부분 일치를 허용합니다.

- *엄격:* 후보 트리플렛과 참조 트리플렛 간에 요소 유형을 포함한 정확한 일치를 요구합니다.

자기 정규화. 자기 정규화 성능 평가를 위해 다음 항목들과 비교를 수행합니다:

- **기준 오픈 KG**: OIE(Open Information Extraction) 단계에서 생성된 초기 오픈 KG 출력입니다. 이는 정규화 과정으로 인한 정밀도와 스키마 간결성의 변화를 설명하기 위한 기준점이 됩니다.

- **CESI**(Vashishth et al., 2018): 오픈 KG 정규화를 위한 선도적인 클러스터링 기반 접근법으로 인정받고 있습니다. CESI를 오픈 KG에 적용함으로써 EDC에 의한 정규화 성능과 대비를 도모합니다.

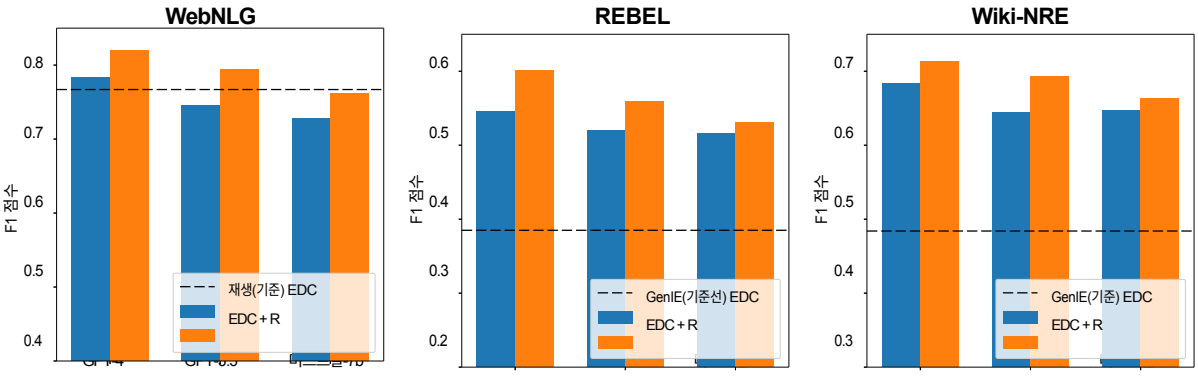

그림 2: 대상 정렬 설정에서 WebNLG, REBEL 및 Wiki-NRE 데이터셋에 대한 EDC 및 EDC+R의 성능을 기준 모델과 비교한 결과('부분' 기준을 적용한 F1 점수). EDC+R은 한계 개선 효과가 감소함에 따라 정제 과정을 한 번만 수행합니다.

정규화된 삼중항은 참조 삼중항과 표현이 다르거나 스키마 외 관계일 수 있으므로, 토큰 기반 평가는 부적합합니다. 따라서 추출된 지식 그래프의 본질적 품질을 반영하는 세 가지 핵심 측면에 초점을 맞춘 수동 평가를 수행합니다:

- *정밀도:* 정규화된 삼중항은 OIE 삼중항에 비해 텍스트에 대해 정확하고 의미 있는 상태를 유지합니다.
- *간결성:* 스키마의 간결성은 관계 유형의 수로 측정됩니다.
- *중복성:* 우리는 중복 점수를 사용합니다 — 각 정규화된 관계와 가장 가까운 대응 관계 간의 평균 코사인 유사도 — 를 사용하며, 낮은 점수는 스키마의 관계들이 의미적으로 구별된다는 것을 나타냅니다.

4.2 결과 및 분석

다음에서는 주요 발견 사항과 결과를 전달하는 데 중점을 둡니다. 전체 결과와 표는 부록을 참조하십시오.

4.2.1 목표 정렬

그림 2의 막대 그래프는 OIE를 위해 서로 다른 LLM을 사용한 세 데이터셋 전체에서 EDC와 EDC+R이 얻은 부분 F1 점수를 각 기준 모델과 비교하여 요약한 것입니다. **EDC는 평가된 모든 데이터셋에서 최첨단 기준 모델보다 우수하거나 동등한 성능을 보여줍니다**. LLM 간 비교 시 GPT-4가 최상위 성능을 보였으며, Mistral-7b와 GPT-3.5-turbo는 유사한 결과를 나타냈습니다. REBEL 및 Wiki-NRE 데이터셋에서는 본 방법론과 기준 모델 간의 격차가 더욱 두드러졌습니다.

데이터셋; 이는 주로 GenIE의 제한된 생성 접근 방식 때문인데, 숫자나 날짜 같은 리터럴을 포함하는 삼중항 추출에 미흡합니다.

정제(EDC+R)는 일관되고 유의미하게 성능을 향상시킵니다. 정제 후 GPT-3.5-turbo와 Mistral-7b 간의 성능 차이는 더 커져, Mistral-7b가 제공된 힌트를 활용하는 능력이 상대적으로 부족했음을 시사합니다. 그럼에도 불구하고 힌트를 활용한 단일 정제 반복은 테스트된 모든 대규모 언어 모델(LLM)의 성능을 개선했습니다.

점수 분석 결과, EDC 성능은 REBEL 및 Wiki-NRE 대비 WebNLG에서 현저히 우수한 것으로 나타났습니다. 그러나 후자 데이터셋에서 유효한 삼중항을 생성했음에도 EDC가 감점을 받는 현상을 관찰했습니다. 그 이유는 해당 데이터셋의 참조 삼중항이 완전하지 않기 때문입니다. 예를 들어, REBEL 데이터셋의 텍스트 *'Romany Love는 프레드 폴이 감독하고 에스몬드 나이트, 플로렌스 맥휴, 로이 트래버스가 출연한 1931년 영국 뮤지컬 영화이다.'*를 고려해 보겠습니다. EDC는 ['Romany Love', '출연진', 'Esmond Knight'], ['Romany Love', '출연진', 'Florence McHugh'], ['Romany Love', '출연진', 'Roy Travers']를 추출하는데, 이는 모두 의미론적으로 정확하지만 참조 세트에는 첫 번째 삼중항만 존재합니다. 데이터셋에는 텍스트와 무관한 정보를 기반으로 한 참조 삼중항도 포함되어 있습니다. 예를 들어, *'Daniel is an Ethiopian foot-baller, who currently plays for Hawassa City S.C.'*에는 대응하는 참조 삼중항 ['Hawassa City S.C.', 'country', 'Ethiopia']이 있습니다.

이러한 문제는 해당 데이터셋 생성 시 채택된 서로 다른 방법론에서 기인할 수 있습니다. WebNLG의 경우, 어노테이터들에게 다음과 같은 작업을 요청했습니다.

표 1: 스키마 리트리버에 대한 제거 연구 결과(모든 기준에 따른 F1 점수). OIE에 사용된 LLM은 GPT-3.5-turbo입니다. S.R.은 스키마 리트리버(Schema Retriever)를 의미합니다.

