Check for updates

Journal of KIIT. Vol. 24, No. 10, pp. 277-287, Oct. 31, 2026. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2026.24.10.277

모호 표현 제어와 교차검증 지오코딩을 활용한 자연어 기반 공간 데이터 시각화

최 권 택*

Vague Expression Control and Cross-Validated Geocoding for Natural Language-based Spatial Data Visualization

Kwon-Taeg Choi*

요 약

자연어를 공간 필터 조건으로 변환하는 작업은 구조적으로 단순하나, 지리 시각화 맥락에서는 데이터 분포 정보 없이 LLM이 임계값을 선택할 때 속성·레이어별로 결과가 달라지는 모호 표현 수치화 문제와, 할루시네이션·좌표 불일치로 인한 지명 공간 범위 결정 문제가 발생한다. 본 연구는 이를 해결하기 위해 모호 표현 매핑 규칙으로 자연어 모호성을 백분위 통계값으로 수치화하는 컨텍스트 주입 방법과, LLM 추론값과 Google Maps API를 교차검증하여 지명-좌표 매핑을 동적으로 구성하는 지오코딩 방법을 제안한다. 18개 GeoServer 레이어, 미국·한국 지명 197개를 대상으로 한 실험에서 모호 표현이 매핑 규칙에 따라 백분위수로 일관되게 수렴하였으며, 지명·거리·공원·구역·도시 유형에서 중심점 오차 중앙값 0.3 km를 확인하였고, 레이어 자동 분류에서 97.22%의 정확도를 달성하였다.

Abstract

Converting natural language into spatial filter conditions is structurally straightforward, yet in the geospatial visualization context two challenges arise: inconsistent threshold selection across attributes and layers when LLMs select thresholds for vague expressions without data distribution information, and place name coordinate-range determination errors caused by hallucination and coordinate discrepancies. To address these, this study proposes a context injection method that numerically defines natural language ambiguity as percentile statistics via vague expression mapping rules, and a geocoding method that dynamically constructs place name-to-coordinate mappings by cross-validating LLM-inferred coordinates against the Google Maps API. Experiments across 18 GeoServer layers and 197 US and Korean place names show that vague expressions consistently converge to percentile values according to the mapping rules, a median centroid error of 0.3 km for place name, street, park, district, and city types, and a layer classification accuracy of 97.22%.

Keywords

natural language processing, large language model, ECQL, geocoding

* 강남대학교 ICT융합공학부 교수
- ORCID: http://orcid.org/0000-0001-5331-321X

· Received: Jul. 06, 2026, Revised: Jul. 28, 2026, Accepted: Jul. 31, 2026
· Corresponding Author: Kwon-Taeg Choi
Dept. of ICT Convergence Engineering, Kangnam University, Yongin, Korea
Tel.: +82-31-280-3660, Email: kwontaeg.choi@kangnam.ac.kr

Ⅰ. 서 론

GIS(Geographic Information System)는 도시계획·환경 모니터링·교통 관리·재난 대응 등 다양한 분야에서 활용되고 있다. 그러나 일반 사용자가 접근할 수 있는 시각화는 시스템에 사전 구성된 결과로 한정된다. 원하는 조건으로 공간 데이터를 필터링하거나 스타일을 바꿔 새로운 시각화를 얻으려면 공간 쿼리 문법·레이어 구조·좌표계 지식 등 GIS 전문 지식이 필요하므로, GIS 비전문가는 자신이 실제로 필요로 하는 공간 데이터 시각화 결과를 직접 얻을 수 없다는 기술적 진입 장벽이 존재한다.

최근 LLM(Large Language Model)의 발전으로 이 장벽을 낮추는 자연어 기반 GIS 인터페이스 연구가 활성화되어, LLM을 핵심 추론 엔진으로 활용하는 Autonomous GIS[1]와 다중 에이전트 프레임워크[2]가 제안되었다. 이러한 연구는 방법론적으로 NL2SQL(Natural Language to SQL) 분야의 SQL(Structured Query Language) 생성 능력 검증 성과[3]-[5]에 근간을 두고 있다. 자연어 공간 질의 변환 연구는 이를 바탕으로 GIS 도메인의 고유한 특성을 반영하여 확장되고 있다. 지리 분석 질의응답 관련 연구에 대한 구조가 체계화되었으며[6], GeoGPT[7]는 에이전트 기반 GIS 도구 호출을, ChatGeoAI[8]는 온톨로지 기반 컨텍스트 주입을 통한 공간 필터 생성을 각각 제안하였다. 또한 "near"·"far" 같은 모호한 공간 표현의 퍼지 의미론이 형식화되었고[9], LLM의 공간 인지 능력에 규모·언어별 성능 편차가 확인되었다[10]. 자연어 공간 질의를 ECQL 필터로 변환하는 작업은 일반 NL2SQL보다 구조가 단순하여 LLM으로 충분히 처리 가능하나, 지리 시각화 맥락에서 두 가지 고유한 어려움이 있다.

첫 번째 어려움은 자연어 모호 표현의 수치화 문제이다. 공간 데이터 시각화 질의는 지도상의 상대적 분포·패턴을 파악하려는 목적을 가지므로, 정확한 값을 지정하는 일반 데이터 질의보다 모호 표현이 더 빈번하게 나타난다. "인구가 많은 주"처럼 상대적 비교만을 표현하는 질의를 수치 필터 조건으로 변환하려면 임계값 결정이 필요하다. 데이터 분포 정보 없이 LLM이 임계값을 선택하면 동일 표현에 대해서도 속성·레이어에 따라 결과가 일관되지 않으며[9], 실제 통계를 컨텍스트로 제공해야 일관성을 확보할 수 있다.

두 번째 어려움은 지명의 정확한 공간 범위 결정 문제이다. "뉴욕 주 경계를 표시해줘" 같은 질의에서 지명을 위경도 좌표로 변환해야 하나, LLM의 지명-좌표 변환 능력은 지역 규모와 언어에 따라 크게 달라진다[10]. 이를 해결하기 위해 지명을 사전에 모두 등록하는 것은 현실적으로 불가능하므로 실용적인 지오코딩 방법이 필요하다.

