Check for updates

Journal of KIIT. Vol. 24, No. 10, pp. 247-264, Oct. 31, 2026. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2026.24.10.247

정부 예산설명서 표준 템플릿의 AI 친화적 구조화 방법론 연구 : 데이터 표현 형식별 질의 가능성 비교

임수혁*, 서영균**

AI-Ready Structuralization of Government Budget Explanation Templates: A Comparison of Queryability across Data Representation Formats

Soohyuk Im*, Young-Kyoon Suh**

이 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(RS-2025-16067408).

요 약

정부 예산설명서는 표준화된 행정문서임에도 HWP·PDF 등 반정형 포맷으로 발행되어 AI 활용에 한계가 있다. 본 연구는 데이터 표현 형식과 활용 도구의 결합이 질의 가능성에 미치는 영향을 분석하기 위해, 2026년 AI 재정사업 533건을 대상으로 메타데이터 스키마와 도메인 온톨로지(BEDO v1.0)를 설계하고 100문항 평가셋에서 5가지 결합 방식의 성능을 비교하였다. 그 결과 XLSX(F1 65.40%), RDF(58.40%), Pandas Agent(54.57%) 세 방식이 Plain Text Vector RAG(33.77%)보다 통계적으로 유의하게 높았으며, 세 방식 간에는 유의한 우열이 확인되지 않았다. 반면 JSON 직렬화(14.83%)는 구조 인식 도구 없이 사용한 조건의 한계를 보였다. 본 평가셋 범위 내에서 반정형 행정문서의 AI 활용성은 머신리더블 수준뿐 아니라 표현-도구 적합성에 의해 좌우될 수 있으며, 이는 「대한민국 인공지능행동계획」 추진을 위한 공공 문서의 질의 가능한 지식 구조 설계에 참고가 될 수 있다.

Abstract

Government budget documents follow standardized administrative templates but are mostly issued in semi-structured formats such as HWP or PDF, limiting AI utilization. To analyze how the combination of data representation format and utilization tool affects queryability, this study designs an AI-ready metadata schema and the Budget Explanation Domain Ontology (BEDO v1.0) using 533 AI-related fiscal projects from 2026, and compares five representation-tool combinations on a 100-question evaluation set. Three methods—XLSX (F1 65.40%), RDF/SPARQL (58.40%), and the Pandas Agent (54.57%)—significantly outperform Plain Text Vector RAG (33.77%), while no significant differences are observed among the three. JSON serialization (14.83%) shows the limitations of using a serialized text representation without a structure-aware executor. Within the scope of this evaluation, the findings suggest that effective AI utilization of semi-structured administrative documents depends on Format-Tool Fit as well as machine-readability, providing a reference for designing public documents as queryable knowledge structures in support of national initiatives such as the Artificial Intelligence Action Plan of the Republic of Korea.

Keywords

public administrative data, government budget explanation, machine-readable data, AI-ready public data, RDF/SPARQL, text-to-query, queryability

* 한국지능정보사회진흥원 책임,
경북대학교 정보과학과 박사수료
- ORCID: https://orcid.org/0009-0008-2323-5618
** 경북대학교 컴퓨터학부 및 정보과학과 교수(교신저자)
- ORCID: https://orcid.org/0000-0003-3124-2566

· Received: May 15, 2026, Revised: Jun. 12, 2026, Accepted: Jun. 15, 2026

· Corresponding Author: Young-Kyoon Suh
Rm. 520, IT-5, 80 Daehak-ro, Buk-gu, Daegu, South Korea
Tel.: +82-53-950-6372, Email: yksuh@knu.ac.kr

Ⅰ. 서 론

정부 예산설명서는 사업명, 소관 부처, 분야·부문, 프로그램·단위사업, 시행기관, 지원형태, 연도별 예산액 등 표준화된 항목을 포함하는 행정문서이다. 그러나 실제 발행 형식은 PDF나 HWP와 같은 문서 중심의 반정형 포맷에 머무는 경우가 많아, AI 시스템을 통한 검색, 집계, 관계 탐색의 자동화에는 한계가 있다. 최근 국가인공지능전략위원회에서 「대한민국 인공지능행동계획(2026~2028)」이 의결되는 등 공공부문의 인공지능 전환은 제도적으로 강화되고 있다[1]. 그러나 공공데이터 정책은 양적 개방 확대에도 불구하고 실제 활용성과 재사용성 측면에서 간극을 보이고 있으며[2], 공공데이터의 인공지능 활용 가능성을 높이기 위해 오픈 포맷, 명시적 스키마, 기계 판독 가능한 메타데이터 등 AI 친화적 관리 원칙이 제시하고 있다[3]. 따라서 예산설명서와 같이 표준 항목을 포함하지만 문서 중심 형식으로 제공되는 반정형 행정문서에 대해서는, AI 시스템이 질의·검색·집계·관계 탐색에 활용할 수 있도록 데이터 구조를 재설계하는 방법론 연구가 필요하다.

FAIR 원칙과 5-star Linked Data 모델은 데이터의 발견 가능성, 접근성, 상호운용성, 재사용성, 기계 판독 가능성 등 데이터 발행과 관리의 규범적 기준을 제시해 왔다[4][5]. 그러나 이러한 논의는 주로 데이터가 어떤 조건을 갖추어야 기계가 찾고 재사용할 수 있는지를 설명하는 데 초점을 두며, 특정 행정문서를 여러 표현 형식으로 변환한 뒤 각 형식을 실제 질의 도구와 결합했을 때 도달 가능한 질의 성능을 실증적으로 비교하는 데에는 초점을 두지 않는다. 한편 기존 검색증강생성(RAG, Retrieval-Augmented Generation) 연구는 외부 문서 검색 결과를 생성 모델에 결합하거나, 검색·재순위화·생성 모듈을 고도화하는 방향으로 발전해 왔다[6][7]. 이러한 연구 흐름은 질의응답 성능 향상에 중요한 기반을 제공하지만, 동일한 원천 행정문서를 서로 다른 데이터 표현 형식으로 변환하고 각 형식을 적합한 활용 도구와 결합했을 때 질의 가능성이 어떻게 달라지는지를 데이터 발행자 관점에서 비교하는 데에는 상대적으로 제한적이었다.

한편 최근에는 그래프 기반 검색증강생성이나 표 구조를 고려한 질의응답 연구가 제안되면서, 데이터의 구조적 표현이 질의응답 과정에서 중요하게 작용할 수 있음이 논의되고 있다[8][9]. 그러나 이러한 연구 역시 정부 예산설명서와 같은 표준 행정문서를 대상으로, 데이터 표현 형식과 활용 도구의 결합 효과를 통제된 조건에서 비교한 것은 아니다. 이에 본 연구는 2026년 AI 재정사업 533건을 대상으로 AI-ready 메타데이터 스키마와 예산설명서 도메인 온톨로지(BEDO v1.0)를 구축하고, 동일한 원천 데이터를 Plain Text, JSON, RDF, XLSX로 변환한 뒤 5가지 표현-도구 결합 방식의 질의 가능성을 비교한다.

본 연구의 연구 문제는 다음과 같다. 정부 예산설명서 표준 템플릿을 문서형, JSON, RDF, XLSX로 표현하고 이를 각기 다른 질의 도구와 결합할 때, 질의 유형별 질의 가능성은 어떻게 달라지는가? 특히 자연어 입력 기반 방식과 구조 질의 조건 기반 방식은 다단계 관계, 역방향 탐색, 복합 조건, 집계 질의에서 어떤 차이를 보이는가? 본 연구는 이를 확인하기 위해 자연어 입력 베이스라인(Vector RAG, Pandas Agent)과 구조 질의 조건 기반의 준상한(Quasi-upper-bound) 방식을 구분하고, 정답을 생성에 사용하지 않는 No-Oracle 평가 원칙과 페어드 부트스트랩 검정을 적용한다.

본 논문의 기여는 세 가지다. 첫째, 정부 예산설명서 표준 템플릿의 AI 활용을 위한 메타데이터 스키마 및 도메인 온톨로지(BEDO v1.0)를 정식화한다. 둘째, 5가지 표현-도구 결합 방식에 대한 100문항 평가를 통해 데이터 표현 형식과 활용 도구의 적합성이 질의 가능성에 미치는 영향을 실증적으로 분석한다. 셋째, 평가셋, 예측 결과, 평가 스크립트, 부트스트랩 결과를 공개하여 재현 가능한 실험 패키지를 제공한다.

본 논문의 구성은 다음과 같다. 2장에서는 검색증강생성, 머신리더블 데이터, 시맨틱 웹 및 구조화 질의 변환 관련 연구를 검토한다. 3장에서는 연구 문제를 제시하고, 4장에서는 데이터셋, AI-ready 스키마, 다중 표현 형식 변환, 질의 방식 및 평가 기준을 설명한다. 5장에서는 실험 환경과 결과를 분석하고, 6장에서는 연구 결과와 한계를 종합한 뒤 향후 연구 방향을 제시한다.

II. 관련 연구

본 연구와 직접 관련된 선행연구는 머신리더블·AI-Ready 공공데이터, 검색증강생성 및 구조화 검색, RDF/SPARQL과 지식그래프, Text-to-Query의 흐름으로 구분된다. 본 장에서는 각 연구 흐름을 간략히 검토한 뒤, 본 연구의 위치와 실험적 차별성을 표 1과 표 2에서 정리한다.