데이터셋	방법	부분	엄격	정확
웹NLG	EDC+R	**0.794**	**0.753**	**0.772**
	EDC+R (S.R. 제외)	0.752	0.701	0.721
	EDC	0.746	0.688	0.713
REBEL	EDC+R	**0.559**	**0.516**	**0.529**
	EDC+R (S.R. 제외)	0.517	0.466	0.482
	EDC	0.506	0.449	0.473
위키-NRE	EDC+R	**0.693**	**0.685**	**0.657**
	EDC+R (S.R. 제외)	0.653	0.645	0.641
	EDC	0.647	0.638	0.640

텍스트를 오직 삼중항만으로 구성한다. 따라서 텍스트와 삼중항은 직접적인 대응 관계를 가지며, 텍스트는 일반적으로 삼중항에서 명백히 드러나는 정보 외에는 포함하지 않는다. 반면 REBEL과 Wiki-NRE는 원격 감독(Smirnova and Cudré-Mauroux, 2018)을 사용하여 텍스트와 삼중항을 정렬함으로써 생성된다. 이러한 접근법은 삼중항 추출이 덜 직관적이고 참조 세트가 불완전해질 수 있어, 데이터셋에 존재하지 않는 올바른 삼중항을 생성하는 EDC와 같은 방법에 대해 지나치게 비관적인 평가를 초래할 수 있습니다(Han et al., 2023; Wadhwa et al., 2023). 평균적으로 EDC는 REBEL 및 Wiki-NRE에서 참조 세트 대비 문장당 하나의 추가 트리플렛을 추출하는 반면, WebNLG에서는 참조 세트와 유사한 수의 트리플렛을 추출합니다.

스키마 검색기의 제거 연구. 정제 과정에서 스키마 검색기가 제공하는 관계의 영향을 평가하기 위해, GPT-3.5-turbo를 사용하여 이러한 관계를 제거하는 제거 연구를 수행했습니다. 표 1의 결과는 **스키마 검색기를 제거하면 성능이 저하됨을** 보여줍니다. 질적으로 살펴보면, 스키마 리트리버는 OIE 단계에서 LLM이 식별하기 어려운 관련 관계를 찾는 데 도움이 되는 것으로 나타났습니다. 예를 들어, *'디종의 부르고뉴 대학에는 16,800명의 학부생이* 있다'는 텍스트에서 LLM은 OIE 단계에서 ['부르고뉴 대학', '위치', '디종'] 관계를 추출합니다. 의미론적으로 정확하지만, 이 관계는 대상 스키마에 존재하는 대학 위치를 나타내는 보다 구체적인 관계인 '캠퍼스'를 간과합니다. 스키마 검색기는 이 더 세분화된 관계를 성공적으로 식별하여, LLM이 추출 결과를 ['부르고뉴 대학교', '캠퍼스', '디종']으로 조정할 수 있게 합니다. 이 실험은 스키마 검색기가 정확하고 문맥에 적합한 관계 추출을 용이하게 하는 데 있어 그 가치를 부각시킵니다.

표 2: 자체 정규화 설정에서의 EDC 성능 (인간 평가 정밀도 및 스키마 지표). 각 데이터셋 및 지표별 최고 결과는 굵게 표시됨. Prec.는 정밀도(Precision), No. Rel.는 관계 수(Number of Relations), Red.는 중복도 점수(Redundancy Score)를 의미함.

데이터셋	방법	정밀도(↑)	관계 수(↓)	관계 수 감소(↓)
웹NLG	EDC	**0.956**	**200**	**0.833**
	CESI	0.724	280	0.893
	오픈 KG	0.982	529	0.927
REBEL	EDC	**0.867**	**225**	**0.831**
	CESI	0.504	307	0.854
	오픈 KG	0.903	667	0.895
Wiki-NRE	EDC	**0.898**	**106**	**0.833**
	CESI	0.753	114	0.849
	오픈 KG	0.909	204	0.881

4.2.2 자체 정규화

여기서는 EDC의 자체 정규화 성능 평가(OIE에 GPT-3.5-turbo 활용)에 초점을 맞춥니다. 자체 정규화 설정에서의 정교화는 이미 상기 연구 및 후속 반복 과정에서 다루어졌으며, 자체 구축된 정규화 스키마가 목표 스키마가 되므로 생략합니다. 기존 연구(Wadhwa et al., 2023; Kolluru et al., 2020)를 따라 지식 그래프에 대한 타깃 인간 평가를 수행했습니다. 이 평가는 시스템 세부 사항에 대한 사전 지식 없이 주어진 텍스트에서 추출된 삼중항(triplet)의 타당성을 평가하는 두 명의 독립적인 주석자가 참여했습니다. 주석자 간 높은 일치도 점수(0.94)를 관찰했습니다.

평가 결과와 스키마 메트릭스는 표 2에 요약되어 있다. OIE 단계에서 생성된 오픈 지식 그래프는 의미적으로 유효한 삼원조(triplet)를 포함하고 있으나(이는 LLM이 유능한 오픈 정보 추출기라는 기존 연구 결과(Li et al., 2023)를 재확인함), 결과 스키마 내에는 상당한 수준의 중복이 존재한다. **EDC는 오픈 KG를 정확하게 정규화하여 CESI에 비해 더 간결하고 중복이 적은 스키마를 생성합니다**. EDC는 CESI의 과도한 일반화 경향을 피합니다. 기존 연구(Putri 외, 2019)와 마찬가지로, CESI가 '사망 장소', '출생 장소', '사망 날짜', '출생 날짜', '사망 원인'과 같은 다양한 관계를 부적절하게 단일 '사망 날짜' 범주로 클러스터링하는 것을 관찰했습니다.

5 결론

본 연구에서는 LLM 기반의 3단계 프레임워크인 EDC를 제시하여, 개방형 정보 추출과 사후 표준화를 통해 KGC 문제를 해결합니다. 실험 결과는

EDC와 EDC+R은 대상 스키마가 제공될 경우 전문적으로 훈련된 모델보다 우수한 지식 그래프(KG)를 추출할 수 있으며, 스키마가 제공되지 않을 경우 동적으로 스키마를 생성할 수 있습니다. EDC의 확장성과 다용도성은 다양한 응용 분야에 많은 기회를 열어줍니다: 위키데이터(Wikidata)와 같은 대규모 스키마(Vrandečić and Krötzsch, 2014)를 사용하여 일반 텍스트에서 고품질 KG를 자동으로 추출할 수 있을 뿐만 아니라, 새로 발견된 관계로 이러한 스키마를 보강할 수도 있습니다.

6 제한 사항 및 향후 연구 방향

향후 연구에서 해결하고자 하는 몇 가지 한계점이 있습니다.

- 본 논문에서는 스키마 정규화만을 고려하였으나, 향후 구축된 지식 그래프의 중복성을 줄이기 위해 엔티티 중복 제거 메커니즘을 통합하는 것이 매우 중요합니다. 예를 들어, 공동참조 해결(Sukthanker et al., 2020)을 통해 이를 구현할 수 있습니다. 본 연구에서는 이 접근법을 간략히 탐구하였으며, 예비 결과는 부록 F에서 확인할 수 있습니다.

- EDC의 구성 요소는 성능 향상을 위해 추가 개선이 가능합니다. 특히 스키마 검색기는 더 다양하고 고품질의 데이터로 훈련할 경우 성능 향상에 도움이 될 수 있습니다.

- 시간 및 자원 제약으로 인해 OIE 모듈에만 다양한 대규모 언어 모델(LLM)을 테스트했으며, EDC의 다른 모든 모듈은 GPT-3.5-turbo에 의존하고 있습니다. 따라서 소규모 오픈소스 모델의 성능을 다른 작업에서도 테스트하는 것이 유용할 것입니다.

- EDC는 다수의 LLM 호출을 포함하는 비용이 많이 드는 프레임워크입니다. 모든 구성 요소에 GPT-3.5-turbo를 사용할 경우 비용은 약
  실험에서 예시당 0.009 USD가 소요되었습니다. 특정 구성 요소를 더 작은 미세 조정 모델로 대체할 수 있습니다. 기존 연구에서는 더 작은 언어 모델을 OIE에 미세 조정할 수 있음을 보여주었으며(Wadhwa et al., 2023), 더 작은 BERT 기반 분류기를 스키마 정규화를 위해 훈련할 수 있음을 보여주었습니다. 또한 부록 G에서 OIE와 스키마 정의의 두 단계를 결합할 가능성도 탐구했습니다.