본 연구는 이 두 문제를 해결하기 위해 LLM 기반 도구 에이전트 시스템을 제안한다. 첫째, 모호 표현 매핑 규칙으로 자연어 표현을 통계 백분위 통계값으로 수치화하는 컨텍스트 프레임워크를 구성한다. 둘째, LLM 추론값과 Google Maps API를 교차검증하여 지명-좌표 매핑을 동적으로 구성하는 지오코딩 방법을 결합한다. 대상 플랫폼은 GeoServer이며, 속성 조건 필터링과 공간 연산자를 지원하는 ECQL(Extended Contextual Query Languag) 필터 생성을 목표로 한다.

본 논문의 구성은 다음과 같다. 2장은 NL2SQL, LLM-GIS 관련 선행 연구를 검토한다. 3장은 시스템 구조, 도구 기반 에이전트, 컨텍스트 구성 방법, 교차검증 지오코딩을 기술한다. 4장은 세 가지 실험 결과를 제시하고, 5장은 결론 및 향후 연구를 논의한다.

Ⅱ. 관련 연구

NL2SQL 연구는 대규모 벤치마크 구축과 이를 기반으로 한 변환 방법론의 고도화를 중심으로 발전하였다. Spider[3]는 200개 데이터베이스에 걸친 크로스도메인 벤치마크로 JOIN·서브쿼리·집계 등 복잡한 SQL 구문을 평가하며, WikiSQL[4]은 단일 테이블 구조에 특화된 80,654개 쌍으로 기초 조건 변환 능력을 측정한다. DIN-SQL[5]은 스키마 링킹·분류·SQL 생성·자기교정의 네 단계로 변환을 분해하고 in-context learning을 결합하여 Spider 기준 85.3%의 실행 정확도를 달성하였다. 이들 연구는 LLM의 SQL 생성 능력을 입증하며, 이들 연구는 LLM의 SQL 생성 능력을 입증하였으며, ECQL 필터 조건은 SQL의 WHERE 절에 대응하는 상대적으로 단순한 변환 대상이므로 이미 충분한 기술적 성

능 기반을 갖추고 있다[11].

GIS 도메인에 특화된 자연어 질의 처리 연구도 다양한 방향에서 진행되고 있다. 우선 장소·시간·공간 관계 등 핵심 개념을 중심으로 지리 분석 질의 응답의 태스크 구조가 체계화되었다[6]. 질의-쿼리 변환 측면에서는 장소 관련 질문을 GeoSPARQL 쿼리로 변환할 때 중간 논리 표현 단계를 거치면 변환 품질이 향상됨이 실험적으로 확인되었다[12]. 모호 표현 해석 측면에서는 "near"·"large" 등 공간 관계 표현의 퍼지 의미론이 형식화되어, 이러한 표현이 이진 조건이 아닌 점진적 멤버십 함수로 표현되어야 함이 제시되었다[9].

LLM과 GIS 도구의 통합 연구도 활발하다. GeoGPT[7]는 에이전트가 자연어 입력을 해석하여 공간 분석 도구를 호출하고 시각화를 자동 수행하는 프레임워크를 제안하였다. ChatGeoAI[8]는 GIS 도메인 온톨로지를 LLM 컨텍스트에 동적으로 주입하여 공간 필터 표현을 생성하는 연구를 수행했다. GPT-4·DeepSeek-R1을 GIS 워크플로에 통합해 자연어 지시를 Python으로 변환하는 방식은 LLM 기반 GIS 자동화 가능성을 보여주었다[13].

RAG(Retrieval-Augmented Generation)[14]는 외부 지식을 동적으로 검색하여 LLM 컨텍스트에 주입함으로써 지식 집약적 태스크의 정확도를 높이는 원리로, 다양한 도메인의 컨텍스트 강화에 응용되고 있다. 본 연구의 컨텍스트 프레임워크는 이 원리를 ECQL 생성에 특화하여, 모호 표현 매핑 규칙으로 자연어를 백분위 통계값으로 수치화하고 레이어 스키마·백분위 통계·매핑 규칙을 컨텍스트로 제공한다.

기존 연구들은 일반적 SQL 변환이나 GIS 작업 자동화에 초점을 맞추며, 모호 표현의 수치화와 지명의 공간 범위 결정 문제를 직접 다루지 않는다. ChatGeoAI는 온톨로지 기반 컨텍스트를 활용해 공간 필터 표현을 생성하지만, 임계값 결정에 실제 데이터 분포 통계를 연동하지 않아 모호 표현 해석의 일관성 문제를 해소하지 못하며, LLM이 추론한 지명 좌표의 타당성을 검증하는 절차도 포함하지 않는다. 본 연구는 이 공백을 채우기 위해, 모호 표현의 수치 정의를 위한 컨텍스트 구성과 교차검증 지오코딩을 통합하는 ECQL 변환 시스템을 제안한다.

III. 제안 방법

3.1 전체 시스템 개요

GIS 전문 지식이 없는 사용자가 "인구가 많은 미국 주를 보여줘", "뉴욕 주변 관심 지점을 표시해줘"와 같이 자연어로 질의를 입력하면, 제안 시스템은 해당 의도를 해석하여 적절한 레이어와 공간 필터 조건을 결정하고, 웹 지도로 시각화한다. 이 과정에서 사용자는 레이어 구조, WMS(Web Map Service) URL 구성, ECQL 필터 문법, 좌표계 등 GIS 지식이 필요하지 않다.

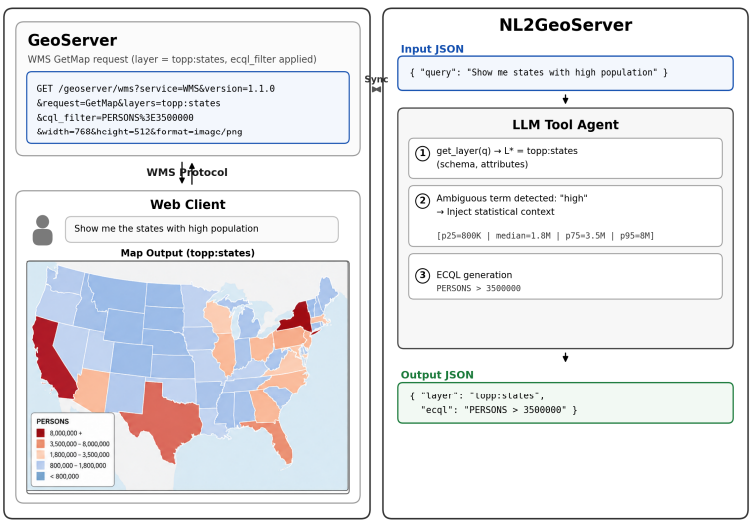


그림 1. 제안 시스템의 전체 파이프라인
Fig. 1. Overall pipeline of the proposed method

그림 1은 웹 기반 지도 시각화, GeoServer, NL2GeoServer로 구성된 전체 시스템 구조를 나타낸다. GeoServer는 OGC(Open Geospatial Consortium) 표준을 준수하는 오픈소스 공간 데이터 서버로, WMS 프로토콜을 통해 레이어 데이터를 웹 지도 이미지로 제공한다. WMS 요청에는 LAYERS, CQL_FILTER 파라미터가 포함된다.