2.1 머신리더블·AI-Ready 공공데이터 연구

공공데이터의 인공지능 활용 가능성을 높이기 위해서는 단순한 파일 공개를 넘어 데이터의 기계 판독성, 구조화 수준, 재사용성을 함께 고려해야 한다. 김학래는 국내 공공데이터 정책이 양적 개방 확대에도 불구하고 실제 활용성과 재사용성 측면에서 한계를 보인다고 지적하였다[2]. 행정안전부와 한국지능정보사회진흥원은 「공공데이터의 인공지능 친화적 관리 가이드라인」을 통해 기존 문서 중심 데이터 제공 관행을 개선하고, 오픈 포맷, 명시적 스키마, 기계 판독 가능한 메타데이터 제공을 주요 원칙으로 제시하였다[3]. 이는 공공데이터가 사람이 읽는 문서 중심 제공 방식에서 벗어나, 인공지능 시스템이 검색·연산·추론에 활용할 수 있는 구조로 전환될 필요가 있음을 보여준다.

데이터의 기계 판독성과 재사용성을 높이기 위한 일반 원칙은 FAIR 원칙과 링크드데이터 논의를 중심으로 발전해 왔다. Wilkinson 등은 데이터가 발견 가능하고, 접근 가능하며, 상호운용 가능하고, 재사용 가능해야 한다는 FAIR 원칙을 제시하여 데이터 관리와 공유의 규범적 기준을 정립하였다[4]. Berners-Lee[5]는 5-star Linked Data 모델을 통해 웹 환경에서 데이터 개방성과 기계 판독 가능성을 단계적으로 설명하였다. 다만 이러한 논의는 주로 데이터 발행 원칙과 관리 방향을 제시하는 데 초점을 두며, 동일한 원천 데이터를 서로 다른 표현 형식으로 변환했을 때 실제 질의 가능성이 어떻게 달라지는지를 직접 비교하지는 않는다.

2.2 검색증강생성 및 구조화 검색 연구

검색증강생성은 대규모 언어모델이 사전학습 과정에서 보유한 지식만으로는 최신 정보나 도메인 특화 지식을 충분히 반영하기 어렵다는 한계를 보완하기 위해 발전해 왔다. Lewis et al.[6]은 외부 문서 검색 결과를 생성 모델의 입력으로 결합하는 검색증강생성 구조를 제안하여 지식집약적 질의응답 연구의 주요 흐름을 형성하였다. 이후 Gao et al.[7]은 검색증강생성 연구 흐름을 기본형, 고도화형, 모듈형 구조로 체계화하며 검색, 재순위화, 생성, 평가 모듈의 발전 방향을 정리하였다. 이러한 연구들은 문서 기반 질의응답 성능을 향상시키는 데 중요한 기반을 제공하였으나, 원천 데이터의 표현 형식 자체가 질의 가능성에 미치는 영향을 데이터 발행자 관점에서 비교하는 데에는 상대적으로 제한적이었다.

최근 연구는 단순한 텍스트 검색을 넘어 데이터의 구조적 특성을 활용하는 방향으로 확장되고 있다. Edge et al.[8]은 단순 텍스트 조각 검색만으로는 개체 간 관계나 전체 맥락을 충분히 반영하기 어렵다는 점을 지적하고, 개체 관계망을 활용한 그래프 기반 검색증강생성 방식을 제안하였다. Yu et al.[9]은 표 데이터를 단순 텍스트로 변환할 경우 행과 열이 형성하는 의미 구조가 훼손될 수 있음을 지적하고, 표 구조를 고려한 질의 및 추론 방식을 제시하였다. 이러한 연구들은 질의응답 성능이 검색 알고리즘뿐 아니라 데이터가 어떤 구조로 표현되고 활용되는지에 영향을 받을 수 있음을 보여준다[8][9].

2.3 RDF/SPARQL과 지식그래프 연구

공공행정 데이터와 같이 분류체계, 기관 관계, 수치 정보가 결합된 데이터는 단순 문서 검색만으로는 복합적인 질의에 대응하기 어렵다. 이와 관련하여 RDF와 지식그래프 연구는 데이터 간 의미 관계를 명시적으로 표현하고 질의하기 위한 기술적 기반을 제공해 왔다. RDF는 정보를 주어-술어-목적어 triple의 집합인 그래프로 표현하기 위한 표준 데이터 모델이며[10], SPARQL은 RDF 그래프에 대한 구조화 질의를 수행하기 위한 표준 질의 언어이다[11]. 또한 지식그래프는 개체와 관계를 명시적으로 표현하여 의미 기반 탐색과 관계 질의를 가능하게 하는 데이터 표현 방식으로 논의되어 왔다[12].

재정 및 공공데이터 영역에서도 RDF와 온톨로지 기반 접근은 데이터의 연결성과 질의 가능성을 높이는 방식으로 연구되어 왔다. Höffner[13]은 OpenBudgets.eu 플랫폼을 통해 예산 데이터를 분산형 오픈소스 데이터 관리 구조로 표현하는 방안을 제시하였고, Cifuentes-Silva et al.[14]은 국가 예산 데이터를 의미 기반 지식 구조로 변환하여 재정 데이터가 기계적으로 탐색 가능한 지식 구조가 될 수 있음을 보였다. 이러한 예산·공공데이터 지식그래프 사례는 공공데이터가 단순 파일 공개를 넘어 의미 연결과 구조화 질의가 가능한 형태로 발전할 수 있음을 보여준다[13][14]. 다만 기존 연구들은 예산 데이터의 지식그래프화 가능성을 보였다는 점에서 중요하지만, 동일 원천 데이터를 Plain Text, JSON, RDF, XLSX 등 복수의 표현 형식으로 변환하여 질의 가능성을 비교하는 실험 설계와는 구분된다.

2.4 자연어-구조화 질의 변환(Text-to-query) 연구

자연어 질의를 구조화된 질의나 실행 가능한 연산으로 변환하려는 연구도 함께 발전해 왔다. Yu et al.[15]은 Spider 벤치마크를 통해 복잡한 스키마와 다양한 도메인을 포함하는 자연어-구조화 질의 평가 환경을 제시하였다. 국내 행정 데이터 환경에서도 템플릿과 벡터 검색을 결합하여 한국어 자연어 질의를 도메인 특화 구조 질의로 변환하려는 연구가 제안되었다[16]. 이러한 연구들은 자연어 질의 처리가 단순 문장 이해를 넘어 스키마 연결, 조건 해석, 집계 연산, 실행 정확도와 밀접하게 관련됨을 보여준다[15][16]. 그러나 Text-to-Query 연구는 주로 자연어 질문을 SQL, pandas 연산, 또는 도메인 특화 구조 질의로 변환하는 문제에 초점을 둔다. 본 연구는 자연어-구조화 질의 변환 자체의 성능을 평가하기보다는, 동일한 공공 행정 원천자료가 서로 다른 표현 형식과 활용 도구에 결합될 때 질의 가능성이 어떻게 달라지는지를 비교한다는 점에서 차이가 있다.

2.5 선행연구와 본 연구의 차별성

이상의 선행연구를 종합하면, 기존 연구들은 공공데이터 발행 원칙의 정립, 검색증강생성 알고리즘의 고도화, 구조 기반 검색, 지식그래프 기반 데이터 연결, 자연어-구조화 질의 변환 등 개별 연구 흐름 안에서 발전해 왔다. 표 1은 이러한 선행연구 흐름의 주요 초점과 본 연구와의 관련성을 정리한 것이다. 본 연구는 기존 연구 흐름을 대체하기보다, 공공 행정문서의 AI 활용성을 데이터 표현 형식과 활용 도구의 결합이라는 관점에서 보완적으로 검증하는 데 초점을 둔다.

표 1. 선행연구 흐름과 본 연구의 위치
Table 1. Research streams and positioning of this study

Research stream	Representative studies	Main focus	Relevance to this study
AI-ready public data and machine-readable data	Kim [2], MOIS·NIA [3], Wilkinson et al. [4], Berners-Lee [5]	Public-data utilization gaps and principles for machine-readability, interoperability, and reusability	Provides the policy and normative background for AI-ready public data structuring
RAG studies	Lewis et al. [6], Gao et al. [7]	Combining retrieved external documents with generative models and improving retrieval and generation modules	Provides the basis for defining the plain-text vector retrieval baseline
Structure-aware RAG and table querying	Edge et al. [8], Yu et al. [9]	Using graph-based retrieval and table-aware reasoning for question answering	Supports the need to preserve structural information in administrative documents
RDF/SPARQL and knowledge graphs	W3C RDF [10], SPARQL [11], Hogan et al. [12]	Representing entities and relations as graphs and enabling structured queries over RDF data	Provides the technical foundation for the BEDO ontology and RDF-based query execution
Budget/public-data knowledge graphs	Höffner et al. [13], Cifuentes-Silva et al. [14]	Transforming budget data into semantic and queryable knowledge structures	Shows that budget data can be represented and explored as structured knowledge
Text-to-query	Yu et al. [15], Kim et al. [16]	Translating natural-language questions into SQL or domain-specific structured queries	Provides the basis for interpreting the pandas agent and structured-query settings

특히 예산설명서와 같은 반정형 공공문서는 사업명, 소관 부처, 시행기관, 분야·부문, 예산액, 지원형태 등 구조화 가능한 정보를 포함하므로, 단순 텍스트 검색만으로는 관계 탐색, 복합 조건, 집계 질의를 안정적으로 처리하기 어렵다. 이에 본 연구는 동일한 2026년 정부 예산설명서 원천 자료를 Plain Text, JSON, RDF, XLSX의 네 가지 표현 형식으로 변환하고, 각 형식에 적합한 다섯 가지 활용 방식을 구성하여 질의 가능성을 비교한다. 표 2는 기존 연구의 공백과 이에 대응하는 본 연구의 실험 설계를 요약한 것이다.