- 우리는 EDC를 구현형 AI 및 로봇 공학에 적용하고자 합니다. 구체적으로 KG는 VLM의 기억 소스를 형성할 수 있으며, 여기에는
  인간(Zhang and Soh, 2023), 작업 또는 목표(Xie et al., 2023), 환경에 관한 사실들을 포함합니다.

7 윤리적 고려 사항

아티팩트 사용. 본 논문에서 사용한 데이터셋은 연구 목적으로만 활용되었으며, 해당 라이선스(예: WebNLG는 cc-by-nc-sa-4.0 적용)를 엄격히 준수합니다. 작업의 특성상 데이터셋에는 개인(특히 유명인)에 대한 정보가 내재적으로 포함될 수 있음을 유의하시기 바랍니다. 본 논문의 소프트웨어 및 코드는 https://github.com/clear-nus/edc에서 공개적으로 이용 가능합니다.
//github.com/clear-nus/edc에서 공개적으로 이용 가능합니다.

인간 어노테이터. 두 명의 어노테이터(남성 1명, 여성 1명)는 모집된 대학생입니다. 어노테이터에게는 공정한 보상이 지급되었으며, 작업을 완료하기 위해 충분하고 유연한 시간이 제공되었습니다. 수집 프로토콜은 우리 기관의 IRB 위원회에 의해 면제 대상으로 결정되었습니다.

잠재적 위험. 현재의 대규모 언어 모델(LLM) 사용은 환각 현상(Xu et al., 2024) 및 개인정보 문제(Yao et al., 2024)와 같은 위험을 초래할 수 있습니다.

감사의 말

본 연구는 싱가포르 국가연구재단(National Research Foundation Singapore) 및 DSO 국립연구소(DSO National Laboratories)의 AI 싱가포르 프로그램(AISG Award No: AISG2-RP-2020-016)의 지원을 받았습니다.

참고문헌

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat 외. 2023. GPT-4 기술 보고서. *arXiv 사전 인쇄본 arXiv:2303.08774*.

오신 아가르왈, 헤밍 게, 시아막 샤케리, 라미 알-르푸. 2020. 지식 강화 언어 모델 사전 훈련을 위한 지식 그래프 기반 합성 코퍼스 생성. *arXiv 사전 인쇄본 arXiv:2010.12688*.

Zhen Bi, Jing Chen, Yinuo Jiang, Feiyu Xiong, Wei Guo, Huajun Chen, Ningyu Zhang. 2024. Codekgc: 생성적 지식 그래프 구축을 위한 코드 언어 모델. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 23(3):1–16.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell 외. 2020. 언어 모델은 소량 학습 학습자이다.

학습자이다. *신경 정보 처리 시스템의 진전*, 33:1877–1901.

페레-루이스 위게 카보트와 로베르토 나빌리. 2021. Rebel: 종단간 언어 생성을 통한 관계 추출. *계산언어학회 연구 성과: EMNLP 2021*, 2370–
2381.

최은솔, 오머 레비, 최예진, 루크 제틀-모이어. 2018. 초정밀 엔티티 타이핑. *arXiv 사전 인쇄본 arXiv:1807.04905*.

Sarthak Dash, Gaetano Rossiello, Nandana Mihin-dukulasooriya, Sugato Bagchi, Alfio Gliozzo. 2020. 변분 자동 인코더를 이용한 오픈 지식 그래프 정규화. *arXiv 사전 인쇄본 arXiv:2012.04780*.

Bayu Distiawan, Gerhard Weikum, Jianzhong Qi, and Rui Zhang. 2019. 지식 기반 확충을 위한 신경망 기반 관계 추출. *제57회 컴퓨터 언어학회 연차 총회 논문집*, 229–240쪽.

Pierre L Dognin, Inkit Padhi, Igor Melnyk, Payel Das. 2021. Regen: 사전 훈련된 언어 모델을 활용한 텍스트 및 지식 기반 생성을 위한 강화 학습. *arXiv 사전 인쇄본 arXiv:2108.12472*.

티아고 카스트로 페레이라, 클레어 가르당, 니콜라이 일리니흐, 크리스 반 데르 리, 시몬 밀레, 디에고 무살렘, 아나스타샤 시모리나. 2020. 2020년 양방향 이중 언어 웹NLG+ 공유 과제 개요 및 평가 결과 (webnlg+ 2020). *제3회 시맨틱 웹 기반 자연어 생성 국제 워크숍(WebNLG+) 논문집*.

*Deb*asis Ganguly, Dwaipayan Roy, Mandar Mitra, Gareth JF Jones. 2015. 정보 검색을 위한 단어 임베딩 기반 일반화 언어 모델. *제38회 국제 ACM SIGIR 정보 검색 연구 및 개발 컨퍼런스 논문집*, 795–798쪽.

*유*리 가닛케비치, 벤자민 반 더메, 크리스 캘리슨-버치. 2013. PPDB: 의역 데이터베이스. *북미계산언어학회 2013년 학술대회 논문집: 인간 언어 기술*, 758–764쪽.

Guo Liang, Yan Fu, Lu Yuqian, Zhou Ming, Yang Tao. 2021. 지식 그래프 기반 자동 가공 공정 의사 결정 시스템. *International journal of computer integrated manufacturing*, 34(12):1348–1369.

Qingyu Guo, Fuzhen Zhuang, Chuan Qin, Hengshu Zhu, Xing Xie, Hui Xiong, Qing He. 2020. 지식 그래프 기반 추천 시스템에 관한 연구. *IEEE Transactions on Knowledge and Data Engineering*, 34(8):3549–3568.

Harsha Gurulingappa, Abdul Mateen Rajput, Angus Roberts, Juliane Fluck, Martin Hofmann-Apitius, Luca Toldo. 2012. 의료 사례 보고서에서 약물 관련 부작용의 자동 추출을 지원하기 위한 벤치마크 코퍼스 개발. Journal of biomedical informatics, 45(5):885–892.

의학 사례 보고서에서 약물 관련 부작용 자동 추출 지원 코퍼스 개발. *Journal of biomedical informatics*, 45(5):885–892.

한리동, 펑타오, 양차오하오, 왕벤유, 류루, 완샹. 2023. ChatGPT로 정보 추출 문제가 해결되었는가? 성능, 평가 기준, 견고성 및 오류 분석. *arXiv 사전 인쇄본 arXiv:2305.14450*.

황샤오, 장징위안, 리딩청, 리핑. 2019. 지식 그래프 임베딩 기반 질문 답변. *제12회 ACM 국제 웹 검색 및 데이터 마이닝 컨퍼런스 논문집*, 105–113쪽.

지샤오슝, 판시루이, 에릭 캄브리아, 페카 마르티넨, 필립 유. 2021. 지식 그래프에 관한 개관: 표현, 획득 및 응용. *IEEE 신경망 및 학습 시스템 트랜잭션*, 33(2):494–514.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Men-sch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guil-laume Lample, Lucile Saulnier 외. 2023. Mistral 7b. *arXiv 사전 인쇄본 arXiv:2310.06825*.

