JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 29 No. 9, pp. 39-48, September 2024
https://doi.org/10.9708/jksci.2024.29.09.039

Research on improving KGQA efficiency using self-enhancement of reasoning paths based on Large Language Models

Min-Ji Seo*, Myung-Ho Kim**

*Student, Dept. of Convergence Software, Soongsil University, Seoul, Korea
**Professor, Dept. of Software, Soongsil University, Seoul, Korea

[Abstract]

In this study, we propose a method to augment the provided reasoning paths to improve the answer performance and explanatory power of KGQA. In the proposed method, we utilize LLMs and GNNs to retrieve reasoning paths related to the question from the knowledge graph and evaluate reasoning paths. Then, we retrieve the external information related to the question and then converted into triples to answer the question and explain the reason. Our method evaluates the reasoning path by checking inference results and semantically by itself. In addition, we find related texts to the question based on their similarity and converting them into triples of knowledge graph. We evaluated the performance of the proposed method using the WebQuestion Semantic Parsing dataset, and found that it provides correct answers with higher accuracy and more questions with explanations than the reasoning paths by the previous research.

▸ **Key words**: Knowledge Graph Question Answering, Self-evaluation, Large Language Models

[요 약]

본 연구에서는 KGQA의 답변 성능 및 설명력을 높이기 위해 제공된 추론 경로를 스스로 평가하고 보강하는 방법을 제안한다. 제안하는 방법에서는 LLMs와 GNN을 활용하여 질의와 관련된 추론 경로를 지식 그래프에서 검색하였다. 검색된 추론 경로를 LLMs가 자가적으로 평가하여 보완이 필요하다고 판단될 경우, 질문과 관련된 외부 정보를 찾고 트리플로 변환하여 지식 그래프에 추가하였다. 이에 따라 LLMs가 보강된 트리플 셋을 통해 정답과 이유를 설명할 수 있도록 하였다. 추론 경로는 추론 결과 혹은 경로가 의미상으로 질문과 관계가 있는지 LLMs 스스로 평가하도록 하였으며, 텍스트 유사도를 통해 질문과 관련된 텍스트를 찾아내어 추론 경로를 보강하여 LLMs가 기존보다 정확하게 질문에 대한 정답을 설명할 수 있도록 하였다. WebQuestion Semantic Parsing 데이터셋을 이용하여 제안 방법의 성능을 평가한 결과, 기존 방법으로 생성한 추론 경로보다 높은 정확도로 정답을 제공하고 더 많은 종류의 질문에 설명을 출력하는 것을 증명하였다.

▸ **주제어**: 지식 그래프 기반 질의응답, 자가 평가, 거대 언어 모델

• First Author: Min-Ji Seo, Corresponding Author: Myung-Ho Kim
*Min-Ji Seo (porito2@soongsil.ac.kr), Dept. of Convergence Software, Soongsil University
**Myung-Ho Kim (kmh@ssu.ac.kr), Dept. of Software, Soongsil University
• Received: 2024. 08. 07, Revised: 2024. 08. 22, Accepted: 2024. 08. 27.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

최근 거대 언어 모델(large language models, LLMs)은 번역(translation), 요약(summarization), 질의응답(question answering)과 같은 다양한 분야의 자연어 처리 작업에서 뛰어난 성과를 보여주고 있다[1,2]. 하지만 LLMs는 학습 데이터에 없는 도메인 특화 및 새로운 도메인에 대한 정보를 검색할 경우 불명확하거나 사실이 아닌 조작된 정보를 답변하는 환각(hallucination) 문제를 가진다. 이에 따라 외부의 광범위한 문서 집합에서 관련 정보를 검색하여 도메인별 상황에 맞는 응답을 생성하는 검색 증강 생성(retrieval-augmented generation, RAG) 기술이 제안되고 있다[3]. 질문에 관련된 정보를 검색하기 위한 외부의 데이터 소스로는 지식 그래프(knowledge graph), 검색 엔진, 기업 내부 문서 및 데이터베이스 등의 정보를 활용한다.

지식 그래프는 막대한 양의 정보를 구조화된 표현인 트리플(triplet) 형식으로 저장하는 그래프 데이터베이스를 의미한다[4]. 지식 그래프는 단계적으로 트리플을 검색한 추론 경로(reasoning path)를 구축하여 필요한 정보를 검색하거나 새로운 지식을 확장할 수 있기 때문에, 지식 그래프 기반의 질의응답(knowledge graph question answering, KGQA)과 같은 정보 집약적인(knowledge-intensive) 과제에 주로 활용된다[5,6].

KGQA는 입력된 자연어 질의에 대하여 지식 그래프에 내재된 정보를 바탕으로 적절한 대답을 찾는 것을 목표로 한다. 최근에는 넓은 분야의 지식을 가진 LLMs를 활용하여 KGQA를 해결하려는 연구가 진행되고 있다. LLMs를 활용한 KGQA 연구는 LLMs의 문장 이해력을 지식 그래프를 단계적으로 검색한 추론 경로에 적용하여, 질문에 대한 정답률을 높이고 질문에서 정답까지 도달하는 과정을 그래프 추론 경로를 참고하여 설명한다. 추론 경로는 지식 그래프에서 탐색한 정답에 도달하는 트리플 경로를 자연어 문장으로 나타내어 LLMs에 입력되어, LLMs가 입력된 추론 경로를 토대로 정답을 질문자가 이해할 수 있는 자연어의 형태로 출력한다[7,8]. 이 경우 LLMs는 주어진 경로를 참고하여 질문에 대한 대답을 출력하기 때문에, 정답의 정확도와 설명력은 추론 경로가 가진 정보량에 따라 영향을 받게 된다. 따라서 질문과 관련된 유효한 추론 경로를 지식 그래프에서 검색하기 위해, 지식 그래프 안에서 질문 키워드에 대응되는 객체를 찾는 entity linking 작업이 중요하다고 할 수 있다[9]. 하지만 지식 그래프에 속하는 개체를 인지할 수 없거나, 제대로 그래프 검색이 수행되지 않는 경우 entity linking 성능 저하와 함께 정답 경로를 단계적으로 추론하지 못해 정답률 저하로 이어지게 된다.