사용자가 질의를 입력하면 NL2GeoServer로 전달되며, 레이어 결정 단계에서 get_layer 도구를 통해 대상 레이어를 식별하고, 해당 레이어의 스키마 정보를 기반으로 컨텍스트를 구성한다. 이때 질의에 모호한 수치 표현이 포함된 경우 해당 속성의 백분위 통계 매핑 정보를 컨텍스트에 추가 주입한다. 지명이 포함된 질의에서는 위치 결정 단계에서 get_geocode 도구를 통해 해당 지명의 위·경도 좌표를 결정한다.

이후 LLM은 구성된 컨텍스트와 위치 정보를 바탕으로 ECQL 조건문을 생성하고, 구조화된 응답을 반환한다. 시스템은 이 응답에서 레이어명과 조건문을 추출하여 WMS URL을 구성하며, 프론트엔드가 해당 URL을 GeoServer에 요청하면 조건을 만족하는 공간 객체만 렌더링된 지도 이미지로 갱신된다.

3.2 도구 기반 Geo 에이전트

제안 시스템은 ReAct(Reasoning and Acting)[15] 패러다임을 기반으로, 두 개의 전용 도구(get_layer, get_geocode)와 동적 컨텍스트 조합 메커니즘을 결합한 LLM 에이전트로 구현된다. Schick et al.[16]의 Toolformer 원리에 따라 LLM은 도구 호출 시점·파라미터를 자율적으로 판단하고 각 도구 출력을 추론에 반영하여 L*와 ECQL을 생성한다. 이 변환 과정을 형식화하면 식 (1)과 같다.

$$(L^{*},\ ECQL) = F(q,\ prompt,\ context_{schema},\ T) \tag{1}$$

여기서 q는 자연어 질의, prompt는 LLM 입력 프롬프트, $context_{schema}$는 레이어명·지오메트리 유형·속성명 등을 포함한 스키마 정보, T = {get_layer, get_geocode}는 도구 집합이며, L*는 결정된 대상 레이어, ECQL은 생성된 필터 조건문이다.

대상 레이어는 식 (2)로 결정된다.

$$L^{*} = get_layer(q,\ context_{schema}) \tag{2}$$

get_layer는 q와 $context_{schema}$를 입력으로 받아 가장 부합하는 레이어명 L*를 출력한다.

자연어를 ECQL 로 변환하는 프롬프트는 정확한 ECQL 생성에 필요한 구성 요소를 명시적으로 조합하는 방식으로 설계되었다. 구체적으로 LLM에 역할을 부여하는 지시(P_{inst}), 호출 가능한 도구 명세(T), 레이어와 질의에 따라 동적으로 구성되는 컨텍스트(Context), 출력 형식을 규정하는 규칙(P_{output})의 네 요소를 순서대로 연결하며, 식 (3)은 이 구성 구조를 형식화한 것이다.

$$prompt = P_{inst} \oplus T \oplus context \oplus P_{output} \tag{3}$$

여기서 ⊕는 프롬프트 구성 요소의 연결(Concatenation) 연산이다. P_{inst}는 GIS 전문가로서 자연어 질의를 ECQL 필터로 변환하는 역할을 LLM에 부여하는 지시를, T는 식 (1)의 도구 집합 명세를, P_{output}은 레이어명과 ECQL을 JSON 형식으로 반환하도록 유도하는 출력 형식 규칙을 담는다. context는 질의와 대상 레이어에 따라 동적으로 구성되는 서브 컨텍스트의 연결이다. 각 구성 요소의 내용과 순서는 연구 목적에 따라 결정되었으며, 별도의 프롬프트 최적화는 수행하지 않았다.

context는 레이어 스키마 정보($context_{schema}$), 분위 통계($context_{stat}$), 모호 표현 매핑 규칙($context_{guide}$)와 1개의 선택적 지명 좌표 정보($context_{geo}$), 이렇게 네 요소로 구성되며, 식 (4)는 이 조합 구조를 나타낸다.

$$context = context_{schema}(L^{*}) \oplus context_{stat}(L^{*}) \oplus context_{guide}\ \ [\oplus context_{geo}] \tag{4}$$

식 (4)에서 []는 선택적 포함을 나타내며, 그림 2는 context의 각 구성 요소의 실제 프롬프트 텍스트 예시를 나타낸다.

```
## Layer Schema (topp:states)
- PERSONS (Integer): population of the state
- STATE_NAME (String): name of the state
- LAND_KM (Double): land area (km²)                    context_schema(L*)
```

```
## Quantile Statistics
{
  "PERSONS": {"p5": 610550, "p25": 1123000, "median": 3665228,
              "p75": 6629000, "p95": 25008017},
  "LAND_KM": {"p5": 12345, "p25": 45678, "median": 98765,
              "p75": 156789, "p95": 432100}
}                                                     context_stat(L*)
```

```
## Vague Expression Mapping Rules
- "most" / "highest" / "largest" → >= p95
- "many" / "high" / "large"      → >= p75
- "few" / "low" / "small"        → <= p25
- "fewest" / "lowest" / "smallest" → <= p5            context_guide
```

```
## Location  ← included only when query contains a place name
- California: POINT(-119.42, 36.78)
  BBOX: (-124.41, 32.53, -114.13, 42.01)              context_geo
```

그림 2. 컨텍스트 구성 요소별 프롬프트 예시
Fig. 2. Example prompt text for each context component

$context_{schema}(L^*)$는 대상 레이어 L의 속성명·타입·설명이며(그림 2 layer schema), $context_{stat}(L^*)$는 각 수치 속성의 백분위 통계(p5·p25·median·p75·p95)를 JSON 형식으로 제공한다(그림 2 quantile statistics). LLM은 이를 참조하여 "인구가 많은"과 같은 모호 표현의 임계값을 실제 데이터 분포에 근거해 결정한다. $context_{guide}$는 레이어·속성에 무관하게 일정하게 적용되는 모호 표현 매핑 규칙으로(그림 2 vague expression mapping rules), 이에 대해서는 3.3절에서 자세히 설명한다.