표 2에서 보듯이 본 연구는 자연어 입력 기반 방식의 현실 성능과 구조 질의 조건이 제공되었을 때의 도달 가능 성능을 구분하여 분석하고, 공공데이터의 인공지능 활용성이 데이터 형식 자체보다 표현 형식과 활용 도구의 적합성에 의해 좌우될 수 있음을 실증적으로 검토한다.

III. 연구 문제

3.1 연구 배경

최근 공공데이터 정책은 단순히 데이터를 공개하는 수준을 넘어, 인공지능이 직접 이해하고 활용할 수 있는 형태로 데이터를 제공하는 방향으로 전환하고 있다. 특히 공공데이터의 인공지능 친화적 관리를 위해서는 오픈 포맷, 명시적 스키마, 기계 판독 가능한 메타데이터, 구조화된 데이터 제공 방식이 필수적이다[3]. 이러한 흐름은 공공데이터가 사람이 읽는 문서 중심의 제공 방식에서 벗어나, 인공지능 시스템이 검색·연산·추론에 활용할 수 있는 구조로 전환되어야 함을 의미한다.

정부 예산설명서는 이러한 문제를 검토하기에 적합한 공공 행정문서이다. 예산설명서는 개별 사업 단위로 작성되며, 사업명, 소관 부처, 분야·부문, 프로그램·단위사업, 시행기관, 지원형태, 연도별 예산액 등 반복적으로 등장하는 구조화 가능 항목을 포함한다[17]. 그러나 실제 예산설명서는 주로 사람이 읽고 검토하기 위한 문서 형식으로 작성되기 때문에, 인공지능이 사업 간 관계를 탐색하거나 조건에 맞는 사업을 필터링하고, 예산액을 집계하는 데에는 한계가 있다. 즉, 문서 내부에는 구조화 가능한 정보가 존재하지만, 그 정보가 곧바로 질의 가능한 데이터 구조로 제공되는 것은 아니다.

최근 검색증강생성 방식은 공공 행정문서의 질의응답에 활용될 수 있는 유력한 접근으로 주목받고 있다. 그러나 일반적인 텍스트 기반 벡터 검색 방식은 문서의 의미적 유사성에 기반하여 관련 문단을 찾아내는 데에는 유용하지만, 예산설명서와 같이 정형 항목, 기관 관계, 계층 분류, 수치 정보가 결합된 문서에서 복합 조건 질의나 집계 질의를 안정적으로 처리하는 데에는 한계가 있을 수 있다.

표 2. 본 연구의 실험적 차별성
Table 2. Experimental differentiation of this study

Axis	Gap in prior studies	This study
Source data	Limited empirical use of real administrative templates [2]	Uses 533 AI budget explanation documents
Domain	Limited focus on Korean government budget explanations [13], [14]	Focuses on Korean government budget documents
Representation format	Limited controlled comparison across machine-readable and semantic formats [4], [5], [10]	Converts the same data into Plain Text, JSON, RDF, and XLSX
Query interface	Limited separation of natural-language and structured-query settings [6], [7], [15], [16]	Separates natural-language input from query_logic-based structured conditions
Tool - representation pairing	Limited analysis of how formats interact with retrieval, table, and graph tools [8], [9], [11]	Compares five representation-tool combinations
Evaluation design	Lack of budget-domain queryability benchmarks [13], [14]	Uses a 100-question set across four query types with paired bootstrap
Public artifacts	Reusable evaluation artifacts and schema resources are limited or vary across studies	Releases the ontology, evaluation set, predictions, scripts, and bootstrap results

예를 들어 “특정 시행기관이 수행하는 사업의 총 예산”, “출연 형태로 지원되는 보건 분야 사업”, “특정 부처 소관 사업과 시행기관의 관계”와 같은 질의는 단순 문장 검색만으로는 충분하지 않으며, 데이터의 구조와 질의 도구가 함께 고려되어야 한다.

한편 JSON, RDF, XLSX와 같은 형식은 텍스트 문서보다 기계가 처리하기 쉬운 구조를 제공할 수 있다. 그러나 특정 데이터 형식이 기계 판독 가능하다는 사실만으로 실제 질의응답 성능이 자동으로 향상된다고 보기는 어렵다. 예를 들어 JSON 형식은 계층 구조를 표현할 수 있지만, 이를 단순 문자열로 변환하여 벡터 검색에 투입하면 구조적 장점이 충분히 활용되지 않을 수 있다. 반대로 RDF는 개체 간 관계 탐색에 적합하고, XLSX는 조건 필터링과 집계 연산에 유리하지만, 각 형식의 장점은 적절한 질의 도구와 결합될 때에만 실질적인 성능으로 이어질 수 있다.

따라서 공공데이터의 인공지능 활용성을 평가하기 위해서는 “어떤 형식이 더 기계 판독 가능한가”라는 형식 중심의 논의만으로는 충분하지 않다. 동일한 원천 자료를 여러 표현 형식으로 변환하고, 각 형식에 적합한 도구를 결합했을 때 실제 질의 성능이 어떻게 달라지는지를 비교할 필요가 있다. 본 연구는 이러한 문제의식에 따라 2026년 인공지능 관련 정부 재정사업 예산설명서 533건을 대상으로, 데이터 표현 형식과 활용 도구의 결합이 질의 가능성에 미치는 영향을 실증적으로 검토하고자 한다.

3.2 연구 문제

본 연구의 목적은 공공 예산설명서와 같은 반정형 행정문서에서 데이터 표현 형식과 활용 도구의 결합이 인공지능 기반 질의응답 성능에 어떠한 영향을 미치는지를 분석하는 것이다. 이를 위해 동일한 원천 데이터를 Plain Text, JSON, RDF, XLSX의 네 가지 표현 형식으로 변환하고, 각 형식에 적합한 다섯 가지 활용 방식을 구성하여 질의 가능성을 비교한다.

본 연구는 특히 공공데이터의 인공지능 활용성이 데이터 형식 자체에 의해 결정되는 것이 아니라, 표현 형식과 활용 도구의 적합성에 의해 좌우될 수 있다는 가설에 주목한다. 이에 따라 자연어 입력 기반 방식과 구조 질의 조건 기반 방식을 구분하고, 질의 유형별 성능 차이와 오류 양상을 함께 분석한다. 본 연구에서 설정한 연구 문제는 표 3과 같다.

표 3. 본 연구의 연구 문제
Table 3. Research questions of this study

Research question	Description
RQ1	How does AI-based query answering performance differ when the same budget explanation data are transformed into different representation formats?
RQ2	How do natural-language input approaches and structured-query-condition approaches differ in multi-hop relation, reverse lookup, complex filtering, and aggregation queries?
RQ3	Is the AI utilization of public data influenced more by representation-tool fit than by data format itself?

RQ1은 데이터 표현 형식의 차이가 질의응답 성능에 미치는 영향을 확인하기 위한 것이다. 본 연구는 동일한 예산설명서 원천 자료를 Plain Text, JSON, RDF, XLSX 형식으로 변환하고, 각 형식이 질의 수행 과정에서 어떤 차이를 보이는지 비교한다. 이를 통해 단순 문서형 데이터, 계층형 데이터, 그래프형 데이터, 표형 데이터가 공공 행정문서 질의에서 갖는 상대적 특성을 분석한다.

RQ2는 입력 방식과 질의 유형에 따른 성능 차이를 분석하기 위한 것이다. 본 연구에서 자연어 입력 기반 방식은 사용자의 질문 문장을 그대로 입력받아 답변을 산출하는 방식이며, 구조 질의 조건 기반 방식은 평가셋에 정의된 정답 도출 논리를 활용하여 RDF 또는 XLSX 데이터에 대해 질의를 수행하는 방식이다. 두 계열은 입력 조건이 다르므로 단순한 절대 우열 비교의 대상은 아니며, 자연어 입력 환경의 현실 성능과 구조화된 질의 조건이 제공되었을 때의 도달 가능 성능을 구분하여 해석한다.

RQ3은 본 연구의 핵심 가설인 표현-도구 적합성을 검증하기 위한 것이다. 기계 판독 가능한 데이터 형식이라도 이를 해석하고 활용할 수 있는 적절한

도구와 결합되지 않으면 실제 질의 성능은 제한될 수 있다. 반대로 데이터의 구조적 특성과 활용 도구가 적절히 결합될 경우, 복합 조건 질의나 집계 질의와 같이 일반적인 텍스트 검색이 취약한 문제에서도 더 높은 질의 가능성을 확보할 수 있다. 본 연구는 이러한 가설을 실험 결과와 오류 분석을 통해 검토한다.

Ⅳ. 방법론

4.1 데이터셋

본 연구의 원천 데이터는 2026년 정부 예산설명서 중 인공지능 관련 재정사업으로 식별된 533건의 사업 정보이다. 해당 자료는 정부 예산 사업 설명자료의 표준 항목 체계를 따르며, 사업명, 소관 부처, 분야·부문, 프로그램·단위사업, 시행기관, 지원형태, 연도별 예산액 등 반복적으로 등장하는 행정·재정 항목을 포함한다. 예산설명서는 개별 사업의 목적과 내용뿐 아니라, 해당 사업이 어떤 예산 분류 체계에 속하고, 어느 부처가 소관하며, 실제 수행 주체가 누구인지 등을 함께 제시한다는 점에서 단순한 설명문이 아니라 구조화 가능한 행정 데이터의 성격을 가진다.