따라서 본 논문에서는 LLMs 기반 KGQA 성능 향상을 위해 LLMs가 제공된 추론 경로의 유효성을 판단하고 필요에 따라 지식 그래프를 보강하는 방법을 제안한다. 제안하는 방법은 LLMs의 자가 평가(self-evaluation)를 통해 초기에 제공된 추론 경로에 문제가 없는지 검사한다. 이후 추론 경로 보강이 필요한 데이터에 대하여, 질문과 가장 관련 있는 웹 페이지의 구문을 찾아 구문의 내용을 바탕으로 트리플을 생성한다. 추론 경로에 불필요한 트리플이 포함되는 경우 나타날 수 있는 LLMs 추론의 신뢰성 저하를 방지하기 위해, 질문의 정답과 관련 있는 트리플을 선택하여 추론 경로를 연결하도록 하여 LLMs에 제공한다. 제안하는 방법은 외부 정보를 활용하여 지식 그래프를 보강하였기 때문에 보강 과정에서 추가적인 LLMs 파인튜닝(fine-tuning) 학습에 대한 부담이 없다. 추가적으로, 질문의 정답과 가장 관련 있는 추론 경로 목록을 보강함으로써 LLMs가 더욱 효율적으로 질문에 대한 답과 추론 과정을 사용자에게 제공할 수 있다.

본 연구를 통해 제공한 추론 경로의 성능은 현재 대표적인 KGQA 데이터셋인 WebQSP(webquestion semantic parse) 데이터셋[10]을 이용하여 평가하였다. 평가는 기존에 연구된 방법으로 생성된 추론 경로들과 본 연구에서 생성한 추론 경로들이 주어졌을 때, LLMs가 대답을 제대로 도출할 수 있는지, 또는 추론 경로를 이용하여 대답의 설명이 가능한지 여부를 평가한다.

본 논문의 구성은 다음과 같다. 2장에서는 지식 그래프, KGQA 및 지식 그래프와 LLMs를 이용하여 KGQA 문제를 해결한 연구에 대하여 소개한다. 3장에서는 제안하는 연구의 방법론에 대해 소개한다. 4장에서는 실험을 통해 제안하는 방법의 성능을 평가하고, 5장에서는 결론을 내린다.

II. Preliminaries

1. Related works

1.1 Knowledge graph question answering

지식 그래프는 현실 세계의 다양한 사실(Facts)을 트리플의 형태로 포함하는 이형 그래프(heterogeneous graph)를 의미한다. 트리플은 특정한 사실의 주체가 되는 엔티티(subject entity)가 다른 엔티티(object entity)와 관계(relation)를 가진 사실(s,r,o)을 의미한다. 지식 그래프는 다수의 트리플로 구성된 그래프를 의미하며,

$G = \{(s,r,o) | s,o \in E, r \in R\}$로 표현할 수 있다. 이 때 E는 엔티티 집합을, R은 엔티티 간 관계 집합을 나타낸다.

KGQA는 자연어 질문에 대하여 지식 그래프로 구성된 트리플을 검색하여 정답이 되는 엔티티를 찾아내기 위한 분야로, 전통적으로 두 가지 방법이 연구되었다. Information Retrieval(IR) 기반 방법[11]은 언어 모델을 통해 질문을 임베딩하여 관련 있는 트리플이나 텍스트 중에서 정답 엔티티를 찾아낸다. Semantic parsing(SP) 기반 방법[12]은 자연어 질문을 SPARQL, S-expression과 같은 논리적 구조를 가진 그래프 쿼리로 변환하여 지식 그래프에서 단계적으로 정답 엔티티를 추론하는 방법이다. IR 기반의 방법은 지식 그래프 검색을 위해 그래프 쿼리를 필요로 하지 않기 때문에 SP 기반의 방법보다는 구현 비용이 적지만, 단계적 추론이 필요한 질의에 대해서는 그래프를 직접 검색하는 SP 기반 방법보다 추론 성능이 떨어지는 단점을 가진다. 반면, SP 기반의 방법은 그래프 쿼리에 구문 및 의미상의 문제가 있을 경우 답이 검색되지 않아 질문에 대답할 수 없는 문제를 가진다.

최근에는 자연어 맥락 이해 및 넓은 범위에 걸친 유연한 사고 능력으로 다양한 분야에서 LLMs가 높은 성능을 보이고 있다. KGQA 분야에서도 LLMs를 도입하여, LLMs의 자연어 이해 능력과 지식 그래프 추론을 통해 복잡한 사실 관계를 파악함으로써 질문에 대한 정답을 도출하고 그에 대한 체계적인 설명을 제공하는 것을 기대하고 있다. 하지만 막대한 양의 정보를 가진 지식 그래프에서 질문에 대한 엔티티 검색과 추론에는 비용 문제가 따르기 때문에, 그래프 쿼리를 통해 도출한 추론 경로를 문서화하여 LLMs에 제공하는 연구들이 제안되고 있다[13,14]. 하지만 문서화된 트리플은 구조화된 엔티티간 관계 정보를 소실하기 때문에, entity linking이 어려워 LLMs가 질문 엔티티와 상관 없는 엔티티를 질문 엔티티로 오해할 수 있는 문제가 생길 수 있다.

따라서, SPARQL이나 그래프 토폴로지 분석에 뛰어난 성능을 보이는 그래프 신경망(graph neural network, GNN)[15]으로 검색한 추론 경로를 LLMs에 제공하는 연구가 진행되고 있다[16]. 답을 찾기 위한 지식 그래프 검색 경로를 화살표 → 형식으로 관계 정보를 잃지 않도록 표현하여 LLMs에 제공하고, LLMs는 제공된 경로를 토대로 정답을 이해하여 기존보다 잘못된 정답을 도출할 가능성을 줄일 수 있다. LLMs에 추론 경로를 제공하기 위해 그래프 신경망으로 지식 그래프를 검색하는 경우, 전체 지식 그래프를 사용하게 되면 비용적인 문제 및 정확성 저하가 발생할 수 있어 가지치기(pruning)와 같은 방법을 통해 그래프를 압축한다. 하지만 그래프의 과도한 압축으로 관련 있는 정보가 무시될 경우, 그래프 신경망에서도 정답과 관련 있는 추론 경로를 찾아내기 어렵다. 이는 정답률이 저하되거나, 정답을 맞히더라도 추론 경로를 이용한 설명이 제대로 이루어지지 않아 정답의 신뢰성이 저하되는 문제를 가질 수 있다. 따라서 본 논문에서는 1차적으로 LLMs와 그래프 신경망으로 생성한 추론 경로를 LLMs 스스로 이상이 있는지 평가하고, 보완이 필요한 경우 외부 정보 검색을 통해 질문과 관련 트리플을 생성함으로써 정답률 향상 및 설명력을 높이는 방법을 제안한다.