$context_{geo}$는 질의에 지명이 포함된 경우에만 추가되며, get_geocode가 반환하는 지명의 위경도 좌표 담는다(그림 2 location). 이에 대해서는 3.4절에서 기술한다.

3.3 통계 기반 모호 표현 처리

Lassiter & Goodman[17]은 "키가 크다", "인구가 많다"와 같은 긍정형 형용사의 해석 임계값이 참조 집단의 통계적 분포에 의존함을 RSA(Rational Speech Acts) 프레임워크로 이론화하였다. 동일한 "많다"라는 표현도 소규모 집단과 대도시 인구 맥락에서 전혀 다른 임계값을 가지며, 데이터 분포 정보 없이 LLM이 임계값을 선택하면 동일 표현에 대해 속성·레이어별로 일관성 없는 결과가 발생한다.

이를 해결하기 위해 본 연구는 대상 레이어의 수치 속성에 대한 통계 정보를 LLM 컨텍스트에 사전 주입하는 방법을 제안한다. 구체적으로, 각 수치 속성의 분포를 요약한 백분위 통계를 산출하여 LLM 입력 컨텍스트에 포함함으로써, LLM이 실제 데이터 분포를 참조하여 임계값을 선택하도록 유도한다.

임계값 선택을 설계자가 명시적으로 제어하기 위해, 본 연구에서는 모호 표현과 통계 지표 간 대응 규칙을 $context_{guide}$로 정의하여 프롬프트에 포함한다. 이 정보는 표현 유형별로 대응 백분위를 지정한 규칙표로, LLM이 이를 참조함으로써 임계값 선택이 설계자가 의도한 통계적 해석으로 수렴한다. 그리고 특정 데이터셋의 분포와 무관한 일반 규칙으로, 데이터셋 맥락과 응용 목적에 따라 설계자가 유형을 세분화하거나 통합하여 조정할 수 있다. 표 1은 본 연구에서 적용한 $context_{guide}$ 매핑 규칙으로, 5개 유형과 방향을 조합한 9개 항목으로 구성하였다. 예를 들어 topp:states 레이어에 "인구가 매우 많은 주를 보여줘"가 입력되면 degree-high 규칙에 따라 PERSONS 속성의 p90에 해당하는 값이 임계값으로 주입되어 PERSONS > 18000000의 ECQL 조건이 생성된다.

표 1. 모호 표현 매핑 규칙
Table 1. Vague expression mapping rules

Direction	Type	Vague expression	Percentile
High	Simple	many	p75
High	Degree	very many	p90
High	Compare	more than avg	median
High	Super	the most	p95
-	Range	middle	p25~p75
Low	Simple	few	p25
Low	Degree	very few	p5
Low	Compare	less than avg	median
Low	Super	the least	p5

3.4 공간 범위 결정

get_geocode 도구는 질의에 포함된 지명을 위경도 좌표로 변환한다. LLM을 활용하면 다양한 AI 모델별 좌표 불일치와 부정확한 위경도를 생성하는 할

루시네이션이 발생할 수 있다[10][18]. 반면 Google Maps Geocoding API는 높은 좌표 정밀도를 제공하지만, 질의의 맥락적 의미를 해석하지 않고 자체 인덱스에서 최상위로 매칭되는 단일 지점을 반환하므로 사용자가 의도한 지명과 다른 결과가 반환될 수 있다.

본 연구에서는 이를 해결하기 위해 LLM과 Google Maps API를 교차검증하여 두 소스의 한계를 상호 보완하되, 오차가 큰 유형에 대해서만 사전 등록을 요구하는 방법을 제안한다. 교차검증 결과는 지명-좌표 매핑을 관리하는 $context_{geomap}$에 누적된다. 여기에 등록된 지명은 외부 호출 없이 즉시 신뢰할 수 있는 위치 정보를 반환하므로 정확도와 처리 효율이 동시에 향상된다. LLM이 자동으로 호출하는 get_geocode 함수의 전체 절차는 알고리즘 1과 같다.

알고리즘 1. LLM - Google maps API 교차검증 지오코딩 흐름도
Algoithm 1. Flowchart of LLM - Google maps API cross-validated geocoding

```
Algorithm 1. get_geocode(location, type)

1:  if location ∈ context_geomap then
2:      return context_geomap[location]
3:  end if
4:  pos_LLM ← LLM.geocode(location)
5:  pos_API ← GoogleMapsAPI.geocode(location)
6:  dH ← Haversine(pos_LLM, pos_API)
7:  if dH ≤ ε then
8:      context_geomap[location] ← pos_API
9:      return pos_API
10: else if type ∉ PRE_REG_TYPES then
11:     context_geomap[location] ← pos_API
12:     return pos_API
13: else
14:     return Error
15: end if
```

알고리즘 1에서 함수의 입력파라미터는 LLM에 의해서 자동으로 호출되고, 이중 type은 지명의 공간 규모에 따라 poi·street·park·district·city·state·region의 7개로 분류된다. poi·street는 소규모(단일 건물·도로), park·district·city는 중규모(국지적 생활공간~지역 행정 단위), state·region은 광역 경계 수준으로 공간 규모가 가장 크다. 유형별로 LLM과 Google API가 일관된 결과를 반환하는 정도가 다르며, 이 중 두 소스 간 오차가 크게 나타나는 유형의 집합을 PRE_REG_TYPES로 정의한다.