마틴 조시포스키, 니콜라 드 카오, 막심 페야르, 파비오 페트로니, 로버트 웨스트. 2022. GenIE: 생성적 정보 추출. *2022년 북미계산언어학회 학술대회: 인간 언어 기술* 논문집, 4626–4643쪽, 미국 시애틀. 계산언어학회.

세라피나 캄프, 모르테자 파야지, 지네브 베나메르-엘, 슈얀 유, 로널드 드레스린스키. 2023. 오픈 정보 추출: 기본 기법, 접근법 및 응용에 대한 검토. *arXiv 사전 인쇄본 arXiv:2310.11644*.

케샤브 콜루루, 바이브하브 아드라카, 사마르트 아가르왈, 수멘 차크라바르티 외. 2020. Openie6: 오픈 정보 추출을 위한 반복적 그리드 라벨링 및 조정 분석. *arXiv 사전 인쇄본 arXiv:2010.03147*.

Luong Thi Hong Lan, Tran Manh Tuan, Tran Thi Ngan, Nguyen Long Giang, Vo Truong Nhu Ngoc, Pham Van Hai 외. 2020. 의사 결정에서의 퍼지 지식 그래프 및 확장을 갖춘 새로운 복합 퍼지 추론 시스템. *Ieee Access*, 8:164899–164921.

마이크 루이스, 인한 리우, 나만 고얄, 마르잔 가즈비니네자드, 압델라흐만 모하메드, 오메르 레비, 베스 스토야노프, 루크 제틀모이어. 2019. Bart: 자연어 생성, 번역 및 이해를 위한 시퀀스-투-시퀀스 사전 훈련의 노이즈 제거. *arXiv 사전 인쇄본 arXiv:1910.13461*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Hein-rich Küttler, Mike Lewis, Wen-tau Yih, Tim Rock-täschel 외. 2020. 지식 집약적 NLP 작업을 위한 검색 강화 생성. *신경 정보 처리 시스템의 진보*, 33:9459–9474.

보 리, 궈샹 팡, 양 양, 콴센 왕, 위 예, 원 자오, 시쿤 장. 2023. ChatGPT의 정보 추출 능력 평가: 성능, 설명 가능성, 보정 및 충실도 평가. *arXiv 사전 인쇄본 arXiv:2304.11633*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, Percy Liang. 2024. 중간에서 길을 잃다: 언어 모델이 긴 문맥을 활용하는 방식. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Pai Liu, Wenyang Gao, Wenjie Dong, Songfang Huang, and Yue Zhang. 2022. 2007년부터 2022년까지의 공개 정보 추출–서베이. *arXiv 사전 인쇄본 arXiv:2208.08690*.

이루안, 루헝 허, 마리 오스텐도르프, 한나네 하지시르지. 2018. 과학 지식 그래프 구축을 위한 명사체, 관계, 지시일치 다중 작업 식별. *arXiv 사전 인쇄본 arXiv:1808.09602*.

Pedro Henrique Martins, Zita Marinho, André FT Martins. 2019. 명명된 엔티티 인식과 엔티티 연결의 공동 학습. *arXiv 사전 인쇄본 arXiv:1907.08243*.

Igor Melnyk, Pierre Dognin, Payel Das. 2022. 텍스트로부터의 지식 그래프 생성. *arXiv 사전 인쇄본 arXiv:2211.10511*.

George A Miller. 1995. Wordnet: 영어를 위한 어휘 데이터베이스. *Communications of the ACM*, 38(11):39–41.

야스마사 오노에와 그렉 듀렛. 2020. 도메인 독립적 엔티티 연결을 위한 세분화된 엔티티 타이핑. *AAAI 인공 지능 학회 논문집*, 제34권, 8576–8583쪽.

Shon Otmazgin, Arie Cattan, and Yoav Goldberg. 2023. LingMess: 언어학적 정보를 활용한 다중 전문가 코어퍼런스 해결 점수 부여기. *제17회 유럽계 컴퓨터언어학회 학술대회 논문집*, 2752–2760쪽, 크로아티아 두브로브니크. 컴퓨터언어학회.

리프키 아피나 푸트리, 기원 홍, 성현 맹. 2019. 단어 임베딩 기반 시아미즈 네트워크를 이용한 오픈 IE 관계와 KB 관계 정렬. *제13회 국제 계산 의미론 학술대회-장문 논문집*, 142–153쪽.

콜린 라펠, 노암 샤지르, 애덤 로버츠, 캐서린 리, 샤란 나랑, 마이클 마테나, 옌치 저우, 웨이 리, 피터 J. 리우. 2020. 통합 텍스트-투-텍스트 트랜스포머를 통한 전이 학습의 한계 탐구. *기계 학습 연구 저널*, 21(140):1–67.

댄 로스와 웬타우 이. 2004. 자연어 처리 작업에서 전역 추론을 위한 선형 계획법 공식화. *제8회 계산 자연어 학습 학술대회(CoNLL-2004) 논문집*, HLT-NAACL 2004, 1–8쪽.

컴퓨테이셔널 자연어 학습에 관한 제8회 학술대회(CoNLL-2004) 논문집, HLT-NAACL 2004, 1–8쪽.

Alisa Smirnova and Philippe Cudré-Mauroux. 2018. 먼 감독을 이용한 관계 추출: 개요. *ACM 컴퓨팅 서베이 (CSUR)*, 51(5):1–35.

Rhea Sukthanker, Soujanya Poria, Erik Cambria, Ramkumar Thirunavukarasu. 2020. 대명사 및 공동참조 해결: 리뷰. *Information Fusion*, 59:139–162.

Qingyu Tan, Lu Xu, Lidong Bing, Hwee Tou Ng, Sharifah Mahani Aljunied. 2022. 관계 추출에서의 거짓 음성 문제 해결을 위한 docred 재검토. *EMNLP 회의록*.

Shikhar Vashishth, Prince Jain, and Partha Talukdar. 2018. Cesi: 임베딩과 부가 정보를 활용한 오픈 지식베이스 정규화. *2018년 월드 와이드 웹 컨퍼런스 논문집*, 1317–1327쪽.

Denny Vrandec˘ic´ 및 Markus Krötzsch. 2014. 위키데이터: 자유 협업 지식베이스. *ACM 커뮤니케이션*, 57(10):78–85.

Somin Wadhwa, Silvio Amir, Byron C Wallace. 2023. 대규모 언어 모델 시대의 관계 추출 재검토. *회의 논문집. Association for Computational Linguistics. Meeting*, volume 2023, page 15566. NIH Public Access.

홍 웨이 왕, 묘 자오, 싱 시에, 원지에 리, 민이 구오. 2019. 추천 시스템을 위한 지식 그래프 컨볼루션 네트워크. *세계 웹 컨퍼런스*, 3307–3313쪽.

량 왕, 난 양, 샤오룡 황, 린준 양, 랑간 마주머, 푸루 웨이. 2023. 대규모 언어 모델을 활용한 텍스트 임베딩 개선. *arXiv 사전 인쇄본 arXiv:2401.00368*.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, et al. 2023. ChatGPT와의 대화를 통한 제로샷 정보 추출. *arXiv 사전 인쇄본 arXiv:2302.10205*.

야치 시에, 천 위, 통야오 주, 진빈 바이, 제 공, 해럴드 소. 2023. 대규모 언어 모델을 활용한 자연어에서 계획 목표로의 변환. *arXiv 사전 인쇄본 arXiv:2302.05128*.

Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. 2024. 환각은 피할 수 없다: 대규모 언어 모델의 선천적 한계. *arXiv 사전 인쇄본 arXiv:2401.11817*.