III. The Proposed Scheme

본 연구에서는 질문에 대한 답을 지식 그래프에서 검색하는 KGQA 분야에 있어, LLMs에서 입력된 추론 경로를 스스로 평가하고 보강하는 방법론을 제안한다. 제안하는 방법의 구성도는 Fig. 1과 같다. 1) 질의와 관련된 트리플 경로를 검색하도록 사전학습된 LLMs와 GNN을 활용하여 질의와 관련된 추론 경로를 지식 그래프에서 검색한다. 2) 검색된 추론 경로를 기반으로 LLMs에서 답변을 도출한 후 스스로 평가하여, 제공된 추론 경로에 문제가 없는지 검사한다. 3) 추론 경로 보완이 필요한 경우, 외부 지식에서 질문과 관련 있는 문맥을 찾고 LLMs를 통해 트리플로 변환하여 지식 그래프에 추가한다. 4) 보강이 완료된 트리플을 LLMs에 제공하여 질문에 대한 답과 도출 과정을 설명하도록 한다.

1. Retrieval model

LLMs에 제공할 첫 번째 추론 경로는 KGQA 훈련용 데이터셋으로 사전 훈련된 LLMs인 RoG[14]를 검색 모델(retrieval model)로 활용하여 추출한다. 훈련 데이터셋에 대하여 주어진 질문과 관련 있는 사실관계들의 흐름(relation path)을 연결하는 추론 경로를 추출하며, 다음과 같은 instruction을 사용한다.

“Please generate a valid relation path that can be helpful for answering the following question: <Question>“.

<Question>에 relation path를 생성하고자 하는 질문을 입력하면, LLMs에서 beam search 알고리즘으로 각 단계에서 높은 누적 확률을 가진 엔티티간 관계를 차례대로 추출한다. 이후 질문 내 엔티티 q_{entity}에 대하여 정답

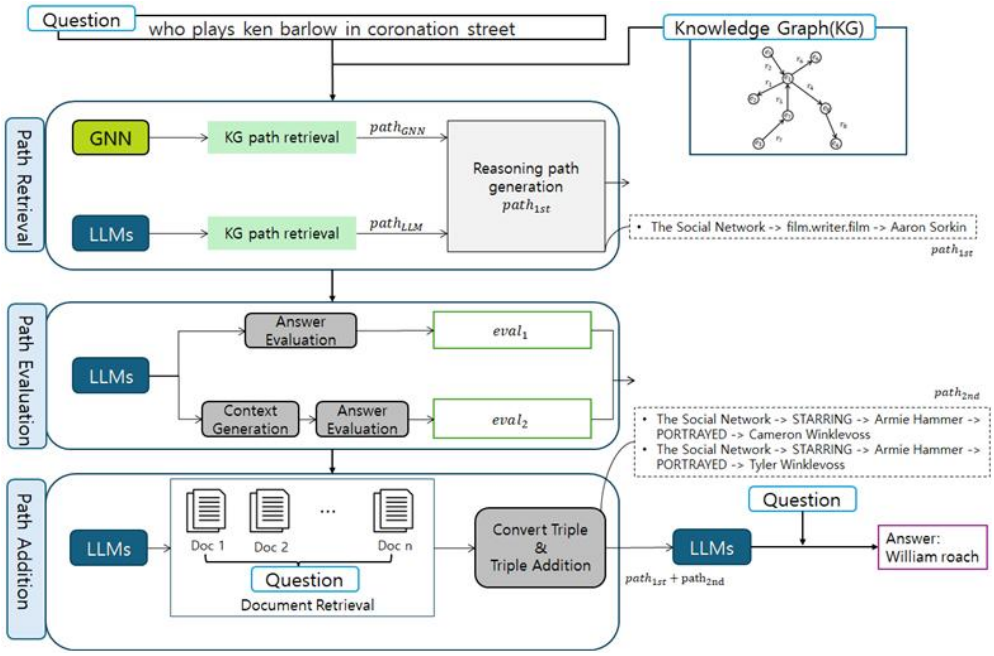

Fig. 1. Architecture for self-enhancements on knowledge graph reasoning path

에 도달할 수 있도록 생성한 relation path를 순차적으로 연결한다.

입력된 질문이 "Who is the child of Alice"이면, relation path $z: marryTo \rightarrow fatherOf$ 를 생성한다. 만약 q_{entity}가 Alice, 정답이 Charlie라면 Alice부터 relation path를 순차적으로 연결하여 Alice $\xrightarrow{marryTo}$ Bob $\xrightarrow{fatherOf}$ Charlie 인 추론 경로가 도출된다. 훈련 데이터셋에 대해 질문 q, 정답 a, 지식 그래프 G가 주어졌을 때, 생성된 추론 경로에서 정답 a를 검색할 확률을 극대화시키도록 relation path z를 생성하는 모델을 훈련시킨다. 훈련식은 식 (1)과 같이 정규화하여 표현할 수 있다.

$$P_\theta(a|q,G) = \sum_{z \in Z} \mathrm{P}_\theta(\mathrm{a}|\mathrm{q},\mathrm{z},\mathrm{G})\mathrm{P}_\theta(\mathrm{z}|\mathrm{q}), \qquad (1)$$

이 때, θ는 LLMs의 학습 파라미터, Z는 생성될 수 있는 relation path z의 집합을 의미한다. $P_\theta(z|q)$는 질의 q에서 z가 생성될 확률을, $P_\theta(a|q,z,G)$는 질문 q, 지식 그래프 G, relation path z가 주어졌을 때 정답 a에 대한 확률을 의미한다. 훈련이 완료되면, 입력된 질문에 대해 생성한 k개의 relation path로 도출된 추론 경로를 결과적으로 LLMs에서 식 (2)와 같이 얻을 수 있다.

$$path_{LLM} = \{q_{entity} \rightarrow r_1 \rightarrow e_1 \rightarrow r_2 \rightarrow \ldots \rightarrow a_{llm}\}_k, \qquad (2)$$

a_{llm}은 LLMs에서 단계적 추론으로 도달한 답변을 의미한다.