정부 예산설명서는 개별 사업 단위로 작성되지만, 그 내부에는 반복적으로 등장하는 구조화 가능 항목이 존재한다. 본 연구는 533건의 예산설명서를 분석하여 사업별 질의응답에 필요한 8개의 핵심 의미 단위를 도출하였다. 구체적으로는 재정사업, 소관 부처, 분야, 부문, 프로그램, 단위사업, 시행기관, 지원형태를 핵심 엔티티로 설정하였다. 이 핵심 엔티티들은 개별 재정사업을 중심으로 소관 부처, 시행기관, 분야·부문, 프로그램·단위사업, 지원형태를 연결하는 구조를 형성하며, 예산설명서 내의 행정적 위계와 사업 수행 관계를 명확히 표현한다. 따라서 이 핵심 엔티티들은 단순 항목 분류에 그치지 않고, 다단계 관계 질의, 역방향 탐색, 복합 조건 필터링, 집계 질의를 수행하도록 데이터 구조의 기반을 제공한다.

또한 본 연구는 도출된 핵심 엔티티가 인공지능 관련 재정사업에만 특화된 구조인지 확인하기 위해, 이질적인 정책 분야의 예산설명서 10건을 추가로 검토하였다. 검토 대상에는 인공지능 분야와 직접 관련되지 않은 다양한 정책 영역의 예산설명서를 포함하였으며, 각 문서에서 사업코드, 프로그램·단위사업 체계, 지원형태, 소관·시행주체 정보, 예산 총괄표 등이 반복적으로 나타나는지를 확인하였다. 검토 결과, 이러한 항목들은 정책 분야가 달라지더라도 공통적으로 확인되었으며, 예산설명서가 일정한 행정·재정 서술 구조를 공유하고 있음을 확인할 수 있었다. 이에 따라 본 연구에서 도출한 핵심 엔티티는 인공지능 재정사업뿐 아니라 예산설명서 일반의 반복 구조를 일정 수준 반영하는 것으로 보았다.

다만 본 연구의 실험 데이터셋은 2026년 인공지능 관련 재정사업 533건을 중심으로 구성되었으므로, 전체 정부 예산사업을 대표하는 범용 데이터셋으로 해석하기보다는 공공 예산설명서 도메인에서 데이터 표현 형식과 활용 도구의 결합 효과를 비교하기 위한 도메인 특화 데이터셋으로 보는 것이 적절하다. 이러한 설정은 본 연구의 목적이 모든 예산사업에 대한 일반 성능을 산출하는 데 있는 것이 아니라, 동일한 행정 원천 자료를 여러 표현 형식으로 변환했을 때 질의 가능성이 어떻게 달라지는지를 통제된 조건에서 비교하는 데 있음을 의미한다.

4.2 접근 방법

본 연구는 동일한 예산설명서 데이터를 여러 표현 형식으로 변환하고, 각 형식에 적합한 활용 방식을 결합하여 질의 가능성을 비교한다. 이를 위해 첫째, 예산설명서의 핵심 의미 단위를 기반으로 인공지능 친화적 스키마와 도메인 온톨로지를 설계하였다. 둘째, 동일한 원천 데이터를 문서형, 계층형, 표형, 그래프형 데이터로 변환하였다. 셋째, 자연어 입력 기반 방식과 구조 질의 조건 기반 방식을 구분하여 5가지 활용 방식을 구성하였다. 넷째, 100문항 평가셋과 통합 채점 기준을 통해 각 방식의 성능을 비교하였다.

4.2.1 AI-Ready 스키마 및 도메인 온톨로지 설계

본 연구는 예산설명서의 반복 구조를 인공지능이 활용 가능한 형태로 표현하기 위해 '예산설명서 도메인 온톨로지 v1.0'을 설계하였다. 이 온톨로지는 예산설명서 도메인에 특화된 경량 온톨로지로, 국가재정운용 체계와 행정 분류 구조를 반영한 클래스와 속성으로 구성되며 본 연구가 도출한 8개 핵심 엔티티는 표 4와 같다.

표 4. 예산설명서의 핵심 엔티티
Table 4. Core entities of the budget explanation

Entity	Description	Example attributes
Project	Unit of fiscal project information	Project name, project type
Ministry	Supervising ministry	Ministry name
Implementing agency	Agency responsible for implementation	Agency name
Field	Policy field	Field name
Sector	Policy sector	Sector name
Program	Program-level classification	Program name
Subprogram	Subprogram-level classification	Subprogram name
Support type	Fiscal support type	Subsidy, contribution, loan, etc.

온톨로지의 중심 클래스는 재정사업(Project)이다. 각 재정사업은 소관 부처, 시행기관, 분야, 부문, 프로그램, 단위사업, 지원형태를 서로 연결한다. 이를 통해 예산설명서의 주요 정보를 단순 텍스트가 아니라 개체와 관계의 구조로 표현한다. 주요 관계에는 소관 부처와의 관계, 시행기관과의 관계, 분야·부문 분류, 프로그램·단위사업 체계, 지원형태 등이 있다. 또한 예산액, 사업기간, 사업목적과 같은 값은 개별 사업의 속성값으로 설정했다

본 연구의 온톨로지는 복잡한 추론 기능을 강조하기보다, 실제 질의와 데이터 변환에 활용 가능한 경량 구조를 지향한다. 이는 공공 행정문서에서 자주 요구되는 조건 검색, 기관 관계 탐색, 예산 집계, 분류 체계 기반 질의를 안정적으로 수행하기 위한 설계이다. 특히 소관 부처와 시행기관을 분리하고, 분야→부문→프로그램→단위사업의 예산 분류 체계를 명시함으로써 예산설명서 도메인 특유의 행정적 위계와 기관 관계를 표현할 수 있도록 하였다.

4.2.2 다중 표현 형식 변환

본 연구는 동일한 533건의 예산설명서 원천 데이터를 표 5와 같이 네 가지 표현 형식으로 변환하였다. 변환 형식은 문서형 텍스트, JSON 직렬화, XLSX 표, RDF 그래프이다. 각 형식은 동일한 핵심 정보를 포함하지만, 데이터 구조를 표현하는 방식과 활용 가능한 질의 도구가 서로 다르다.

표 5. 원천 데이터의 다중 표현 형식 변환
Table 5. Multi-format conversion of the source data

Format	Unit and structure	Purpose of use
Plain text	One text bundle per project	Baseline for general document-based vector retrieval
JSON (Serializat-ion)	List of 533 project objects	Control condition for searching hierarchical data without structure-aware tools
XLSX table	Normalized into a project-level table and an implementing-agency-level table	Conditional filtering, table joins, and aggregation operations
RDF graph	533 fiscal project instances and related resources for agencies, classifications, and support types	Entity-relation traversal and structured querying

문서형 텍스트는 사업명, 소관 부처, 분야, 프로그램, 지원형태, 시행기관, 예산액, 사업목적 등 주요 정보를 사업 단위로 묶어 생성했다. 이 형식은 일반적인 문서형 검색증강생성(RAG) 방식의 기준선(Baseline) 역할을 한다.

JSON 직렬화 형식은 각 사업의 핵심 필드를 객체 형태로 구성한 뒤, 이를 문자열로 변환하여 벡터 검색에 투입하였다. 본 연구에서 JSON 방식은 JSON 형식 자체의 일반적 성능을 평가하기 위한 것이 아니라, 기계 판독 가능한 형식이라도 구조 인식 도구와 결합되지 않으면 질의 가능성이 제한될 수 있음을 확인하기 위한 대조군으로 설계하였다.

따라서 JSONPath, 키-값 조회, 스키마 인식 검색 등 JSON 구조를 직접 활용하는 방식은 본 연구의 검증 대상에서 제외하였다.

XLSX 표 형식은 사업 단위 표와 시행기관 단위 표로 정규화하였다. 사업 단위 표는 사업명, 소관 부처, 분야·부문, 프로그램·단위사업, 지원형태, 예산액 등 사업별 속성을 포함한다. 시행기관 단위 표는 하나의 사업에 복수 시행기관이 연결될 수 있는 경우를 고려하여 별도 표로 구성하였다. 이 구조는 조건 필터링, 표 간 결합, 예산 합계, 사업 개수 산출과 같은 표 기반 연산으로 작동한다.

RDF 그래프 형식은 앞서 설계한 예산설명서 도메인 온톨로지를 적용해 구축했다. 최종 그래프는 533개의 재정사업 인스턴스를 포함하며, 각 인스턴스를 소관 부처, 시행기관, 분야, 부문, 단위사업, 지원형태 등의 리소스와 유기적으로 연결한다. 이 형식은 여러 단계를 거치는 관계 탐색, 특정 조건에서의 역방향 탐색, 분류 체계 기반 질의를 수행하는 데 탁월한 성능을 발휘한다.

4.2.3 5가지 질의 방식 설계

본 연구는 데이터 표현 형식과 활용 도구의 결합 방식이 질의 성능에 미치는 영향을 비교하기 위해 표 6과 같이 5가지 질의 방식을 구성하고, 이를 입력 조건에 따라 자연어 입력 기반 방식과 구조 질의 조건 기반 방식으로 구분하였다.

자연어 입력 기반 방식은 사용자의 질문 문장을 그대로 입력받아 답변을 생성하는 방식이다. 여기에 해당하는 방식은 문서형 벡터 검색, JSON 직렬화 벡터 검색, Pandas Agent이다. 이 방식들은 실제 사용자 질문 환경에 가깝지만, 자연어 해석 과정에서 조건 누락, 기관명 해석 오류, 컬럼 선택 오류 등이 발생할 수 있다.