야오 이판, 두안 진하오, 쉬 카이디, 차이 위안팡, 쑨 지보, 장 유에. 2024. 대규모 언어 모델(LLM) 보안 및 개인정보 보호에 관한 조사: 장점, 단점, 그리고 문제점. *High-Confidence Computing*, 100211쪽.

야스나가 미치히로, 렌 홍위, 보슬루 안투안, 리앙 퍼시, 레스코베츠 쥬레. 2021. Qa-gnn: 질문 답변을 위한 언어 모델과 지식 그래프를 활용한 추론. *arXiv 사전 인쇄본 arXiv:2104.06378*.

예홍빈, 장닝위, 천후이, 천화준. 2022. 생성적 지식 그래프 구축: 리뷰. *arXiv 사전 인쇄본 arXiv:2210.12714*.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceed-ings of the 2015 conference on empirical methods in natural language processing*, pages 1753–1762.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. 딥 컨볼루션 신경망을 통한 관계 분류. *제25회 국제 계산언어학 학술대회(COLING 2014) 논문집: 기술 논문*, 2335–2344쪽.

장보원, 소해럴드. 2023. 인간-로봇 상호작용을 위한 제로샷 인간 모델로서의 대규모 언어 모델. *2023 IEEE/RSJ 국제 지능형 로봇 및 시스템 학회(IROS)*, 7961–7968쪽. IEEE.

Lingfeng Zhong, Jia Wu, Qian Li, Hao Peng, and Xin-dong Wu. 2023. 자동 지식 그래프 구축에 관한 포괄적 서베이. *ACM 컴퓨팅 서베이*, 56(4):1–62.

저우 샤오원, 위 보원, 쑨 아이신, 롱 청, 리 징양, 위 하이양, 쑨 지안, 리 용빈. 2022. 신경망 기반 공개 정보 추출에 관한 연구: 현황과 향후 방향. *arXiv 사전 인쇄본 arXiv:2205.11725*.

안드레이 주코프-그레고리치, 요람 바크라흐, 샘 쿠프. 2018. 병렬 재귀 신경망을 이용한 명명된 개체 인식. *제56회 전산언어학회 연차대회 논문집 (제2권: 단편 논문)*, 69–74쪽.

A 구현 세부사항

A.1 모델 및 인프라 세부 사항

우리는 OpenAI의 두 모델인 GPT-3.5-turbo와 GPT-4(현재 크기 미상) 및 오픈소스 모델인 Mistral-7b(70억 매개변수)를 사용합니다. 오픈 소스 모델의 훈련 및 추론은 AMD EPYC 7543P 32코어 프로세서와 252GB RAM을 탑재하고 NVIDIA RTX A6000(48GB) GPU 4개를 장착한 단일 머신에서 수행되었습니다. GPT-3.5-turbo와 GPT-4는 OpenAI API를 통해 접근했습니다. EDC 코드는 https://github.com/clear-nus/edc에서 확인할 수 있습니다.

A.2 프롬프팅 관련 하이퍼파라미터

EDC의 모든 모듈에 대해 소량 데이터 프롬프팅을 사용하며, 각 데이터셋에서 경험적으로 6샷 예제를 선택합니다. 스키마 정규화 단계에서 사용되는 객관식 문제의 경우, 스키마에서 의미적으로 유사한 상위 5개 관계를 후보로 추출합니다. 정제 단계에서는 스키마 검색기가 스키마에서 가장 관련성이 높은 상위 10개 관계를 후보 관계로 추출합니다. 이러한 하이퍼파라미터는 성능과 추론 비용의 균형을 맞추기 위해 경험적으로 선택되었습니다.

A.3 스키마 검색기 훈련

우리는 원본 논문(Wang et al., 2023)에 상세히 기술된 원래 훈련 방법론을 따릅니다. 이는 텍스트 쌍과 그에 대응하는 정의된 관계를 활용하는 것을 포함합니다. 주어진 긍정적 텍스트-관계 쌍(t^+, r^+)에 대해, t^+에 대한 지시 템플릿을 사용하여 새로운 텍스트 t^+를 생성합니다 $= _{inst}$ "지시: 주어진 텍스트에 존재하는 관계를 추출하라 \n 질의: {t^+}".

그런 다음 InfoNCE 손실을 사용하여 임베딩 모델을 미세 조정하여 주어진 텍스트와 관련된 올바른 관계와 다른 관련 없는 관계를 구별합니다.

$$\min L = -\log \frac{\phi(t^+_{inst}, r^+)}{\phi(t^+_{inst}, r^+) + \sum_{n_i \in N} \phi(t_{inst}, n_i)}$$

여기서 N은 음성 샘플 집합을 나타내며, ϕ는 코사인 유사도 함수를 의미합니다. 추가적인 훈련 세부 사항은 부록을 참조하십시오.

훈련을 위해 위키데이터 삼중항을 위키백과 텍스트에 정렬하여 생성된 대규모 텍스트 삼중항 데이터셋인 TEKGEN 데이터셋(Agarwal et al., 2020)을 활용하여 텍스트-관계 쌍 데이터셋을 합성했습니다. 훈련 데이터셋은 양성 샘플과 음성 샘플로 균등하게 분할된 37,500개의 쌍으로 구성되었습니다.

온라인 오픈소스 구현체와 하이퍼파라미터 구성을 훈련에 채택했습니다.

정밀 조정된 스키마 검색기의 성능은 WebNLG, REBEL, Wiki-NRE 데이터셋의 테스트 분할에서 평가되었습니다. 해당 데이터셋에서의 recall@10 점수는 각각 0.823, 0.663, 0.818로, 다양한 지식 그래프 환경에서 검색기의 효과성을 입증합니다.

A.4 정제 힌트 세부 사항

정제 힌트는 후보 엔티티와 후보 관계로 구성됩니다. 이 섹션에서는 이들의 획득 방법과 OIE 성능 향상을 위해 어떻게 사용되는지 자세히 설명합니다. 3절에서 사용한 예시를 계속 사용하겠습니다: "*Alan Shepard는 1923년 11월 18일에 태어났고 1959년에 NASA에 선발되었습니다. 그는 아폴로 14호 승무원이었다*"이며, EDC가 첫 번째 반복에서 추출한 삼중항은 ['Alan Shepard', 'birthDate', 'Nov 18, 1923'], ['Alan Shepard', 'mission', 'Apollo 14']이다.

A.4.1 후보 엔티티 획득

후보 엔티티는 두 가지 출처에서 비롯됩니다:

- 이전 반복에서 EDC가 추출한 엔티티, 즉 ['Alan Shepard', 'Nov 18, 1923', 'Apollo 14']
- LLM에 엔티티 추출 작업을 수행하도록 프롬프트하여 텍스트에서 추출한 엔티티. 트리플릿 추출 작업과 유사함.

엔티티 추출 프롬프트

주어진 텍스트에서 엔티티 목록을 추출하세요.
다음은 몇 가지 예시입니다:
예시 1:
텍스트: 길이 17068.8 밀리미터의 ALCO RS-3는 디젤-전기 변속기를 장착하고 있다.
엔티티: ['ALCO RS-3', '디젤-전기 변속기', '17068.8 (밀리미터)']
...
이제 다음 텍스트에서 엔티티를 추출해 주세요: 앨런 셰퍼드는 1923년 11월 18일에 태어났으며 1959년 NASA에 선발되었습니다. 그는 아폴로 14호 승무원 중 한 명이었습니다.

그리고 결과 엔티티는 ['앨런 셰퍼드', '1923년 11월 18일', 'NASA', '1959년', '아폴로 14']

그런 다음 엔티티들을 후보 엔티티로 병합합니다.