사전 등록된 지명은 $context_{geomap}$에서 즉시 반환하며(1~3행), 미등록 지명에 대해서는 pos_LLM과 pos_Google을 획득하여 Haversine 거리 dH를 산출한다(4~6행). dH는 구면 위의 두 점 사이의 대원 거리를 계산하며 식 (5)와 같다.

$$d_H = 2r\arcsin\left(\sqrt{\sin^2\left(\frac{\Delta\phi}{2}\right)+\cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\Delta\lambda}{2}\right)}\right) \tag{5}$$

여기서 ϕ는 위도, λ는 경도, r은 지구 평균 반경(6,371 km)이며, r의 단위가 km이므로 dH는 km 단위로 산출된다.

dH ≤ ε이면 일치로 판단하여 pos_Google을 $context_{geomap}$에 누적하고(7~9행), dH > ε이면 Google 결과를 우선 채택하되 type ∈ PRE_REG_TYPES인 경우 경계 오차가 ECQL 필터 정밀도에 영향을 미치므로 Error를 반환한다(10~15행). PRE_REG_TYPES는 4장 실험 2에서 측정한 유형별 오차 분포를 근거로 결정된다. ε는 두 소스 간 오차의 허용 범위를 정하는 임계값으로, 크게 설정할수록 오차가 있어도 자동 누적되어 처리 효율은 높아지지만 정밀도는 낮아지며, 작게 설정할수록 정밀도는 높아지나 캐시 적중률이 낮아져 매번 LLM과 Google API를 함께 호출해야 하는 트레이드오프를 가진다.

Ⅳ. 실 험

4.1 실험 데이터셋

본 연구는 세 가지 실험을 통해 제안 방법을 검증한다. 실험 1은 컨텍스트 수준에 따른 모호 표현 임계값 수렴 여부를, 실험 2는 교차검증 기반 지오코딩의 유형별 정확도를, 실험 3은 레이어 자동 분류 성능을 각각 측정한다. 각 실험에 사용된 데이터셋의 구성과 특성을 표 2에 정리한다.

표 2. 실험 데이터셋 구성
Table 2. Experimental dataset composition

Dataset	N	Description
GeoServer layer	18	GeoServer-provided layers from four workspaces (tiger, topp, sf, ne), including Point, Polygon, and Line geometries, with 1–23 attributes per layer.
Vague expression query	224	Natural language queries generated by combining directional and query types (GPT-5).
U.S. place name	100	Collected place names for seven spatial entity types: poi, street, park, district, city, state, and region.
Korean place name	97	Collected place names for the same seven spatial entity types, reflecting the Korean administrative hierarchy and jurisdictional structure
Layer categories	288	Natural language expressions generated by combining 18 layers with Strong (attribute names) and Medium (conceptual terms) across eight combinations (GPT-5).

GeoServer가 기본적으로 제공하는 레이어는 실험 1과 실험 3의 공통 기반으로, 실험에서 사용한 18개 레이어는 tiger,topp,sf,ne 네 워크스페이스에 분포한다. tiger 워크스페이스의 3개 레이어는 뉴욕시 도시 인프라(관심지점·랜드마크·도로망)를, topp의 5개 레이어는 미국·태즈메이니아 행정·자연 경계를, sf의 5개 레이어는 Spearfish 고고·생태 지점을, ne의 5개 레이어는 전 세계 국가 경계·해안선·주요 도시를 대상으로 한다. 지오메트리 유형은 점, 다각형, 선을 포함하며 레이어 별 속성 수는 1개에서 23개까지 다양하다. 속성 유형도 수치형·문자형·범주형으로 다양하고, 주제 영역도 도시 인프라·행정 경계·자연 지형·전 세계 지리에 걸쳐 다양한 레이어 구조와 질의 상황을 포괄하는 평가 환경을 구성한다. 수치 속성이 포함된 4개 레이어(topp:states·tasmania_cities·tasmania_roads·tasmania_water_bodies)에 대해서는 속성별 백분위 통계를 산출하여 실험 1의 컨텍스트 구성에 활용하였다.

모호 표현 데이터셋은 수치 조건을 명시하지 않고 상대적 크기나 정도만을 나타내는 자연어 질의로, simple("많은·적은"), degree("매우 많은·적은"), compare("평균보다 많은·적은"), super("가장 많은·적은"), range("중간 정도")의 5개 유형으로 구성된다. 질의 템플릿은 속성 표현, 모호 수식어, 레이어를 조합해 자연스러운 문장이 만들어 지도록 LLM(GPT-5-mini)으로 생성했다.

지명 데이터셋은 poi,street,park,district,city,state,region의 7개 유형으로 공간 규모가 미터 수준(poi)부터 수천 km(region)까지 지리 시각화 질의의 전체 스펙트럼을 포괄한다.

레이어 분류 평가셋은 18개 레이어에 대해 Strong(고유 속성명 포함)·Medium(지역명·주제 개념 단서만) 두 난이도로 LLM(GPT-5-mini)으로 자동 생성한 한국어 질의이다. 본 연구에서 구성한 모호 표현 질의, 지명, 레이어 분류 데이터셋은 Hugging Face Hub에서 다운로드할 수 있다[19].

평가 실험에 사용한 LLM은 상용 모델 2종(GPT-4o-mini, GPT-5-mini)과 공개 모델 3종(openai/gpt-oss-20b, openai/gpt-oss-120b, llama-3.3-70b-versatile)으로 총 5종이다.

4.2 실험

첫 번째 실험은 LLM이 모호 표현을 통계 기반으로 일관되게 해석할 수 있는지를 검증한다. 데이터 분포 정보 없이 LLM이 "인구가 많은"과 같은 표현을 해석하면 속성·레이어에 따라 임계값이 달라질 수 있으며, 백분위 통계($context_{stat}$)와 매핑 규칙($context_{guide}$)을 제공했을 때 임계값이 설계 의도대로 수렴하는지를 확인한다.

실험 대상은 4.1절의 모호 표현 데이터셋 224개 질의이다(states 168개, cities 21개, roads 14개, water_bodies 21개). 질의 예시로는 "인구가 많은 주를 보여줘"(Simple-high), "면적이 매우 좁은 주는?"(Degree-low), "평균보다 인구가 많은 주"(Compare-high), "가장 면적이 넓은 주"(Super-high), "면적이 중간 정도인 주를 보여줘"(Range)가 있다. 분석 지표는 threshold_ratio로, LLM이 생성한 임계값을 해당 속성의 최솟값과 최댓값 사이로 정규화하여 [0, 1] 구간으로 변환한 값이다. 속성마다 값 범위가 상이하므로 정규화를 통해 서로 다른 속성

간 임계값 위치를 동일한 척도에서 비교할 수 있다. range 유형은 구간의 하한·상한을 각각 산출한다.