구조 질의 조건 기반 방식은 자연어 질문이 아니라 평가셋에 정의된 정답 도출 논리를 입력으로 사용한다. 여기에 해당하는 방식은 RDF 기반 SPARQL 질의와 XLSX 질의 방식이다. 이 방식들은 일반적인 자연어 질의응답 성능을 의미하지 않으며, 각 표현 형식이 적절한 질의 조건을 제공받았을 때 도달 가능한 질의 성능을 확인하기 위한 준상한 비교로 해석한다.

4.2.4 평가셋 및 채점 기준

본 연구의 평가셋은 정부 예산설명서 도메인에 특화된 질의 가능성 평가셋으로, 표 7과 같이 총 100문항으로 구성하였다 평가셋은 특정 범용 모델의 질의응답 성능을 측정하기 위한 일반 벤치마크가 아니라, 동일한 예산설명서 데이터를 여러 표현 형식과 도구 결합으로 활용했을 때 성능이 어떻게 달라지는지 비교하기 위한 도메인 특화 평가셋이다.

표 6. 5가지 질의 방식의 입력 조건 및 실행 방식
Table 6. Input conditions and execution methods of the five approaches

Approach	Input	Execution method	Evaluation meaning
Plain text vector RAG	Natural-language question + plain-text source	Generate an answer using an LLM after vector retrieval	Real-world performance of document-based Retrieval-Augmented Generation (RAG)
JSON-serialized vector retrieval	Natural-language question + JSON serialization	Generate an answer after retrieval over JSON-serialized text	Performance of machine-readable data without structure-aware tools
Pandas agent	Natural-language question + tabular schema	Generate a query plan and execute table operations	Real-world performance of converting natural language into structured table operations
RDF-based SPARQL query	Structured query condition + RDF graph	Execute SPARQL over the query template	Quasi-upper-bound performance of graph-based relation queries
XLSX query	Structured query condition + tabular data	Execute filtering, joins, and aggregation operations	Quasi-upper-bound performance of table-based structured queries

표 7. 질의 유형별 평가 문항 구성
Table 7. Evaluation questions by query type

Query type	Number of questions	Description
Multi-hop relation query	30	Questions that traverse relationships among implementing agencies, supervising ministries, and projects
Reverse lookup	30	Questions that identify projects or agencies satisfying a specific condition
Complex filtering	25	Questions that combine multiple conditions such as ministry, field, support type, and budget amount
Aggregation query	15	Questions requiring quantitative calculations such as project counts or budget sums

평가 문항은 질의 유형에 따라 다단계 관계 질의, 역방향 탐색, 복합 조건 필터링, 집계 질의로 구분하였다. 각 유형은 예산설명서 데이터에서 실제로 발생할 수 있는 질의 패턴을 반영한다.

평가 문항의 유형별 비중은 예산설명서 도메인의 질의 빈출 패턴을 반영하여 설계하였다. 다단계 관계 질의와 역방향 탐색은 예산설명서가 가진 소관부처-시행기관-사업의 다층적 관계 구조에서 가장 빈번하게 발생하는 질의 패턴이므로 각 30문항으로 우선 배분하였다. 복합 조건 필터링은 부처·분야·지원형태·예산액 등 다중 조건 결합 질의의 대표성을 확보하기 위해 25문항, 집계 질의는 도메인 특화 정량 계산의 비중을 고려해 15문항으로 구성하였다. 본 구성은 균등 분배를 따르지 않으며, 행정 실무에서 빈출되는 관계·역방향 질의에 가중치를 둔 도메인 특화 설계 선택의 결과이다.

또한 각 문항은 질문 표현의 변형 수준에 따라 정형 질문, 일반 질문, 변형 질문으로 구분하였다. 정형 질문은 원본 예산서 용어와 정형 표현에 가까운 질문이며, 일반 질문은 실무자가 사용할 법한 자연스러운 질문이다. 변형 질문은 축약, 구어적 표현, 어순 변경 등을 포함한 상대적으로 어려운 질문이다. 다만 이 세 수준은 동일 정답 도출 논리를 공유하는 반복 표현 집합이 아니라 독립 문항군으로 구성되었다. 따라서 본 연구에서의 변형 수준별 비교는 자연어 노이즈에 대한 엄밀한 인과 검정이 아니라, 문항군 간 성능 추이를 살펴보기 위한 보조 분석으로 한정한다.

각 문항은 기준 정답, 허용 정답 집합, 평가 모드, 답변 개수 제한, 구조 질의 조건, 정답 산출에 사용된 원천 컬럼을 포함한다. 허용 정답 집합은 답변 생성 단계에는 제공하지 않고, 채점 단계에서만 사용하였다. 구조 질의 조건은 RDF 기반 질의와 XLSX 기반 실행 방식에서 정답 도출 논리로 활용하였다.

채점 전에는 모든 방식의 답변을 동일한 기준으로 정규화하였다. 정규화 과정에서는 기관명과 부처명의 약어·별칭을 정식 명칭으로 변환하고, 백만원·억원 등 예산 단위를 비교 가능한 형태로 통일하였다. 수치형 답변은 숫자 추출 과정을 거쳐 비교하였고, 목록형 답변은 정답 집합과 예측 집합의 교집합을 기준으로 평가하였다. 전체 성능은 정밀도(Precision), 재현율(Recall), F1 점수(F1 score)를 중심으로 산출하였다. 본 연구에서 정밀도는 예측 답변 중 기준 정답에 포함된 항목의 비율을 의미하며, 재현율은 기준 정답 중 예측 답변이 맞힌 항목의 비율을 의미한다. F1 점수는 정밀도와 재현율의 조화평균으로, 각 질의 방식이 산출한 답변과 기준 정답 간의 일치 정도를 종합적으로 나타낸다. 따라서 F1 점수는 단일 정답 문항뿐 아니라 복수의 사업명, 기관명, 부처명 등을 요구하는 목록형 문항에서 부분 일치 정도를 함께 반영한다.

평가 모드는 세 가지로 구분하였다. 첫째, 단일 값 일치 방식은 예산액이나 사업 개수와 같이 하나의 값이 정답인 문항에 적용하였다. 둘째, 집합 일치 방식은 복수의 사업명, 기관명, 부처명 등을 요구하는 문항에 적용하였다. 셋째, 후보군 일부 적중 방식은 가능한 정답 후보가 넓고 제한된 개수의 답변을 허용하는 문항에 적용하였다. 모든 방식은 답변 생성 단계에서 기준 정답을 참조하지 않도록 하여 평가의 객관성을 확보하였다.

방식 간 성능 차이의 통계적 신뢰도를 확인하기 위해 문항별 F1 점수 벡터를 기준으로 대응표본 부트스트랩 검정을 수행하였다. 각 비교쌍에 대해

10,000회의 재표본추출을 수행하여 평균 차이의 95% 신뢰구간을 산출하였으며, 신뢰구간이 0을 포함하지 않는 경우 통계적으로 유의한 차이가 있는 것으로 해석하였다. 이 검정은 두 방식의 문항별 F1 점수를 동일한 문항 기준으로 짝지어 비교함으로써, 평가 문항 구성에 따른 변동성을 반영하여 방식 간 평균 성능 차이를 확인하기 위한 절차이다.

V. 실험 및 결과 분석

5.1 실험 환경

본 연구의 5가지 질의 방식은 동일한 조건에서 비교할 수 있도록 하나의 통합 실험 환경에서 실행하였다. 모든 방식은 동일한 평가셋, 동일한 정답셋, 동일한 채점 기준을 사용하였다. 답변 비교 전에는 기관명, 부처명, 예산 단위, 숫자 표현 등을 동일한 기준으로 정규화하였다. 예를 들어 기관·부처의 약어와 별칭은 정식 명칭으로 변환하였고, 백만원·억원 등 서로 다른 예산 단위는 비교 가능한 형태로 통일하였다. 또한 수치형 답변은 숫자 추출 과정을 거쳐 기준 정답과 비교하였다.

본 연구의 주요 실험 환경은 표 8과 같다. 벡터 검색 저장소로는 ChromaDB를 사용하였으며, RDF 처리는 SPARQL 1.1 질의가 가능한 rdflib 환경에서 수행하였다.

표 8. 실험 환경
Table 8. Experimental setup

Component	Setting
Embedding model	jhgan/ko-sroberta-multitask(Korean-specific)
LLM	GPT 4-o
Vector DB	ChromaDB(in-memory)
RDF processing	rdflib(SPARQL 1.1)
XLSX processing	pandas +openpyxl

본 연구는 실험의 투명성과 재현성을 높이기 위해 평가셋, 정답셋, 각 방식의 예측 결과, 부트스트랩 검정 결과, 실행 스크립트, 실행 환경 설정 파일을 함께 공개하였다(https://anonymous.4open.science/r/budget_explanation_queryability). 다만 본 연구에서 사용한 상용 생성 모델 API는 시간이 지나면서 모델 버전이나 내부 동작 방식이 변경될 수 있으므로, 동일한 코드를 다시 실행하더라도 완전히 같은 결과가 항상 보장되지는 않는다. 따라서 본 연구는 이미 저장된 예측 결과를 활용하여 본문에 제시된 F1 점수와 대응표본 부트스트랩 95% 신뢰구간을 다시 계산하는 경로를 함께 제공하였다. 이 경로는 외부 API를 호출하지 않기 때문에 본문 결과를 안정적으로 확인할 수 있다.

5.2 실험 결과 및 분석

본 연구의 100문항 평가셋에 대해 5가지 질의 방식을 비교한 결과는 표 9와 같다. 입력 방식의 차이를 고려하여 자연어 입력 기반 방식과 구조 질의 조건 기반 방식을 구분하여 제시하였다.