A.4.2 후보 관계 획득

후보 관계도 두 가지 출처에서 비롯됩니다:

- 이전 반복에서 EDC가 추출한 관계, 즉 ['birthDate', 'mission']
- 스키마 검색기가 입력 텍스트와 스키마 내 관계 간의 관련성 점수를 계산하여 추출한 관계. 이 경우 상위 5개 검색 관계는 ['birthDate', , 'selectedByNasa', 'mission', 'draftPick', 'occupation']입니다.

이후 관계와 해당 정의가 후보 관계로 통합됩니다. 다른 RAG 기반 방법과 마찬가지로 검색기가 관련 없는 정보를 추출할 가능성이 있다는 점에 유의해야 합니다. 이 경우 관계 정의가 유용하게 활용될 수 있는데, 이는 LLM이 해당 관계가 텍스트에 대해 유효한지 여부를 판단하는 데 더 많은 정보를 제공하기 때문입니다.

A.4.3 정제된 OIE를 위한 힌트 사용법

정제된 OIE에 대한 힌트는 프롬프트에 적절히 포함되어 LLM이 후보 엔티티 및 후보 관계를 고려하도록 지시합니다(이에 국한되지 않음):

정제된 OIE 프롬프트

주어진 텍스트에서 [주체, 관계, 객체] 형식의 관계 삼중항을 추출하십시오. 예시는 다음과 같습니다:
예시 1:
텍스트: 길이 17068.8 밀리미터의 ALCO RS-3는 디젤-전기 변속기를 장착하고 있습니다.
엔티티: ['ALCO RS-3', '디젤-전기 변속기', '17068.8 (밀리미터)']
삼원조: [['ALCO RS-3', 'powerType', '디젤-전기 변속기'], ['ALCO RS-3', 'length', '17068.8 (밀리미터)']]
...
이제 다음 텍스트에서 트리플렛을 추출해 주세요: 앨런 셰퍼드는 1923년 11월 18일에 태어났으며 1959년 NASA에 선발되었습니다. 그는 아폴로 14호 승무원 중 한 명이었습니다. 엔티티: ['앨런 셰퍼드', '1923년 11월 18일', 'NASA', '1959년', '아폴로 14호']
추출 과정에서 확인할 수 있는 잠재적 관계와 그 설명은 다음과 같습니다:
1. birthDate: 주체 엔티티는 객체 엔티티가 지정한 날짜에 태어났습니다.
2. mission: 대상 개체가 명시한 사건 또는 작전에 주체 개체가 참여했습니다.
3. selectedByNasa: 대상 개체가 지정한 연도에 NASA에 의해 선택되었습니다.
...

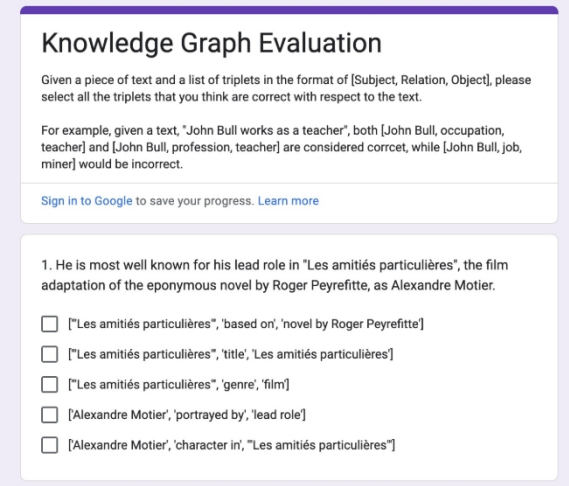

그림 3: 주석 작업자에게 제공된 지침을 포함한 설문지의 예시 스크린샷.

정제된 OIE로 추출된 삼중항은 다음과 같습니다: ['앨런 셰퍼드', '생년월일', '1923년 11월 18일'], ['앨런 셰퍼드', '임무', '아폴로 14호'], ['앨런 셰퍼드', 'NASA 선발', '1959년']. 이는 힌트 사용 없이는 놓쳤을 미묘하고 세분화된 관계 'selectedByNasa'를 성공적으로 복원합니다. 또한 의미론적으로 풍부한 설명은 LLM이 스키마 검색기가 추출한 잡음이 많은 관계를 과도하게 추출하는 것을 방지하는 데 도움이 됩니다.

우리는 두 출처의 엔티티, 즉 마지막 라운드에서 추출된 것과 별도의 모듈(엔티티 추출 또는 스키마 리트리버)에 의해 발견된 엔티티를 모두 포함하는 것이 중요하다는 것을 발견했습니다. 스키마 리트리버의 중요성은 이미 4.2.1절의 제거 연구에서 보여졌습니다.

B 주석 지침

주석 작업자에게 제공된 설문지 및 지침의 형식을 설명하기 위해 그림 3에 예시 스크린샷을 제공합니다. 데이터 수집 목적은 주석 작업자에게 구두로 전달되었습니다.

C 대상 정렬의 상세 결과

C.1 완전한 결과

WebNLG, REBEL 및 Wiki-NRE에 대한 EDC와 EDC+R의 완전한 결과는 각각 표 3, 표 4 및 표 5에 요약되어 있습니다. EDC는 모든 기준(부분, 엄격, 정확)에서 모든 지표(정밀도, 재현율, F1) 측면에서 최첨단 기준 모델보다 우수하거나 비슷한 성능을 보이며, EDC+R은 일관되게 성능을 향상시킬 수 있습니다.

표 3: WebNLG 데이터셋에서 EDC 및 EDC+R의 완전한 결과(기준 모델 REGEN 대비, '부분적', '엄격한', '정확한' 기준에 따른 정밀도, 재현율, F1 점수). EDC+R은 1회의 정제만 수행함. 최상의 결과는 굵게 표시됨.

방법	OIE용 LLM	부분			엄격			정확		
		정밀도	리콜	F1	정밀도	리콜	F1	정밀도	리콜	F1
EDC	GPT-4	**0.776**	**0.796**	**0.783**	**0.729**	**0.741**	**0.733**	**0.751**	**0.765**	**0.756**
	GPT-3.5	0.739	0.760	0.746	0.684	0.697	0.688	0.708	0.722	0.713
	미스트랄-7b	0.723	0.739	0.728	0.668	0.679	0.672	0.692	0.703	0.696
EDC+R	GPT-4	**0.814**	**0.831**	**0.820**	**0.782**	**0.794**	**0.786**	**0.796**	**0.808**	**0.800**
	GPT-3.5	0.788	0.806	0.794	0.749	0.761	0.753	0.768	0.781	0.772
	미스트랄-7b	0.756	0.775	0.762	0.716	0.727	0.720	0.735	0.747	0.739
기준선	재생	0.755	0.788	0.767	0.713	0.735	0.720	0.714	0.738	0.723

표 4: REBEL 데이터셋에서 기준 모델 REGEN 대비 EDC 및 EDC+R의 완전한 결과 (‘부분적’, ‘엄격한’, ‘정확한’ 기준에 따른 정밀도, 재현율, F1 점수). EDC+R은 정제 과정을 1회만 수행합니다. 최상의 결과는 굵게 표시했습니다.