추가적으로 GNN을 활용하여 지식 그래프에서 질문에 대한 답에 도달하는 경로를 추론한다. GNN은 그래프 구조 데이터 분석에 뛰어난 성능을 가진 신경망으로, message passing 전략을 통해 이웃 노드 정보와 상호작용하여 목적에 맞게 각 노드의 표현을 최적화시킨다. 질의응답에서는 주어진 질문과 관련된 서브 그래프 g에서 질문 엔티티가 속하는 트리플(s_q, r_q, o_q)에 대하여, GNN 레이어 함수 $\phi(w(q_e, r_e))$를 통해 그래프를 구성하는 엔티티의 표현을 업데이트하고, 가장 정답에 근접한 노드를 찾는 것을 목표로 한다. $w(q_e, r_e)$는 임베딩 모델로 표현된 질문과 엔티티간 관계 $q_e = LM(q)$ 및 $r_e = LM(r_q)$ 사이의 의미적 유사성을 계산하며, $\sigma(q_e \odot r_e)$으로 표현한다. $LM(\cdot)$은 SentenceBERT와 같은 임베딩 모델을, σ는 활성화(activation) 함수, $\odot$은 원소간 곱(element-wise multiplication)을 의미한다. ϕ는 Rearev[17], Nutrea[18]와 같은 KGQA에 최적화된 GNN 모델 레이어를 의미한다. GNN을 통해 모든 엔티티 노드 E_g에 대한 업데이트가 완료되었을 때, 가장 확률 점수가 높은 엔티티인 $a_g = softmax(E_g \theta_g)$를 찾는다. θ_g는 learnable parameter를, $softmax$는 softmax 함수를 의미한다. 최종적으로 입력 질문에 대한 추론 경로는 질문 엔티티 q_{entity}부터 단계적 추론으로 도출한 정답 a_g로 도달하는 최단 경로를 서브 그래프 g에서 너비 우선 탐색(breadth

first search, BFS)를 적용하여 탐색할 수 있다. 해당 과정은 식 (3)-(7)과 같이 정리할 수 있다.

$$q_e = LM(q),\ r_e = LM(r_q), \tag{3}$$

$$w(q_e, r_e) = \sigma(q_e \odot r_e), \tag{4}$$

$$e_g = \phi(w(q_e, r_e)), where\ e_g \in E_g, \tag{5}$$

$$a_g = softmax(E_g \theta_g), \tag{6}$$

$$path_{GNN} = BFS(g, q_{entity}, a_g), \tag{7}$$

최종적으로 서로 다른 방법으로 생성된 두 추론 경로를 합쳐 $path_{1st} = path_{LLM} + path_{GNN}$를 생성하게 된다.

2. Generate and inspect first-round answers

LLMs와 GNN을 통해 생성한 $path_{LLM}$ 및 $path_{GNN}$은 FastStore[19]과 같은 트리플 저장소에서 그래프 쿼리를 통해 부분적으로 추출한 서브 그래프를 활용한다. 본 절에서는 $path_{1st}$를 LLMs에 제공하였을 때, LLMs가 주어진 추론 경로를 통해 질문에 대한 답을 도출할 수 있는지 확인한다. 추론 경로는 두 가지 방법으로 검사된다. 1) 추론 경로를 LLMs에 제공하여 도출된 답이 맞는지 LLMs 스스로 평가시킨다. 2) 추론 경로를 LLMs가 이해하기 쉬운 문장 조합으로 구성하고, LLMs가 답을 도출할 수 있는지 평가시킨다.

첫 번째로, 주어진 추론 경로를 기반으로 LLMs에서 답을 도출할 수 있는지 확인하였다. 이 때, KGQA 훈련 데이터셋의 일부를 활용하여 Fig. 2와 같이 원 샷 프롬프트 형식으로 instruction을 설계해 LLMs가 매끄러운 답변을 출력할 수 있도록 하였다. Fig. 2의 {reasoning}에는 추론 경로가, {question}에는 질문이 입력된다.

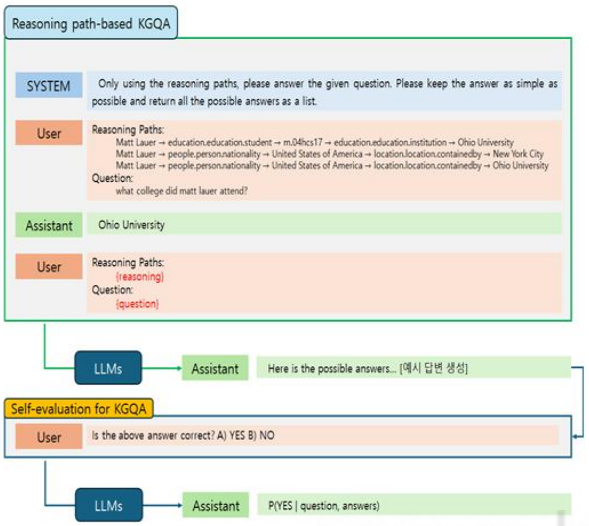

Fig. 2. Self-evaluation example of reasoning path in LLMs

LLMs가 생성한 출력을 자가 평가하는 것으로 피드백 루프를 형성하여, 잘못된 예측이나 출력의 불확실성을 보정하여 출력을 개선할 수 있다고 주장되었다[20]. 본 연구에서는 이와 같은 주장에 영향을 받아, 1차적으로 출력된 정답에 대하여 LLMs 스스로 자가 평가를 진행하도록 하였다. 자가 평가 결과는 정답/오답인 이진(binary) 분류 점수로 나타내며, 식 (8), (9)로 표현한다.

$$\hat{y_1} = \arg max\, \Pi_{i=1}^{n}\, p(a_i | q, path_{1st}), \tag{8}$$

$$eval_1 = p(yes | q, \hat{y_1}), \tag{9}$$

두 번째로, 추론 경로를 질문과 관련된 문장들의 조합으로 나타내고, 생성된 텍스트에서 정답을 도출할 수 있는지 확인한다. 첫 번째 방법은 LLMs가 추론 과정을 파악하기 쉽게 →로 엔티티간 관계의 흐름을 나타낸 것과 달리, 두 번째 방법에서는 LLMs가 추론 경로의 내용을 쉽게 파악할 수 있게 문장화하였다. Fig. 3과 같이 제로 샷 프롬프트를 활용하여 LLMs에서 추론 경로를 다른 관점에서 평가하였다.