표 9. 질의 방식별 전체 성능 비교
Table 9. Overall performance of input methods

Input type	Method	Precision	Recall	F1
Natural-language input	Pandas agent	58.00%	53.17%	54.57%
Natural-language input	Plain text vector RAG	43.50%	30.67%	33.77%
Natural-language input	JSON-serialized Vector RAG	18.00%	13.83%	14.83%
Structured query condition	XLSX query	66.50%	64.67%	65.40%
Structured query condition	RDF-based SPARQL query	59.50%	57.67%	58.40%

자연어 입력 기반 방식에서는 Pandas Agent가 F1 54.57%로 가장 높은 성능을 보였다. 문서형 벡터 검색은 F1 33.77%를 기록하였고, JSON 직렬화 벡터 검색은 F1 14.83%로 가장 낮은 성능을 보였다. 이는 동일하게 자연어 질문을 입력으로 사용하더라도, 도구에 제공되는 구조 정보의 수준과 이를 활용하는 방식에 따라 실제 질의응답 성능이 크게 달라질 수 있음을 보여준다.

구조 질의 조건 기반 방식에서는 XLSX 질의가 F1 65.40%, RDF 기반 SPARQL 질의가 F1 58.40%

를 기록하였다. 두 방식은 자연어 질문을 직접 해석하지 않고 평가셋에 정의된 구조 질의 조건을 활용하므로, 자연어 입력 방식과 단순한 절대 우열 비교 대상으로 해석하기는 어렵다. 다만 동일 평가셋에서 구조화된 표현 형식이 적절한 질의 조건과 결합될 때 도달 가능한 성능이 자연어 입력 방식보다 높은 수준에 위치함을 확인할 수 있다.

특히 JSON 직렬화 벡터 검색의 낮은 성능은 본 연구의 대조군 설계와 관련하여 중요한 의미를 가진다. JSON은 기계 판독 가능한 형식이지만, 본 실험에서는 JSON 구조를 직접 활용하지 않고 문자열로 변환하여 벡터 검색에 투입하였다. 그 결과 JSON 직렬화 벡터 검색은 문서형 벡터 검색보다도 낮은 성능을 보였다. 이는 데이터가 기계 판독 가능한 형식으로 제공되더라도, 해당 구조를 인식하고 활용하는 도구와 결합되지 않으면 실제 질의 가능성이 제한될 수 있음을 보여준다.

5.3 통계적 유의성 검정

방식 간 F1 차이에 대해 대응표본 부트스트랩 검정을 수행하였다. 각 비교는 동일한 100개 문항에 대한 두 방식의 F1 점수 벡터를 기준으로 하였으며, 10,000회의 재표본추출을 통해 평균 차이와 95% 신뢰구간을 산출하였다. 본 연구에서는 95% 신뢰구간이 0을 포함하지 않는 경우 두 방식 간 성능 차이가 통계적으로 유의한 것으로 해석하였다.

검정 결과, 표 10과 같이 XLSX 질의, RDF 기반 SPARQL 질의, Pandas Agent는 모두 문서형 벡터 검색보다 유의하게 높은 성능을 보였다. 반면 XLSX 질의와 RDF 기반 SPARQL 질의, XLSX 질의와 Pandas Agent, RDF 기반 SPARQL 질의와 Pandas Agent 간 차이는 95% 신뢰구간이 0을 포함하여 통계적으로 유의하지 않았다. 이는 구조화 정보를 활용하는 방식들이 문서형 벡터 검색보다 우수한 성능을 보였지만, 구조화 활용 방식들 사이에서는 특정 방식의 절대적 우위를 단정하기 어렵다는 점을 의미한다.

한편 JSON 직렬화 벡터 검색은 문서형 벡터 검색보다 유의하게 낮은 성능을 보였다. 이는 JSON 형식 자체의 일반적 한계를 의미하는 것이 아니라, 구조 인식 도구 없이 JSON을 단순 문자열로 처리했을 때의 한계를 보여주는 결과이다. 따라서 본 결과는 공공데이터의 인공지능 활용성을 높이기 위해 데이터 형식뿐 아니라 해당 형식을 해석하고 실행할 수 있는 도구의 결합이 중요하다는 점을 뒷받침한다.

표 10. 5가지 접근 방식별 F1 점수 차이의 통계적 비교
Table 10. Statistical comparison of F1 score differences across five approaches

Comparison	Mean difference	95% confidence interval	Significance	
			Raw	Adjusted
XLSX query – plain text vector retrieval	31.63	[20.40, 42.70]	Significant	Significant
RDF-based SPARQL query – plain text vector retrieval	24.63	[14.97, 34.40]	Significant	Significant
Pandas agent – plain text vector retrieval	20.8	[11.30, 30.57]	Significant	Significant
JSON-serialized vector retrieval – plain text vector retrieval	-18.93	[-26.60, -11.37]	Significant	Significant
XLSX query – RDF-based SPARQL query	7	[-1.13, 15.13]	Not significant	Not significant
XLSX query – pandas agent	10.83	[-1.57, 22.83]	Not significant	Not significant
RDF-based SPARQL query – pandas agent	3.83	[-7.77, 15.13]	Not significant	Not significant

Note: A 95% confidence interval excluding 0 was interpreted as a statistically significant difference.

5.4 질의 유형별 성능분석

질의 유형별 성능은 표 11과 같다. 평가 문항은 집계, 복합 조건, 다단계 관계, 역방향 탐색의 네 가지 유형으로 구분하였다. 각 방식은 질의 유형에 따라 서로 다른 강점을 보였다.

집계 질의에서는 XLSX 질의와 RDF 기반

SPARQL 질의가 상대적으로 안정적인 성능을 보였다. 반면 문서형 벡터 검색과 JSON 직렬화 벡터 검색은 집계 질의에서 F1 0.00%를 기록하였다. 이는 단순 문서 검색 방식이 사업 개수나 예산 합계와 같은 정량 연산을 안정적으로 수행하기 어렵다는 점을 보여준다.

복합 조건 질의에서는 XLSX 질의와 Pandas Agent가 높은 성능을 보였다. XLSX 질의는 구조 질의 조건을 표 데이터의 필터링과 결합 연산으로 변환하기 때문에 복수 조건 처리에 유리하였다. Pandas Agent 역시 자연어 질문을 표 연산 계획으로 변환할 수 있는 경우 복합 조건 질의에서 높은 성능을 보였다. 다만 자연어 해석 단계의 오류 가능성이 존재하므로, 구조 질의 조건을 직접 활용하는 XLSX 방식과는 평가 의미가 다르다.

다단계 관계 질의에서는 XLSX 질의가 가장 높은 성능을 보였으며, RDF 기반 SPARQL 질의도 높은 성능을 유지하였다. RDF 방식은 개체 간 관계를 명시적으로 표현하기 때문에 관계 탐색형 질의에 적합한 구조를 가진다. 다만 본 실험에서는 RDF 기반 SPARQL 질의가 사전에 정의된 질의 틀에 의존하므로, 질의 틀의 설계 범위에 따라 성능이 제한될 수 있다.

역방향 탐색에서는 Pandas Agent가 가장 높은 성능을 보였다. 이는 자연어 질문에서 특정 조건을 해석하고, 해당 조건에 맞는 사업이나 기관을 표 데이터에서 역으로 탐색하는 과정이 비교적 효과적으로 수행된 결과로 볼 수 있다. 그러나 Pandas Agent는 집계 질의에서 낮은 성능을 보였으므로, 모든 질의 유형에서 우수한 방식이라고 보기는 어렵다.

종합하면, 구조화 정보를 활용하는 방식들은 질의 유형별로 서로 다른 강점을 보였다. XLSX 질의는 집계와 다단계 관계 질의에서 강점을 보였고, RDF 기반 SPARQL 질의는 관계 탐색형 질의에서 비교적 안정적이었다. Pandas Agent는 복합 조건과 역방향 탐색에서 강점을 보였지만, 집계 질의에서는 한계를 보였다. 따라서 본 연구 결과는 단일 표현 형식이나 단일 도구가 모든 질의 유형에서 우세하다기보다, 질의 유형과 데이터 구조에 따라 적합한 표현-도구 조합이 달라질 수 있음을 시사한다.

표 11. 5가지 방식의 질의 유형별 F1 성능 비교
Table 11. Comparison of F1 performance across query types for five utilization approaches

Method	Aggregation	Complex filtering	Multi-hop relation	Reverse lookup
XLSX query	53.33%	61.07%	83.78%	56.67%
RDF-based SPARQL query	46.67%	41.33%	70.22%	66.67%
Pandas agent	13.33%	61.20%	39.78%	84.44%
Plain text vector retrieval	0.00%	15.07%	45.56%	54.44%
JSON-serialized vector retrieval	0.00%	3.20%	41.00%	5.78%

5.5 오류 진단 및 질문 변형 수준 분석

5.5.1 Pandas Agent 오류 진단

Pandas Agent의 저성능 사례를 분석하기 위해 표 12와 같이 F1 0.5 미만에 해당하는 심각한 실패 사례 40건을 대상으로 주요 오류 유형을 분류하였다. 오류 유형은 조건 해석 오류, 테이블·컬럼 선택 오류, 기관명·약어 해석 오류의 세 가지로 구분하였다.

조건 해석 오류는 자연어 질문에 포함된 조건을 질의 계획의 필터 조건으로 변환하는 과정에서 조건이 누락되거나 잘못 해석된 경우를 의미한다.