먼저 레이어 스키마 정보만 제공한 조건(Schema)과 백분위 통계를 추가한 조건(Stat)을 비교하여 통계 정보가 임계값 선택의 일관성에 미치는 영향을 분석한다. 기준 모델은 GPT-4o-mini이다.

그림 3은 many·few 2개 유형의 threshold_ratio 분포이다. X축은 threshold_ratio, Y축은 4개 레이어의 4개 수치 속성이며, 파란 점(Schema)과 주황 점(stat)으로 각 응답을 나타낸다. 동일 속성에 대해 점이 넓게 산포될수록 LLM의 해석이 일관되지 않음을 의미하고, 한두 개의 점으로 밀집될수록 일관되게 수렴함을 의미한다. 수렴 위치를 통계적으로 확인하기 위해 p25·median·p75·max 참조선을 함께 표시하였다. schema 조건에서는 응답 점들이 [0, 1] 전반에 넓게 산포되어 동일한 모호 표현에 대한 임계값이 속성별로 불일치하는 반면, stat 조건에서는 "많은(Many)" 유형이 p75, "매우 적은(Very few)" 방향 유형이 p25 참조선 근방으로 수렴한다. 이는 백분위 통계가 제공될 때 LLM이 데이터 분포를 참조하여 속성의 종류에 관계없이 일관된 기준으로 모호 표현을 해석함을 나타낸다.

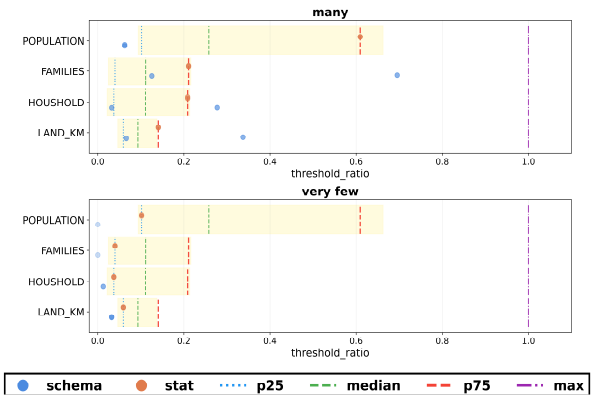


그림 3. threshold_ratio 분포 비교: schema vs stat (2개 모호 유형)

Fig. 3. Threshold_ratio distribution comparison: schema vs stat (2 ambiguity types)

그림 4는 range 유형의 BETWEEN 구간 추정을 비교한다. 각 속성에 대해 LLM이 추정한 구간을 하한(바 왼쪽 끝)과 상한(바 오른쪽 끝)을 잇는 수평 바로 표현하였다.

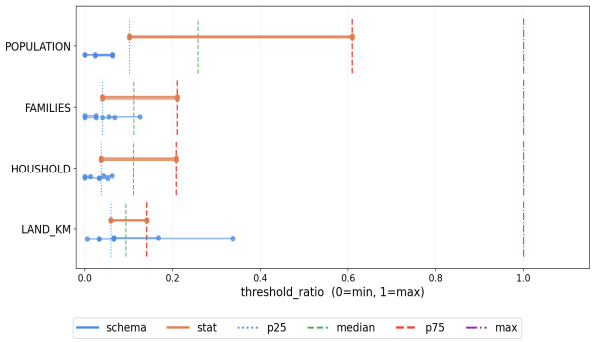


그림 4. BETWEEN 구간 비교: schema vs stat

Fig. 4. BETWEEN interval comparison: schema vs stat

schema 조건에서는 하한·상한 모두 속성마다 위치가 달라 일관성이 없고, 구간의 폭 자체도 속성에 따라 크거나 좁게 나타난다. 예를 들어 FAMILIES 속성의 경우 schema 컨텍스트에서 얻어진 구간이 [-0.0028, 0.0257]과 [0.0542, 0.1254]로 서로 겹치지 않을 만큼 상이하게 추정되었으며, 이는 동일한 속성·동일한 모호 표현 유형에 대해서도 모델이 구간의 위치(하한)뿐 아니라 폭(상한-하한)까지 안정적으로 추정하지 못함을 보여준다.

stat 조건에서는 대부분의 속성에서 하나의 수평 바로 수렴하며, 하한은 p25, 상한은 p75 참조선 근방에 위치한다. 다만 POPULATION과 FAMILIES 속성에서는 2개의 수평 바가 나타나는데, 이는 동일 질의에 대해 LLM이 상황에 따라 상한을 p75 또는 그보다 높은 값으로 다르게 해석함을 의미한다. 백분위 통계로 하한의 수렴은 안정적으로 제어되나, 상한 해석에는 속성의 분포 특성에 따른 부분적 변동이 남아 있음을 시사한다.

그림 5는 식 (4)의 제안 방법, 즉 $context_{guide}$(매핑 규칙)를 적용한 결과이다. 각 속성의 LLM 응답 점과 해당 규칙의 목표 백분위를 나타내는 참조선을 함께 표시했다.

그림 3에서 응답 점들이 [0, 1] 전반에 산포되었던 것과 달리, 그림 5에서는 속성의 종류에 관계없이 응답 점들이 참조선 근방에 밀집된다. 구체적으로 "많은(Many)"은 p75, "매우 적은(Very few)"은 p10 참조선 근방으로 수렴하며, 이는 모든 속성에서 일관되게 나타난다. 모호성이 통제되지 않은 그림 3의 결과와 비교할 때, $context_{guide}$의 추가가 LLM의 모호 표현 해석을 통계적으로 일관된 의미 체계로 제어할 수 있음을 보여준다.

두 번째 실험은 LLM마다 지오코딩 정밀도가 상이할 수 있으므로, 미국 지명 100개, 한국 지명 97개에 대해 5종 LLM의 지오코딩 결과를 Google Maps Geocoding API 응답과 유형별로 비교·분석한다. 평가 지표는 중심점 위치 오차를 나타내는 Haversine 거리로 측정했다.