프롬프트 결과로 LLMs에서는 추론 경로로 텍스트를 출력한다. 첫 번째 방법과 마찬가지로, LLMs에서 입력된 텍스트와 질문을 바탕으로 답을 출력할 수 있는지 이진 분류로 자가 평가한다. 이 과정은 식 (10), (11)으로 나타낼 수 있다.

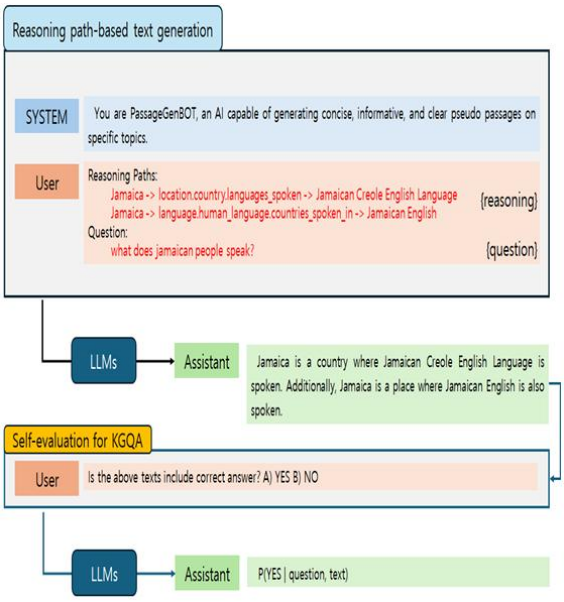

Fig. 3. Self-evaluation example of contextual reasoning in LLMs

$$\widehat{y_2} = \Pi_{i=1}^{t} p(y_i^* | y_{<i}^*, q, path_{1st}), \quad (10)$$

$$eval_2 = p(yes | q, \widehat{y_2}), \quad (11)$$

y^*는 추론 경로를 통해 생성되는 텍스트 단어를 의미한다. 식 (9) 및 (11)을 통해 LLMs 스스로 추론 경로를 평가한 결과, $eval_1$과 $eval_2$ 중 한 번이라도 0이라고 판단하게 되면 경로에 이상이 있다고 판단하여 트리플 보강 작업을 수행하게 된다.

3. Triple enhancement

본 절에서는 자가 평가 결과로 입력된 추론 경로에 문제가 있다고 판단되면 RAG를 통해 질문과 관련된 문서를 검색하여 트리플을 생성한다. 관련 문서는 질문과의 코사인 유사도로 계산하였다. 사전 훈련된 sentence transformers 모델[21]인 'all-miniLM-L6-v2'를 사용하여 임베딩한 질의 문장과 검색 문서간 유사도로 계산하였으며, 해당 과정은 google search와 같은 웹 검색 엔진으로 진행하였다. 이 때, LLMs에서 처리할 수 있는 컨텍스트의 길이는 최대 값이 정해져 있으므로, 웹사이트 문서는 고정된 길이 단위로 청킹(chunking) 분할하였다. 또한 웹사이트 문서에는 질문과 불필요한 정보 제거를 위해 관련도로 랭킹(ranking)시켜 낮은 순위의 청크 문서는 제거한다. 문서의 랭킹은 청크 문서 집합과 질문 사이의 코사인 유사도로 계산하며, Facebook Research에서 공개한 FAISS[22]를 사용하였다. 두 임베딩된 텍스트 벡터 s_1, s_2가 주어졌을 때, 두 벡터간 코사인 유사도는 식 (12)와 같이 계산한다.

$$\text{cosinSim}(s_1, s_2) = \frac{s_1 \cdot s_2}{\| s_1 \| \| s_2 \|}, \quad (12)$$

다음으로, 상위 청크 문서 집합의 내용을 분석하여 트리플을 생성한다. 트리플 생성은 퓨 샷 프롬프트로 문장 속에서 생성될 수 있는 트리플 구조에 대한 예시 정보를 LLMs가 동적으로 문장을 분석하여 엔티티간 관계를 추출하며, 랭체인에서 제공하는 프롬프트 템플릿을 사용하였다.

트리플 구조에 대한 예시 정보는 문장과 문장에서 나타날 수 있는 주체, 타겟 엔티티, 관계 정보와 각 타입을 제공한다. 예를 들어, {"text": "Microsoft Word is a lightweight app that accessible offline", "head": "Microsoft Word", "head_type": "Product", "relation": "HAS_CHARACTERISTIC", "tail":"accessible offline", "tail_type": "Characteristic"} 같은 입력 문장 및 생성할 수 있는 트리플 예시를 몇 가지 제공하고, LLMs가 이를 참고하여 제공된 텍스트에서 트리플을 추출하도록 하였다.

이후 생성된 트리플들을 그래프로 연결하여 서브 그래프 g_{rag}를 생성한다. 모든 트리플 후보를 추론 경로로 LLMs에 제공하면 답변에 noise가 생겨 정답률이 저하될 가능성이 있다. 따라서 q_{entity}부터 검색한 웹 문서에서 도출된 답변 a_{rag}에 도달하는 최단 경로를 탐색하여 식 (13)의 추론 경로를 생성한다. 따라서 생성된 추론 경로는 $path_{1st}$에 연결함으로써 최종적으로 LLMs에 제공할 추론 경로 $path_{final}$을 식 (14)와 같이 나타낸다.

$$path_{2nd} = BFS(g_{rag}, q_{entity}, a_{rag}), \quad (13)$$

$$path_{final} = [path_{1st}; path_{2nd}], \quad (14)$$

4. Answer prediction and explanation based on reasoning paths

최종적으로 생성된 추론 경로를 기반으로 LLMs에서 질문에 대한 답을 생성한다. Fig. 2의 원 샷 프롬프트를 동일하게 사용하여 질의와 추론 경로를 입력하고, 식 (15)와 같이 LLMs가 답을 출력한다.

$$\widehat{y_F} = \arg max\, \Pi_{i=1}^{n} p(a_i | q, path_{final}). \quad (15)$$

또한 추론 경로를 통해 답을 도출할 수 있는지 확인하기 위해, 답과 이유를 묻는 프롬프트를 LLMs의 입력으로 제공하여 답변을 도출한다. 프롬프트는 "Based on the reasoning paths, please answer the given question and explain why. Reasoning Paths: {reasoning} Question: {question}" 을 사용했으며, 퓨 샷 프롬프트로 파인 튜닝된 LLMs 모델을 사용하여 결과를 출력한다.