표 12. Pandas Agent 응답 실패 사례(F1<0.5, n=40)
Table 12. Pandas Agent severe failure (F1<0.5, n=40

Major error type	Count	Ratio (%)
Filter condition error	19	47.50%
Table/column selection error	11	27.50%
Institution/abbreviation interpretation error	10	25.00%
Total	40	100.00%

테이블·컬럼 선택 오류는 사업 단위 표와 시행기관 단위 표 중 적절한 테이블을 선택하지 못하거나, 예산액·기관명·사업명 등 필요한 컬럼을 잘못 참조

한 경우를 의미한다. 기관명·약어 해석 오류는 NIA, KISA, IITP 등 영문 약어가 정식 기관명으로 변환되지 않아 질의 조건과 실제 데이터가 제대로 연결되지 않은 경우를 의미한다. 한 문항에서 둘 이상의 오류가 동시에 관찰된 경우에는 답변 실패에 가장 큰 영향을 미친 주된 원인을 기준으로 하나의 유형에 분류하였다.

분석 결과, 조건 해석 오류가 전체 실패 사례의 47.5%를 차지하며 가장 빈번하게 발생했다. 이는 자연어 질문을 필터 조건으로 변환하는 단계의 실패가 Pandas Agent의 주요 성능 저하를 이끌었음을 보여준다. 특히 두 가지 원인이 결합하면서 집계 질의의 성능을 낮추는 결과로 이어졌다. 하나는 테이블·컬럼 선택 오류로 인해 합계나 개수 산출에 필요한 컬럼이 잘못 참조된 경우이고, 다른 하나는 조건 해석 오류로 인해 필터링 결과가 비어 있는 값으로 반환된 경우이다.

한편 다단계 관계 질의에서는 기관명·약어 해석 오류가 집중적으로 관찰되었다. 이는 영문 약어와 정식 기관명 사이의 연결 실패가 기관 관계를 따라가는 질의에서 중요한 병목으로 작용했음을 의미한다. 이러한 오류 분포는 Pandas Agent 방식 자체의 본질적 한계라기보다, 본 연구에서 구현한 프롬프트, 스키마 설명, 기관명 약어 변환 절차의 제약에서 비롯된 것으로 볼 수 있다. 따라서 후속 연구에서는 부처·기관 약어 사전을 보강하고, 질문에 적합한 테이블과 컬럼을 선택하는 절차를 정교화하며, 복합 조건을 여러 하위 질의로 나누어 처리하는 방식을 적용함으로써 본 절에서 확인된 오류 유형을 개선할 수 있을 것이다.

5.5.2 질문 변형 수준별 성능 분석

질문 표현의 변형 수준에 따른 성능 차이를 확인하기 위해 표 13과 같이 정형 질문, 일반 질문, 변형 질문으로 구분하여 F1 성능을 비교하였다. 다만 4.2.4절에서 설명한 바와 같이, 본 연구의 변형 수준별 문항군은 동일 정답 도출 논리를 공유하는 반복 표현 집합이 아니라 독립 문항군이다. 따라서 본 절의 비교는 자연어 노이즈의 단독 효과를 검증하는 인과 분석이 아니라, 문항군 간 성능 추이를 살펴보기 위한 보조 분석으로 해석한다.

자연어 입력 기반 방식에서는 질문 표현이 강하게 변형된 문항군에서 성능 저하가 크게 나타났다. Pandas Agent는 정형 질문과 일반 질문에서는 각각 59.22%, 66.60%를 기록하였으나, 변형 질문에서는 17.50%로 크게 하락하였다. 문서형 벡터 검색도 변형 질문에서 2.50%의 낮은 성능을 보였고, JSON 직렬화 벡터 검색은 변형 질문에서 0.00%를 기록하였다. 이는 자연어 입력 방식이 질문 표현의 변화와 약어, 구어적 표현, 어순 변경 등에 민감할 수 있음을 보여준다.

반면 구조 질의 조건 기반 방식인 XLSX 질의와 RDF 기반 SPARQL 질의는 변형 질문에서도 상대적으로 높은 성능을 유지하였다. 다만 이는 두 방식이 자연어 질문 자체를 직접 해석하지 않고, 평가셋에 정의된 구조 질의 조건을 입력으로 사용하기 때문에 나타난 결과이다. 따라서 이 결과를 자연어 표현 변화에 대한 강건성으로 직접 해석해서는 안 되며, 입력 조건 차이에 따른 성능 차이로 이해해야 한다.

이 분석은 향후 연구에서 보다 엄밀한 질문 변형 평가셋이 필요함을 보여준다. 동일한 정답 도출 논리에 대해 정형 질문, 일반 질문, 변형 질문을 짝지어 구성한다면, 자연어 노이즈가 각 방식의 성능에 미치는 영향을 더 명확하게 분리하여 검증할 수 있을 것이다.

표 13. 질문 변형 수준별 답변 성능 결과
Table 13. F1 Performance by question variation level

Method	Formal question	General question	Modified question
XLSX query	80.00%	59.20%	59.00%
RDF-based SPARQL query	65.33%	57.60%	50.00%
Pandas agent	59.22%	66.60%	17.50%
Plain text vector retrieval	32.50%	47.00%	2.50%
JSON-serialized vector retrieval	27.78%	13.00%	0.00%

5.6 한계점

본 연구는 정부 예산설명서의 AI-ready 구조화

가능성과 표현-도구 결합별 질의 가능성을 실증적으로 비교하였으나, 다음과 같은 한계를 갖는다.

첫째, 본 연구의 외적 타당성은 정부 예산설명서 도메인으로 제한한다. 예산설명서는 사업명, 소관부처, 시행기관, 분야·부문, 지원형태, 연도별 예산액 등 반복적으로 등장하는 표준 항목을 포함하므로 구조화 가능성이 비교적 높은 행정문서이다. 따라서 본 연구의 결과를 회의록, 보도자료, 민원 문서, 법령 해석 문서와 같이 비정형성이 높거나 서술 구조가 다른 공공문서 전반으로 일반화하기 위해서는 추가 검증이 필요하다.

둘째, RDF 기반 SPARQL 질의와 XLSX 질의 결과는 자연어 질문을 직접 구조 질의로 변환하는 end-to-end 시스템 성능이 아니라, 질의 논리(query_logic)가 주어진 조건에서 각 표현 형식이 제공할 수 있는 질의 가능성의 준상한으로 해석해야 한다. 본 연구의 목적은 특정 알고리즘의 절대적 우열을 검증하는 것이 아니라, 동일한 원천 데이터가 서로 다른 표현 형식으로 제공되고 각 형식에 적합한 활용 도구와 결합될 때 질의 가능성이 어떻게 달라지는지를 비교하는 데 있다.

셋째, 본 실험은 특정 특정 거대 언어모델(LLM, Large Language Model), 임베딩 모델(Embedding model), 청크 분할 전략(Chunking strategy), 프롬프트 설정(Prompt configuration)에 기반하므로 절대 성능 수치는 실험 환경에 의존적일 수 있다. 따라서 본 연구의 핵심 해석은 특정 점수의 일반화가 아니라, 동일 조건에서 표현 형식과 활용 도구의 결합에 따라 질의 가능성이 달라졌다는 상대적 비교에 있다. 향후 연구에서는 다양한 거대 언어모델, 임베딩 모델, 재순위화(Reranker), 혼합 검색(Hybrid retrieval) 구성을 적용하여 모델 의존성을 추가 검증할 필요가 있다.

넷째, JSON 대조군의 해석에는 주의가 필요하다. 본 연구의 JSON 방식은 JSON 구조를 별도의 파서(Parser)나 스키마 인식 실행기(Schema-aware executor)로 처리한 것이 아니라, JSON 직렬화 텍스트를 벡터 검색 대상으로 사용한 조건이다. 따라서 JSON 방식의 낮은 성능은 JSON 표현 형식 자체의 한계라기보다, 구조 인식 도구 없이 JSON을 텍스트 검색 대상으로 처리했을 때의 한계로 해석해야 한다.

다섯째, 질문 변형 수준 1~3수준(L1~L3) 분석은 본 연구 평가셋에서 정의한 질문 표면형 변형에 대한 진단적 분석이다. 따라서 해당 결과는 일반적인 자연어 견고성 전체를 평가한 것이 아니라, 본 연구의 질의 유형과 변형 규칙 안에서 자연어 입력 방식과 구조 질의 조건 기반 방식의 성능 차이를 관찰한 결과로 해석해야 한다.

이러한 한계를 고려할 때, 본 연구의 결과는 “RDF 또는 XLSX가 일반적으로 RAG보다 우수하다”는 결론이 아니라, 표준 항목을 갖는 예산설명서 도메인에서 표현 형식과 활용 도구의 적합성이 질의 가능성에 중요한 영향을 미친다는 근거로 해석되어야 한다. 향후 연구에서는 텍스트-SPARQL 변환(Text-to-SPARQL), 텍스트-SQL 변환(Text-to-SQL), 판다스 도구(Pandas tool), 계산기(Calculator), 혼합 RAG(Hybrid RAG)등을 결합하여 자연어 질의에서 구조 질의로 이어지는 전체 처리(End-to-end) 시스템 성능을 추가적으로 검증할 필요가 있다.

Ⅵ. 결 론

본 연구는 공공 예산설명서와 같은 반정형 행정문서에서 데이터 표현 형식과 활용 도구의 결합이 인공지능 기반 질의응답 성능에 어떤 영향을 미치는지를 검토하였다. 기존 공공데이터 논의는 주로 데이터 개방, 기계 판독성, 표준 형식 제공에 초점을 두어 왔으나, 동일한 원천 자료를 여러 데이터 형식으로 변환했을 때 실제 질의 가능성이 어떻게 달라지는지에 대한 실증적 검토는 제한적이었다. 이에 본 연구는 공공데이터의 인공지능 활용성을 표현 형식과 활용 도구의 적합성 관점에서 분석하고자 하였다.