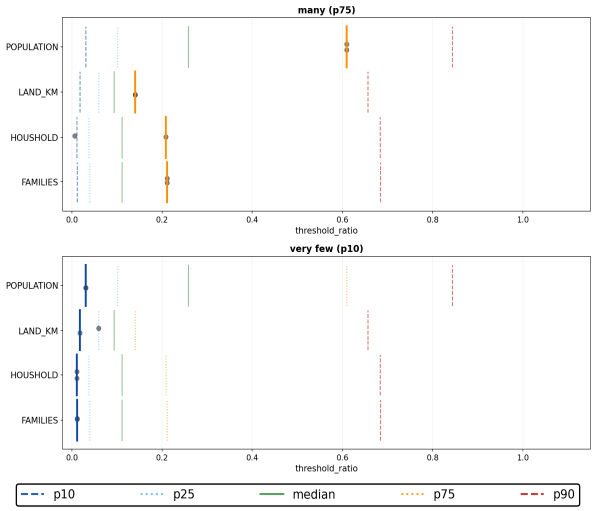


그림 5. $context_{guide}$ 매핑 규칙 준수율

Fig. 5. $context_{guide}$ mapping rule compliance distribution

표 3은 유형별 중심점 오차 중앙값(km)이다. poi와 city는 전 모델에서 0.1km 수준으로 매우 정확하다. poi는 특정 건물·기념물처럼 고유한 위치 정보가 학습 데이터에 풍부하게 포함되어 있고, city는 위키피디아·공개 지리 데이터 등 다양한 출처에서 일관된 좌표로 학습되기 때문으로 보인다. street·park·district는 모델 간 편차가 있다. street는 동일 이름의 도로가 여러 도시에 존재하거나 대표 좌표 기준이 모호하고, park는 광대한 면적을 갖는 경우 중심점 정의 자체가 불명확하며, district는 neighborhood 경계가 출처에 따라 달라 LLM마다 다르게 학습될 수 있다. 반면 state는 수십 km로 오차가 크다. 광역 행정 단위의 기하학적 중심과 행정 중심지가 수십 km 이상 떨어져 있어 LLM과 Google API가 서로 다른 기준점을 사용하기 때문으로 보인다. region은 LLM과 Google API 오차가 극단적이다. 따라서 state,와 region의 오차는 교차검증으로도 해소되지 않으며, 이에 따라 Algorithm 1에서 PRE_REG_TYPES = {state, region}으로 설정하였다.

표 3. 중심점 오차 중앙값(km) (US 데이터, state·region 제외 유형별 평균)

Table 3. Median centroid error (km) (US data, mean by Type, state·region excluded)

Type	GPT-4o -mini	GPT-5 -mini	OSS -20B	OSS -120B	Llama -70B
Poi	0.02	0.00	0.01	0.00	0.03
Street	0.57	0.70	2.02	0.37	0.72
Park	0.28	0.29	4.43	2.25	2.33
District	0.61	0.43	1.68	1.30	1.14
City	0.11	0.11	0.11	0.09	0.15
State	19.1	36.5	72.9	43.8	49.5
Region	979	1331	1084	960	1086
ALL	0.31	0.30	1.65	0.80	0.87

한국 지명셋에 대한 실험은(GTP-5-mini) 전체 오차 1.66km 로 미국 지명셋의 결과 0.30 km보다 높았다, 이는 LLM 학습 데이터의 언어 편중에 기인한다. region 유형은 예외적으로 한국(38.7 km)이 미국(1,331 km)보다 오차가 작은데, 이는 한국과 미국의 대륙 크기 차이에서 발생한 것으로 보인다.

한국 지명셋에 대해 유형별로 자세히 분석하면 poi·city(0.00~0.15 km)는 오차가 매우 작기 때문에 동적 누적에 적합하고, street·park·district(0.37~4.43 km)는 오차 편차가 있으나 동적 누적으로 대응 가능한 수준이다. state·region은 미국 지명셋에서와 같이 중심점 오차가 수십 km 이상에 달하며, 교차검증으로 해소되지 않는 체계적 불일치를 가진다. 이로부터 PRE_REG_TYPES = {state, region}의 설계 결정이 실험적으로 뒷받침된다.

마지막 실험은 18개 GeoServer 레이어에 대해 레이어별 Strong·Medium 질의를 각 8개씩 자동 생성하였다(18 × 8 × 2 = 288개). Strong 질의는 레이어 고유 속성명(예: PERSONS, LAND_KM, CFCC 등)을 포함하며, Medium 질의는 속성명 없이 지역명·주제 개념 단서(예: "맨해튼 도로", "세계 해안선")만을 사용한다. 모든 질의는 한국어로 구성된다.

LLM 기반 레이어 분류 성능을 정확도로 측정한 결과, Strong 95.38%, Medium 98.61%, 전체 97.22%를 달성하였다. 속성명에 대한 단서가 없는 Medium이 Strong보다 낮을 것이라는 초기 예상과 달리, 두 조건 모두 높은 정확도를 기록하였으며 Medium이 오히려 소폭 상회하는 역전이 나타났다.

이 결과는 분류 난이도가 Strong·Medium 구분보다 레이어 간 의미론적 유사성에 의해 결정됨을 시사한다. Strong 질의에서의 오분류는 NAME, STATE_NAME 등 여러 레이어에 공유된 동명 속성이 주된 원인으로, 속성명 자체가 대상 레이어를 특정하는 고유 신호로 작용하지 못할 때 발생했다.

V. 결 론

본 논문은 공간 데이터 시각화 환경에서 자연어 질의를 ECQL 필터로 변환하는 LLM 기반 도구 에이전트 시스템을 제안하였다. 지리 시각화 맥락에서 발생하는 모호 수치 표현의 임계값 결정과 지명의 공간 좌표 변환 문제를 각각 전담하는 메커니즘을 설계하고 실험으로 유효성을 검증하였다.

모호 표현은 방향과 유형을 조합한 매핑 규칙으로 백분위 통계값에 대응시켜 LLM의 임계값 선택을 제어하였으며, 실험 결과 지정 백분위수로 일관되게 수렴함을 확인하였다. 지오코딩에서는 지명을 7개 유형으로 분류하여 poi·street·park·district·city 5개 유형은 교차검증 기반 동적 누적(두 소스 간 중심점 오차 중앙값 0.30 km)으로, state·region은 사전 등록으로 각각 처리하였다. 레이어 자동 분류에서는 97.22%의 정확도를 달성하였다.