IV. Experiments

본 연구에서는 제안하는 방법의 성능 실험을 위해 대표적인 KGQA 데이터셋인 WebQuestion Semantic Parsing(WebQSP) 데이터셋을 사용하였다. WebQSP 데이터셋은 Freebase[23]에서 유래한 4,737개의 질문-답변의 쌍으로 구성되어 있으며, 최대 2-홉의 추론을 필요로 한다. WebQSP 데이터셋은 훈련용 질문-답변 쌍 2,826개와 테스트용 질문-답변 쌍 1,628개로 구성된다. 본 연구

에서는 실험을 위해 테스트용 질문-답변 쌍을 사용하였다.

성능 평가 지표로는 정확도(accuracy), Hit, Hits@1, F1 점수를 사용하였다. 정확도는 LLMs가 예측한 전체 대답 중에서 올바르게 예측한 비율을 나타낸다. Hit은 LLMs의 성능을 평가하기 위해 주로 사용되는 지표로, LLMs가 예측한 결과가 정답 중 하나에 포함되는지 평가한다. Hit@1은 모델이 첫 번째로 출력한 결과가 정답과 일치하는지 측정한다. F1 점수는 예측 결과의 정밀도(precision)과 재현율(recall)의 조화 평균으로 계산한다. 정밀도는 모델이 예측한 결과 중에서 실제 정답이 차지하는 비율을, 재현율은 실제 정답 중에서 모델이 예측한 결과의 비율을 나타낸다. 본 실험에서는 서로 다른 유형의 추론 경로를 기반으로 8B 파라미터의 수를 가진 Llama3[24]를 사용하여 성능을 판단하였다.

먼저 제안한 방법으로 생성한 추론 결과가 LLMs의 정답률에 영향을 미치는지 측정하였다. WebQSP 데이터셋을 기반으로 파인튜닝된 LLMs로 생성한 추론 경로와 GNN을 통해 생성한 추론 경로 및 제안하는 방법을 통해 보강된 추론 경로를 사용하였을 때, Hit, Hits@1, F1 점수의 변화를 측정하였다. 실험 결과는 Table 1과 같다.

Table 1. KGQA performances based on reasoning path settings using WebQSP

Reasoning path setting	Dataset			
	WebQSP			
	Acc. (%)	Hit (%)	Hit@1 (%)	F1 (%)
LLMs	75.1	85.8	78.5	69.0
LLMs+GNN	75.9	86.4	78.6	69.4
LLMs+GNN+ our method	80.9	90.5	81.0	71.1

실험 결과를 통해, 정확도, Hits, Hits@1, F1 점수 측면에서 모두 제안하는 연구에서 생성한 추론 경로가 정답을 정확하게 예측하는 것을 알 수 있다. 이는 추론 경로를 기반으로 질문에 답변을 생성하였을 때, 제안하는 방법이 파인튜닝을 활용한 LLMs 생성 추론 경로 및 GNN을 활용한 추론 경로의 성능을 보완하는 것을 나타낸다.

또한 제안하는 방법이 WebQSP 데이터셋을 기반으로 파인튜닝된 LLMs로 생성한 추론 경로와 GNN을 통해 생성한 추론 경로로 오답을 도출한 경우를 잘 파악하는지 정확성을 측정하였다. 이에 대해 제안한 방법으로 이상 트리플을 탐지한 정확도 결과는 Table 2와 같다.

Table 2. KGQA dataset anomaly detection accuracy performances

Reasoning path setting	Acc.(%)
LLMs	94.8
LLMs+GNN	97.3

실험 결과, LLMs 기반 추론 경로를 이용하여 오답을 도출한 경우를 94.8% 로, LLMs와 GNN을 사용한 추론 경로를 이용하였으나 오답이 도출된 경우를 97.3%로 이상을 탐지할 수 있었다. 따라서 두 가지 경우 모두 90% 이상의 정확도로 경로의 이상을 판단하는 것을 알 수 있었으며, 이를 통해 추론 경로 및 추론 경로 기반의 텍스트를 기반으로 LLMs가 자가 평가하는 것으로 주어진 데이터셋을 평가할 수 있다고 판단된다.

다음으로 제안 방법으로 생성한 추론 경로가 기존 경로보다 해석 가능한 설명을 제공하고 있는지 확인하였다. 입력으로 질의와 추론 경로를 LLMs에 제공하고, LLMs가 질의에 대한 답과 그 이유에 대한 설명을 출력하도록 하였다. 해당 출력은 WebQSP 훈련 셋으로 설명을 출력시키기 위해 파인튜닝된 llama2-chat-hf 모델인 RoG[14]를 사용하였다. 설명의 출력 여부는 LLMs에서 답을 출력하지 못하거나, ‘cannot provide’, ‘no (specific) information’ 문구의 포함 여부로 판단하였다. 추론 경로의 설명력에 대한 성능은 설명률 및 정답 설명률로 평가하였다. 설명률은 전체 문제 중에 정답을 설명을 출력한 비율을 나타내며, 정답 설명률은 각 추론 경로를 기반으로 정답을 맞혔으며, 정답을 설명할 수 있는지 여부를 나타낸다. 실험 결과는 Table 3과 같다.

Table 3. LLMs explainability performance depending on reasoning path setting

Reasoning path setting	Dataset	
	WebQSP	
	Explanation rates (%)	Answer explanation rates (%)
LLMs	82.5	85.69
LLMs+GNN	84.8	86.77
LLMs+GNN+ our method	91.1	93.31

실험 결과, 기존의 추론 경로보다 제안하는 추론 경로가 실질적으로 더 많은 질의를 설명할 수 있는 것을 알 수 있다. 이는 기존 추론 경로를 사용할 경우엔 LLMs에서 정답의 이유를 설명할 수 없었으나, 제안하는 방법으로 나타낸 추론 경로를 사용할 경우엔 비교적 정확하게 정답의 이유