이를 위해 2026년 인공지능 관련 정부 재정사업 예산설명서 533건을 대상으로 문서형 텍스트, JSON 직렬화, XLSX 표, RDF 그래프의 네 가지 표현 형식을 구성하였다. 이후 문서형 벡터 검색, JSON 직렬화 벡터 검색, Pandas Agent, RDF 기반 SPARQL 질의, XLSX 질의의 다섯 가지 활용 방식을 설계하고, 100문항 평가셋을 통해 질의 유형별 성능을 비

교하였다. 평가 문항은 다단계 관계 질의, 역방향 탐색, 복합 조건 필터링, 집계 질의로 구성하였으며, 방식 간 성능 차이는 대응표본 부트스트랩 95% 신뢰구간을 통해 검토하였다.

실험 결과, 구조화 정보를 활용하는 방식들은 문서형 벡터 검색보다 높은 성능을 보였다. 특히 XLSX 질의와 RDF 기반 SPARQL 질의는 구조 질의 조건이 제공되었을 때 높은 수준의 질의 성능을 보였으며, Pandas Agent는 자연어 입력 방식 중 가장 높은 성능을 기록하였다. 반면 JSON 직렬화 벡터 검색은 문서형 벡터 검색보다 낮은 성능을 보여, 기계 판독 가능한 형식이라도 구조를 해석하고 활용하는 도구와 결합되지 않으면 실제 질의 가능성이 제한될 수 있음을 확인하였다.

또한 질의 유형별 분석 결과, 단일 방식이 모든 질의 유형에서 일관되게 우수하지는 않았다. XLSX 질의는 집계와 다단계 관계 질의에서 강점을 보였고, RDF 기반 SPARQL 질의는 관계 탐색형 질의에서 비교적 안정적인 성능을 보였으며, Pandas Agent는 복합 조건과 역방향 탐색에서 강점을 보였다. 이는 공공 행정문서의 인공지능 활용에서 특정 형식이나 도구의 절대적 우위보다, 질의 유형과 데이터 구조에 맞는 표현-도구 조합을 설계하는 것이 중요함을 시사한다.

본 연구의 의의는 검색증강생성, 시맨틱 웹, 자연어-구조화 질의 변환, 인공지능 친화적 공공데이터 논의를 하나의 실험 프레임 안에서 연결했다는 점에 있다. 특히 공공데이터의 인공지능 활용성은 단순히 데이터를 공개하거나 기계 판독 가능한 형식으로 변환하는 것만으로 확보되지 않으며, 데이터 표현 형식과 이를 해석·실행하는 활용 도구가 함께 설계되어야 함을 실증적으로 제시하였다. 이러한 결과는 향후 공공데이터 정책이 형식 표준화뿐 아니라 실제 질의 가능성과 활용 도구 연계를 함께 고려하는 방향으로 발전할 필요가 있음을 보여준다.

향후 연구에서는 본 연구의 평가 대상을 복지·고용·국토·과학기술 등 다양한 정책 분야의 예산설명서와 실제 사용자 질의 로그로 확장하고, 한국어 특화 및 공개형 대규모 언어모델을 함께 비교하여 모델 의존성을 추가 검증할 필요가 있다. 또한 JSONPath·스키마 인식 검색 등 JSON 구조를 직접 활용하는 방식과 부처·기관 약어 사전 보강·스키마 연결 절차 정교화를 통해 자연어 입력 기반 구조 질의 방식의 안정성을 개선할 수 있으며, 공공데이터 포털과 연계한 통합형 질의 환경 구축으로 본 연구의 결과를 실증적으로 확장할 수 있다.

References

[1] Presidential Council on National AI Strategy, "Artificial Intelligence Action Plan: National AI Strategy (2026-2028)", https://www.aikorea.go.kr. [accessed: May 11, 2026]

[2] H. Kim, "An Analysis of the Gap between Quantitative Expansion and Substantive Utilization of Public Open Data", Journal of the Korean Library and Information Science Society, Vol. 54, No. 2, pp. 5-26, Jun. 2023. https://doi.org/10.16981/kliss.54.2.202306.5.

[3] Ministry of the Interior and Safety and NIA, "Guidelines for AI-Friendly Management of Public Data V1.0", https://www.data.go.kr. [accessed: May 11, 2026]

[4] M. D. Wilkinson, M. Dumontier, I. J. J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J. W. Boiten, L. B. da Silva Santos, and P. E. Bourne, "The FAIR Guiding Principles for Scientific Data Management and Stewardship", Scientific Data, Vol. 3, No. 160018, pp. 1-9, Mar. 2016. https://doi.org/10.1038/sdata.2016.18.

[5] T. Berners-Lee, "Linked Data — Design Issues (5-Star Open Data Model)", https://www.w3.org/DesignIssues/LinkedData.html. [accessed: May 11, 2026]

[6] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. T. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks", Proc. 34th Conf. on Neural Inform. Process. Syst. (NeurIPS 2020),

Vancouver, Canada, pp. 9459-9474, Dec. 2020. https://doi.org/10.48550/arXiv.2005.11401.

[7] 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, Dec. 2023. https://doi.org/10.48550/arXiv.2312.10997.

[8] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, and J. Larson, "From Local to Global: A GraphRAG Approach to Query-Focused Summarization", arXiv preprint arXiv:2404.16130, Apr. 2024. https://doi.org/10.48550/arXiv.2404.16130.

[9] S. Yu, C. Tang, B. Xu, J. Cui, J. Ran, Y. Yan, Z. Liu, S. Wang, X. Han, Z. Liu, and M. Sun, "TableRAG: Million-Token Table Understanding with Language Models", Proc. 38th Conf. on Neural Inform. Process. Syst. (NeurIPS 2024), Vancouver, British Columbia, Canada, Vol. 37, pp. 74899-74921, Dec. 2024. https://doi.org/10.52202/079017-2382.

[10] World Wide Web Consortium (W3C), "RDF 1.1 Concepts and Abstract Syntax", https://www.w3.org/TR/rdf11-concepts/. [accessed: May 11, 2026]

[11] S. Harris and A. Seaborne (eds.), "SPARQL 1.1 Query Language", https://www.w3.org/TR/sparql11-query/. [accessed: May 11, 2026]

[12] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. de Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, A. C. N. Ngomo, A. Polleres, S. M. Rashid, A. Rula, L. Schmelzeisen, J. Sequeda, S. Staab, and A. Zimmermann, "Knowledge Graphs", ACM Computing Surveys, Vol. 54, No. 4, pp. 1-37, Jul. 2021. https://doi.org/10.1145/3447772.

[13] K. Höffner, J. Lehmann, and R. Usbeck, "OpenBudgets.eu: A Distributed Open-Source Budget Data Management Platform", Proc. 13th International Conf. on Semantic Systems (SEMANTiCS 2016), Leipzig, Germany, pp. 1-8, Sep. 2016. https://doi.org/10.1145/2993318.2993345.

[14] R. Cifuentes-Silva, J. M. Álvarez-Rodríguez, J. E. Labra-Gayo, and N. Lawrynowicz, "Semantic Knowledge Extraction from National Budget Data: an Open Data Driven Approach", Proc. 19th International Conf. on Web Engineering (ICWE 2019), Daejeon, South Korea, pp. 1-12, Jun. 2019. https://doi.org/10.1007/978-3-030-26023-1_15.

[15] T. Yu, R. Zhang, K. Yang, M. Yasunaga, D. Wang, Z. Li, J. Ma, I. Li, Q. Yao, S. Roman, Z. Zhang, and D. Radev, "Spider: A Large-Scale Human-Labeled Dataset for Complex and Cross-Domain Semantic Parsing and Text-to-SQL Task", Proc. 2018 Conf. on Empirical Methods in Natural Language Processing (EMNLP 2018), Brussels, Belgium, pp. 3911-3921, Nov. 2018. https://doi.org/10.18653/v1/D18-1425.

[16] G. H. Kim, S. H. Lim, J. Y. Park, and Y. G. Kim, "A Template-Vector Hybrid Approach to Domain-Specific Text-to-SQL for Korean Administrative Data", Journal of KIIT, Vol. 22, No. 8, pp. 145-156, Aug. 2024. https://doi.org/10.14801/jkiit.2024.22.8.145.

[17] Ministry of Economy and Finance, "Detailed Guidelines for the Formulation of the 2026 Budget and Fund Management Plan", Mar. 2025.

저자소개

임 수 혁 (Soohyuk Im)



2025년 11월 ~ 현재 : 국가인공지능전략위원회 AI데이터·규제혁신팀 전문관
2022년 2월 : 경북대학교 정보과학과(박사수료)
2016년 6월 ~ 현재 : 한국지능정보사회진흥원(NIA) 책임연구원
관심분야 : 국가 데이터 플랫폼, 데이터 거버넌스, 메타데이터 일관성 및 구조적 표준화, AI Readiness

서 영 균 (Young-Kyoon Suh)



2005년 2월 : 한국과학기술원
컴퓨터과학(석사)
2015년 5월 : 美 Univ. of Arizona
컴퓨터과학(박사)
2017년 9월 ~ 현재 : 경북대학교
컴퓨터학부 정보과학과 부교수
관심분야 : 데이터 사이언스, 엣지 컴퓨팅, 응용 AI