방법	OIE용 LLM	부분			엄격			정확		
		정밀도	리콜	F1	정밀도	리콜	F1	정밀도	리콜	F1
EDC	GPT-4	**0.543**	**0.552**	**0.546**	**0.498**	**0.503**	**0.500**	**0.511**	**0.517**	**0.514**
	GPT-3.5	0.503	0.512	0.506	0.448	0.453	0.449	0.471	0.476	0.473
	미스트랄-7b	0.512	0.523	0.516	0.450	0.457	0.453	0.481	0.488	0.483
EDC+R	GPT-4	**0.599**	**0.606**	**0.601**	**0.557**	**0.561**	**0.559**	**0.572**	**0.576**	**0.574**
	GPT-3.5	0.556	0.565	0.559	0.513	0.519	0.516	0.527	0.533	0.529
	미스트랄-7b	0.525	0.550	0.531	0.461	0.462	0.462	0.506	0.511	0.505
기준선	GENIE	0.381	0.391	0.385	0.353	0.361	0.356	0.362	0.369	0.364

표 5: Wiki-NRE 데이터셋에서 기준 모델 REGEN 대비 EDC 및 EDC+R의 전체 결과 (‘부분적’, ‘엄격한’, ‘정확한’ 기준에 따른 정밀도, 재현율, F1 점수). EDC+R은 1회의 정제만 수행합니다. 최상의 결과는 굵게 표시했습니다.

방법	OIE용 LLM	부분			엄격			정확		
		정밀도	리콜	F1	정밀도	리콜	F1	정밀도	리콜	F1
EDC	GPT-4	**0.682**	**0.686**	**0.683**	**0.675**	**0.679**	**0.677**	**0.676**	**0.680**	**0.678**
	GPT-3.5	0.645	0.651	0.647	0.636	0.640	0.638	0.638	0.643	0.640
	미스트랄-7b	0.644	0.650	0.647	0.636	0.640	0.637	0.637	0.641	0.639
EDC+R	GPT-4	**0.712**	**0.715**	**0.713**	**0.708**	**0.710**	**0.709**	**0.708**	**0.711**	**0.709**
	GPT-3.5	0.691	0.696	0.693	0.684	0.688	0.685	0.685	0.689	0.687
	미스트랄-7b	0.661	0.667	0.663	0.647	0.652	0.649	0.656	0.661	0.658
기준선	GENIE	0.482	0.486	0.484	0.462	0.464	0.463	0.477	0.479	0.478

표 6: 추가 반복 정제 결과(모든 기준에 따른 F1 점수). OIE에 사용된 대규모 언어 모델(LLM)은 GPT-3.5-turbo입니다. EDC+2xR은 2회 정제 반복을 거친 EDC입니다.

방법	WebNLG			REBEL			위키-NRE		
	부분	엄격	정확	부분	엄격	정확	부분	엄격	정확
EDC+2xR	0.797	0.761	0.775	0.564	0.521	0.535	0.697	0.689	0.660
EDC+R	0.794	0.753	0.772	0.559	0.516	0.529	0.693	0.685	0.657
EDC	0.746	0.688	0.713	0.506	0.449	0.473	0.644	0.634	0.637

표 7: 정제 힌트에서 마지막 라운드에서 추출된 엔티티와 관계를 제거한 결과(모든 기준에 대한 F1 점수). OIE에 사용된 LLM은 GPT-3.5-turbo입니다. EDC+R-lastround는 정제를 거친 EDC이지만, 마지막 라운드에서 추출된 엔티티와 관계는 정제 힌트에서 제거됩니다.

방법	WebNLG			REBEL			Wiki-NRE		
	부분	엄격	정확	부분	엄격	정확	부분	엄격	정확
EDC+R	0.794	0.753	0.772	0.559	0.516	0.529	0.693	0.685	0.657
EDC+R-마지막 라운드	0.748	0.698	0.720	0.534	0.485	0.505	0.634	0.622	0.625
EDC	0.746	0.688	0.713	0.506	0.449	0.473	0.644	0.634	0.637

표 8: 세 데이터셋 전체에서 문장당 추출된 삼중어 평균 개수. WebNLG의 기준 모델은 REGEN이며, Rebel과 Wiki-NRE의 기준 모델은 GENIE이다. 괄호 안 숫자는 참조 주석과의 차이이다.

OIE용 LLM	WebNLG	REBEL	Wiki-NRE
GPT-4	3.47(+0.04)	5.11(+1.11)	3.49(+0.63)
GPT-3.5	3.44(+0.01)	5.01(+1.01)	3.49(+0.63)
미스트랄7b	3.45+(0.02)	4.68(+0.68)	3.75(+0.89)
기준선	-	2.20(-1.80)	3.08(+0.22)
참조	3.43	4.00	2.86

이 모든 측면에서도 이를 입증합니다. 이러한 결과는 EDC 및 EDC+R의 성능을 보다 포괄적으로 보여줍니다.

C.2 추가 정제 반복의 효과

표 6은 모든 데이터셋에 대해 EDC를 사용한 추가 정제 반복의 결과를 보여줍니다. 추가 정제는 결과를 안정적으로 개선하지만, 수익 감소 현상이 관찰되므로 주요 결과에는 한 번의 반복만 보고합니다.

C.3 최종 추출 단계에 대한 제거 연구

표 7은 정제 힌트에서 마지막 라운드 추출물에서 관계와 엔티티를 제거한 결과를 보여줍니다. 이는 정제를 반복적으로 수행하는 것의 중요성을 보여줍니다. 두 소스를 병합함으로써 텍스트 내 엔티티와 관계의 커버리지가 향상되어 더 나은 KGC를 얻었습니다.

C.4 KGC 데이터셋 어노테이션에 관한 논의

4.2절에서 언급한 바와 같이, EDC는 불완전한 주석으로 인해 Rebel 및 Wiki-NRE 데이터셋에서 평가기(scorer)에 의해 불이익을 받는 것으로 관찰됩니다. 이는 (Wadhwa et al., 2023; Han et al., 2023)의 이전 연구 결과와 일치하는데, 대규모 언어 모델(LLMs)이 주석에 누락된 정확한 결과를 종종 추출할 수 있어 지나치게 비관적인 평가를 초래한다는 점입니다. 표 8에서 보듯이, EDC는 참조 주석 및 기준 모델인 GenIE에 비해 현저히 더 많은 삼중항을 추출하는 경향이 있습니다. 또한 표 2의 수동 평가에서 알 수 있듯이, 이러한 삼중항 중 상당수는 입력 텍스트에 대해 실제로 의미 있고 정확한 것입니다. 그럼에도 불구하고, EDC에 대한 자동 평가 결과가 지나치게 비관적임에도 불구하고, 여전히 기준선을 크게 상회하며 추출된 삼중항 수의 차이를 고려하면 실제 성능은 더 클 수 있습니다.

D 새로운 데이터셋에 대한 실험

테스트된 데이터셋은 수년 전에 생성되었으며 사용된 대규모 언어 모델(LLM)의 훈련 세트가 알려지지 않았기 때문에, 해당 LLM들이 이미 이 데이터셋으로 훈련되었을 위험이 존재합니다. 이러한 우려를 해결하기 위해, 우리는 가상의 개체와 정보로 구성된 새로운 소규모 데이터셋(50개 항목)을 생성했습니다. 예를 들어, "*에버그린 대학교는 에밀리 존슨이 생물학 학위를 취득한 곳이다*"와 같은 내용이며, 이를 Wiki-NRE 스키마를 사용하여 주석 처리했습니다. 표 9는 EDC 및 EDC+R이 여전히 기준 모델인 GenIE보다 우수한 성능을 보임을 보여줍니다.