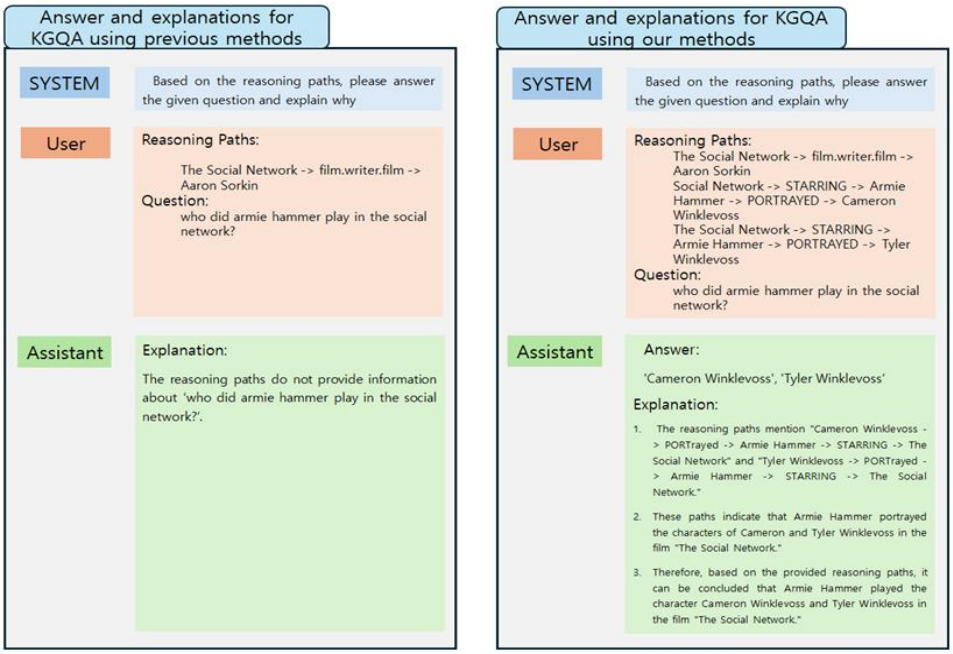

Fig. 4. KGQA explanations in LLMs based on reasoning paths

를 설명할 수 있음을 보인다. 따라서 사용자가 LLMs의 정답 추론 과정을 이해할 수 있어 환각 현상의 저하와 추론 성능 강화를 위한 데이터셋 보완이 가능한 것을 기대할 수 있다. Fig. 4는 실질적인 예를 나타낸다. 질문 'who did armie hammer play in the social network'를 이용하여 기존 방법과 제안하는 방법을 이용하여 질문에 대한 답과 그 이유를 출력시켜 보았다. 기존 방법으로 도출한 추론 경로의 경우, 'The Social Network → film.writer.film → Aaron Sorkin' 만 도출되어 질문에 대한 정답을 추론할 수 없는 것으로 보여진다. 반면에 제안 방법으로 추론 경로를 보강했을 경우, 'The Social Network → STARRING → Armie Hammer → PORTRAYED → Cameron Winklevoss', 'The Social Network → STARRING → Armie Hammer → PORTRAYED → Tyler Winklevoss' 경로가 추가되어 'The social network'에서 정답인 'Cameron Winklevoss' 및 'Tyler Winklevoss'까지 도달할 수 있도록 단계적 추론이 가능한 것을 보였다.

V. Conclusion

기존의 KGQA 분야에서는 SPARQL과 같은 그래프 검색을 위한 논리적 질의를 통해 검색된 추론 경로를 활용하여 정답을 추출하는 형식을 채택하였다. 하지만 문법적 혹은 의미적으로 질의에 문제가 생길 경우 제대로 추론 경로가 생성되지 않아 LLMs가 주어진 질문에 제대로 답하지 못하는 문제가 생긴다. 또한 추론 경로를 통해 LLMs가 주어진 답변을 출력한 이유를 판단할 수 있어야 하는데, LLMs에서는 답변을 출력하였으나 그 이유를 출력할 수 없어 사용자가 LLMs의 판단을 신뢰할 수 없는 문제점을 가진다. 따라서 본 연구에서는 LLMs 스스로 주어진 추론 경로를 토대로 질문의 답에 도달할 수 있는 평가하고, 문제가 있다고 판단되면 외부 정보 검색을 통해 질문에 대한 답과 그에 도달하기 위한 추론 경로를 생성하는 방법을 제안하였다. 이 때 LLMs 자가 평가의 신뢰성을 높이기 위하여 1) LLMs가 추론 경로를 통해 문제에 대한 답을 맞추는 과정으로 LLMs의 추론 능력에 문제가 있었는지 2) 텍스트화한 추론 경로를 LLMs의 문장 이해 능력으로 분석시킴으로써 추론 경로에 필요 정보가 누락되어 있는지를 판단하였다. 자가 평가에 대한 성능 평가 결과, 추론 경로에 문제가 있는 데이터셋에 대하여 높은 정확도로 이상을 판단하고 있음을 보였다.

또한 제안하는 연구는 외부 정보 전체를 추론 경로로 변환하는 것이 아닌 유사도 검색을 통해 관련 텍스트만 추론 경로로 변환함으로써 신뢰성 있는 추론 경로를 생성하였다. KGQA의 성능을 평가한 결과, 기존 연구로 생성된 추론 경로보다 제안하는 방법으로 생성한 추론 경로를 사용할 때 더 높은 정답률을 보이는 것을 확인할 수 있었다. 또한 LLMs와 정답과 함께 그에 대한 설명을 제공할 수 있는지 여부를 실험하였을 때, 제안하는 방법을 사용한 추론 경로가 더 많은 유형의 질문에 대한 설명을 제공할 수 있

음을 판단할 수 있었다.

LLMs에 외부 정보가 주어진 경우, RAG의 성능은 외부 정보의 정확성에 영향을 받게 된다. 따라서 외부 정보를 LLMs에 제공해야 하는 경우 LLMs 스스로 정보를 평가하고 보완하는 과정은 추후 질의응답 성능이 긍정적인 영향으로 이루어질 수 있다. 향후 연구로는 자가 보완을 통한 KGQA의 정확성 향상을 진행할 필요성이 있다.

ACKNOWLEDGEMENT

This work was supported by the Korea Institute of Energy Technology Evaluation and Planning(KETEP) and the Ministry of Trade, Industry & Energy(MOTIE) of the Republic of Korea (No. 20222020900003C).

REFERENCES

[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," Proceedings of Advances in Neural Information Processing Systems, vol. 33, pp. 1877-1901, 2020.