E 기존 LLM 기반 접근법과의 비교

비록 이는 EDC의 의도된 사용 시나리오가 아니지만, 기존 LLM 기반 방법들과 비교하기 위한 보다 포괄적인 평가를 위해 이러한 실험 결과를 포함합니다. 우리는 세 가지 데이터 세트, CoNLL04 (4가지 관계 유형) (Roth and Yih, 2004), SciERC (7가지 관계 유형) (Luan et al., 2018) 및 Wiki-NRE (45가지 관계 유형)의 하위 샘플링 버전을 사용하여 실험을 수행합니다. 비교의 공정성을 보장하기 위해, 비교 대상 방법 모두에 GPT-3.5-turbo를 사용합니다.

표 10에서 볼 수 있듯이, 관계 수가 적을 때(CONLL 및 SciERC) EDC 단독은 프롬프트에서 스키마를 제외했기 때문에 기준 방법보다 우수하지 않을 수 있습니다. 그러나 정제 과정을 통해 EDC+R은 훨씬 더 나은 결과를 달성할 수 있습니다. 이는 정제 단계에서 의미적으로 풍부한 관계 설명을 사용했기 때문일 수 있습니다. 구체적으로, 추출 과정에서 발생할 수 있는 두 가지 유형의 오류를 수정하는 데 도움이 됩니다: 1. 정의 단계는 동음이의어를 명확히 하는 데 도움이 됩니다. 예를 들어, "John follows Taoism"에서 "follows"라는 단어는 두 가지 다른 의미를 가집니다.

v.s. "John follows Mary". EDC는 "John follows Taoism"에서 "fol-lows"를 "adheres to"로 변경합니다. 2.

표 9: 새로운 허구 데이터셋에서 EDC 및 EDC+R의 완전한 결과와 기준 모델 GenIE 비교 (정밀도, 재현율, '부분적', '엄격한', '정확한' 기준에 따른 F1 점수). EDC+R은 1회의 정제만 수행함. 최상의 결과는 굵게 표시됨. OIE에 사용된 LLM은 GPT-3.5-turbo임.

	부분			엄격			정확		
방법	정밀도	리콜	F1	정밀도	리콜	F1	정밀도	리콜	F1
EDC	0.731	0.771	0.751	0.687	0.704	0.691	0.702	0.720	0.707
EDC+R	0.761	0.782	0.767	0.733	0.750	0.738	0.733	0.750	0.738
GenIE	0.521	0.547	0.530	0.426	0.443	0.432	0.467	0.483	0.472

표 10: CONLL, SciERC 및 Wiki-NRE 데이터셋에서 EDC, EDC+R의 완전한 결과와 기존 LLM 기반 접근법인 CodeKGC 및 ChatIE와의 비교. 비교의 공정성을 위해 여기에서 사용된 LLM은 GPT-3.5-turbo입니다. 최상의 결과는 굵게 표시했습니다.

		부분			엄격			정확		
데이터셋	방법	정밀도	리콜	F1	정밀도	리콜	F1	정밀도	리콜	F1
CONLL	EDC	0.536	0.552	0.543	0.481	0.491	0.485	0.503	0.515	0.509
	EDC+R	**0.580**	**0.593**	**0.585**	**0.514**	**0.522**	**0.517**	**0.549**	**0.558**	**0.552**
	코드KGC	0.542	0.55	0.545	0.503	0.506	0.504	0.542	0.546	0.543
	ChatIE	0.463	0.477	0.468	0.360	0.366	0.363	0.418	0.427	0.421
SciERC	EDC	0.389	0.408	0.395	0.288	0.301	0.292	0.352	0.365	0.357
	EDC+R	**0.447**	**0.461**	**0.451**	**0.340**	**0.349**	**0.343**	**0.406**	**0.416**	**0.410**
	코드KGC	0.389	0.398	0.392	0.277	0.283	0.279	0.346	0.353	0.349
	ChatIE	0.351	0.367	0.357	0.212	0.221	0.215	0.294	0.302	0.297
Wiki-NRE	EDC	0.645	0.651	0.647	0.636	0.640	0.638	0.638	0.643	0.640
	EDC+R	**0.691**	**0.696**	**0.693**	**0.684**	**0.688**	**0.685**	**0.685**	**0.689**	**0.687**
	코드KGC	0.611	0.614	0.612	0.605	0.607	0.606	0.607	0.609	0.608
	ChatIE	0.569	0.574	0.571	0.541	0.545	0.543	0.553	0.557	0.555

관계 정의를 활용하여, 정제 단계가 머리-꼬리 관계 오류를 수정함을 확인했습니다. 예를 들어 "아버지" 관계의 경우 주어와 목적어 중 누가 아버지인지 불분명한데, 이 정의는 일관성 없는 사용을 방지합니다. 이러한 오류 수정 효과는 기존 방법에서는 불가능했습니다.

중간 규모의 스키마를 가진 Wiki-NRE에서 테스트했을 때, EDC는 이미 기준 방법들을 크게 능가하며, 이는 아마도 긴 문맥을 처리할 때 LLM의 혼란 때문일 수 있습니다(Liu et al., 2024). 또한 ChatIE와 CodeKGC는 프롬프트에 전체 스키마가 제공되었음에도 스키마 외 관계어를 출력할 수 있음을 관찰했으며, 이는 기존 연구 결과(Wadhwa et al., 2023)를 반영합니다.

F EDC를 다른 IE 도구와 결합하기

EDC는 청크화, 공동참조, 중복 엔티티 제거 등 다른 정보 추출 도구와 통합될 수 있습니다. 이는 LLM의 컨텍스트 창 길이를 초과하는 긴 문서를 처리하는 등의 시나리오에서 유용합니다. 우리는 EDC를 최첨단 지시 참조 해결 방법인 LingMess(Otmazgin et al., 2023) 및 간단한 문장 수준 청킹과 결합하여 Re-DOCRED(Tan et al., 2022)에 대한 실험을 수행했습니다. 엄격한 마이크로 F1 점수가 0.132에서 0.234로 증가하는 것을 관찰했으며,

LLM에 직접 프롬프트를 제공했을 때는 0.060에 불과했습니다. 또한 EDC와 결합한 엔티티 중복 제거의 효과도 탐구했습니다. EDC로 생성된 지식 그래프(KG) 내 엔티티 중복 제거를 위해 최첨단 사후 정규화 방법인 CESI(Vashishth et al., 2018)를 적용했습니다. 그 결과 REBEL 데이터셋에서 '부분적' 기준 하에 F1 점수가 0.516에서 0.520으로 소폭 향상된 것을 관찰했습니다. '부분적' 기준 하에서 REBEL 데이터셋에서 F1 점수가 0.516에서 0.520으로 소폭 개선된 것을 확인했습니다.

G OIE와 스키마 정의 결합

EDC 비용을 절감하기 위한 시도로서, 우리는 OIE 단계와 스키마 정의 단계를 결합하는 방안을 탐구했습니다. 이전에는 예비 실험에서 OIE가 더 어려운 작업으로 나타났고, 두 하위 작업을 분리함으로써 OIE에는 더 고성능 모델을, 스키마 정의에는 더 작고 저렴한 모델을 사용할 수 있었기 때문에 이 두 단계를 분리했습니다. 그러나 별도의 LLM 호출은 파이프라인의 지연 시간(동일 LLM 사용 시 비용도)을 증가시킵니다. 또한 추출된 트리플과 함께 정의를 LLM이 출력하도록 하면 일관성이 향상될 수 있습니다. GPT-3.5-turbo를 사용해 REBEL에서 EDC와 스키마 정의를 결합한 추가 실험에서, 우리는 성능이 약간 향상되었고('부분적' 기준 하에서 0.516에서 0.518로) 토큰 비용이 감소한 것을 관찰했습니다(예시당 *약* 3k에서 2k 토큰).