[2] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, "Palm: Scaling language modeling with pathways," Journal of Machine Learning Research, vol. 24, no. 240, pp. 11324-11436, March. 2024.

[3] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang, "Retrieval-augmented generation for large language models: A survey," arXiv preprint arXiv:2312.10997, March 2023. DOI: 10.48550/arXiv.2312. 10997.

[4] Y. Dai, S. Wang, N. N. Xiong, and W. Guo, "A survey on knowledge graph embedding: Approaches, applications and benchmarks," Electronics, vol. 9, no. 5, pp. 750, May 2020. DOI: 10.3390/electronics9050750.

[5] N. Kandpal, H. Deng, A. Roberts, E. Wallace, and C. Raffel, "Large language models struggle to learn long-tail knowledge," in Proceedings of Machine Learning Research, pp. 15696-15707, 2023.

[6] S. Xu, L. Pang, H. Shen, X. Cheng, and T. S. Chua, "Search-in-the-Chain: Interactively Enhancing Large Language Models with Search for Knowledge-intensive Tasks," in Proceedings of the ACM Web Conference, pp. 1362-1373, 2024.

[7] C. Feng, X. Zhang, and Z. Fei, "Knowledge solver: Teaching LLMs to search for domain knowledge from knowledge graphs," arXiv preprint arXiv:2309.03118, September 2023. DOI: 10.48550/arXiv. 2309.03118.

[8] X. Wang, Q. Yang, Y. Qiu, J. Liang, Q. He, Z. Gu, Y. Xiao, and W. Wang, "Knowledgpt: Enhancing large language models with retrieval and storage access on knowledge bases," arXiv preprint arXiv:2308.11761, August 2023. DOI: 10.48550/arXiv. 2308.11761.

[9] Y. Yasunaga, H. Ren, A. Bosselut, P. Liang, and J. Leskovec, "QA-GNN: Reasoning with language models and knowledge graphs for question answering," arXiv preprint arXiv:2104.06378, December 2021. DOI: 10.48550/arXiv.2104.06378.

[10] J. Berant, A. Chou, R. Frostig, and P. Liang, "Semantic parsing on freebase from question-answer pairs," in Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1533-1544, 2013.

[11] J. Jiang, K. Zhou, W. X. Zhao, and J. R. Wen, "Unikgqa: Unified retrieval and reasoning for solving multi-hop question answering over knowledge graph," arXiv preprint arXiv:2212.00959, March 2022. DOI: 10.48550/arXiv.2212.00959.

[12] H. Luo, H. E, Z. Tang, S. Peng, Y. Guo, W. Zhang, C. Ma, G. Dong, M. Song, W. Lin, Y. Zhu, and L. A. Tuan, "Chatkbqa: A generate-then-retrieve framework for knowledge base question answering with fine-tuned large language models," arXiv preprint arXiv:2310.08975, May 2023. DOI: 10.48550/arXiv.2310.08975.

[13] J. Sun, C. Xu, L. Tang, S. Wang, C. Lin, Y. Gong, L. M. Ni, H.-Y. Shum, and J. Guo, "Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph," arXiv preprint arXiv:2307.07697, March 2024. DOI: 10.48550/arXiv. 2307.07697.

[14] L. Luo, Y. F. Li, G. Haffari, and S. Pan, "Reasoning on graphs: Faithful and interpretable large language model reasoning," arXiv preprint arXiv:2310.01061, Feb 2024. DOI: 10.48550/arXiv.2310. 01061.

[15] J. Zhou, G. Cui, Z. Zhang, C. Yang, Z. Liu, and M. Sun, "Graph

neural networks: A review of methods and applications," AI Open, vol. 1, pp. 57-81, 2020. DOI: 10.1016/j.aiopen.2021.01.001.

[16] C. Mavromatis and G. Karypis, "GNN-RAG: Graph Neural Retrieval for Large Language Model Reasoning," arXiv preprint arXiv:2405.20139, May 2024. DOI: 10.48550/arXiv.2405.20139.

[17] C. Mavromatis and G. Karypis, "Rearev: Adaptive Reasoning for Question Answering Over Knowledge Graphs," arXiv preprint arXiv:2210.13650, October 2022. DOI: 10.48550/arXiv.2210.13650.

[18] H. K. Choi, S. Lee, J. Chu, and H. J. Kim, "NuTrea: Neural Tree Search for Context-Guided Multi-Hop KGQA," Proceedings of Advances in Neural Information Processing Systems, vol. 36, 2024.

[19] Z. Xiong, D. Jiang, and J. Xiong, "FastStore: A High-Performance RDMA-Enabled Distributed Key-Value Store with Persistent Memory," Proceedings of the 2023 IEEE 43rd International Conference on Distributed Computing Systems, pp. 406-417, 2023.

[20] J. Ren, Y. Zhao, T. Vu, P. J. Liu, and B. Lakshminarayanan, "Self-evaluation improves selective generation in large language models," Proceedings of Machine Learning Research, pp. 49-64, 2023.

[21] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," arXiv preprint arXiv:1908.10084, August 2019. DOI: 10.48550/arXiv.1908.10084.

[22] M. Douze, A. Guzhva, C. Deng, J. Johnson, G. Szilvasy, P.-E. Mazaré, M. Lomeli, L. Hosseini, and H. Jégou, "The FAISS library," arXiv preprint arXiv:2401.08281, January 2024. DOI: 10.48550/arXiv.2401.08281.

[23] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp. 1247-1250, 2008.

[24] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, É. Grave, and G. Lample, "Llama: Open and efficient foundation language models," arXiv preprint arXiv:2302.13971, February 2023. DOI: 10.48550/arXiv.2302.13971.

Authors

Min-Ji Seo received the B.S. in Computer Science and Engineering (2015) and the M.S. and ph.D in software convergence from Soongsil University, Seoul, Korea, in 2017 and 2022, respectively.

Her research interests are Deep Learning, Machine Learning, Audio-based sentimental analysis, Time series anomaly detection, Explainable Artificial Intelligence, Large Language Models and Big data analysis.

Myung-Ho Kim received the B.S. in Department of Computer Science and Engineering from Soongsil University, Korea, in 1989. M.S. and Ph.D. degrees in Department of Computer Engineering from Postech University, Korea, in 1991 and 1995, respectively. He is currently a professor in the Dept. of Software, Soongsil University. He is interested in Machine Learning, Deep Learning and Block chain.