향후 연구에서는 AND/OR 기반 복합 ECQL 질의를 지원하도록 에이전트를 확장하고, 지명 정규화와 지오코딩 검증 및 보정 프롬프트를 통해 한국어 지리 정보 처리 정확도를 향상시킬 계획이다. 또한 대규모 국내외 지명 데이터셋을 활용해 제안 기법의 일반화 가능성을 검증하고자 한다.

References

[1] Z. Li and H. Ning, "Autonomous GIS: the next-generation AI-powered GIS", International Journal of Digital Earth, Vol. 16, No. 2, pp. 4668-4686, Dec. 2023. https://doi.org/10.1080/17538947.2023.2278895.

[2] E. J. Pour, Y. Chen, A. Kesminas, and A. Rajabifard, "An Automated Framework for Natural Language-Based Spatial Query Using Large Language Models", Journal of Geovisualization and Spatial Analysis, Vol. 10, No. 1, Art. no. 19, Jun. 2026. https://doi.org/10.1007/s41651-026-00260-9.

[3] T. Yu, R. Zhang, K. Yang, M. Yasunaga, D. Wang, Z. Li, J. Ma, I. Li, W. 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. EMNLP 2018, Brussels, Belgium, pp. 3911-3921, Nov. 2018. https://doi.org/10.18653/v1/D18-1425.

[4] V. Zhong, C. Xiong, and R. Socher, "Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning", arXiv preprint, arXiv:1709.00103, pp. 1-12, Aug. 2017. https://arxiv.org/abs/1709.00103.

[5] M. Pourreza and D. Rafiei, "DIN-SQL: Decomposed In-Context Learning of Text-to-SQL with Self-Correction", Adv. Neural Inf. Process. Syst. (NeurIPS), New Orleans, Louisiana, USA, Vol. 36, pp. 36339-36348, Dec. 2023. https://arxiv.org/abs/2304.11015.

[6] S. Scheider, E. Nyamsuren, H. Kruiger, and H. Xu, "Geo-analytical question-answering with GIS", International Journal of Digital Earth, Vol. 14, No. 1, pp. 1-14, Jan. 2021. https://doi.org/10.1080/17538947.2020.1738568.

[7] Y. Zhang, C. Wei, S. Wu, Z. He, and W. Yu, "GeoGPT: An assistant for understanding and processing geospatial tasks", International Journal of Applied Earth Observation and Geoinformation, Vol. 131, Art. no. 103976, Jul. 2024. https://doi.org/10.1016/j.jag.2024.103976.

[8] A. Mansourian and R. Oucheikh, "ChatGeoAI: Enabling Geospatial Analysis for Public through Natural Language, with Large Language Models", ISPRS International Journal of Geo-Information, Vol. 13, No. 10, Art. no. 348, Oct. 2024. https://doi.org/10.3390/ijgi13100348.

[9] X. Wang, S. Du, C.-C. Feng, X. Zhang, and X. Zhang, "Interpreting the Fuzzy Semantics of Natural-Language Spatial Relation Terms with the Fuzzy Random Forest Algorithm", ISPRS International Journal of Geo-Information, Vol. 7, No. 2, Art. no. 58, Feb. 2018. https://doi.org/10.3390/ijgi7020058.

[10] A. Yang, C. Fu, Q. Jia, X. Hou, and Z. Zhu, "Evaluating and enhancing spatial cognition abilities of large language models", International Journal of Geographical Information Science, Vol. 39, No. 9, pp. 2009-2044, Sep. 2025. https://doi.org/10.1080/13658816.2025.2490701.

[11] S. Hou, H. Jiao, Z. Liu, L. Xie, G. Chen, S. Wu, X. Guan, and H. Wu, "GeoSQL-Eval: First Evaluation of LLMs on PostGIS-Based NL2GeoSQL Queries", Expert Systems with Applications, Vol. 320, Art. no. 132122, Jul. 2026. https://doi.org/10.1016/j.eswa.2026.132122.

[12] E. Hamzei, M. Tomko, and S. Winter, "Translating Place-Related Questions to GeoSPARQL Queries", Proc. ACM Web Conference 2022 (WWW '22), Lyon, France, pp. 902-911, Apr. 2022. https://doi.org/10.1145/3485447.3511933.

[13] R. Pierdicca, N. Muralikrishna, F. Tonetto, and A. Ghianda, "On the Use of LLMs for GIS-Based Spatial Analysis", ISPRS International Journal of Geo-Information, Vol. 14, No. 10, Art. no. 401, Oct. 2025. https://doi.org/10.3390/ijgi14100401.

[14] 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", Adv. Neural Inf. Process. Syst. (NeurIPS), Vancouver, BC, Canada, Vol. 33, pp. 1-19, Dec. 2020. https://arxiv.org/abs/2005.11401.

[15] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, "ReAct: Synergizing Reasoning and Acting in Language Models", Proc. ICLR 2023, Kigali, Rwanda, pp. 1-33, May 2023. https://arxiv.org/abs/2210.03629.

[16] T. Schick, J. Dwivedi-Yu, R. Dessì, R. Raileanu, M. Lomeli, L. Zettlemoyer, N. Cancedda, and T. Scialom, "Toolformer: Language Models Can Teach Themselves to Use Tools", Adv. Neural Inf. Process. Syst. (NeurIPS), New Orleans, Louisiana, USA, Vol. 36, pp. 68539-68551, Dec. 2023. https://arxiv.org/abs/2302.04761.

[17] D. Lassiter and N. D. Goodman, "Context, scale structure, and statistics in the interpretation of positive-form adjectives", Semantics and Linguistic Theory (SALT 23), Santa Cruz, California, USA, pp. 587-610, Aug. 2013. https://doi.org/10.3765/salt.v23i0.2658.

[18] O. R. Abbasi, F. Welscher, G. Weinberger, and J. Scholz, "The World As Large Language Models See It: Exploring the Reliability of LLMs in Representing Geographical Features", arXiv preprint arXiv:2506.00203, pp. 1-9, May 2025.https://arxiv.org/abs/2506.00203.

[19] nl2geo-benchmark, Hugging Face, 2026. https://huggingface.co/datasets/clearbench/nl2geo-benchmark. [accessed: Jul. 27, 2026]

저자소개

최 권 택 (Kwon-Taeg Choi)



2006년 2월 : 연세대학교 컴퓨터공학과(공학석사)
2011년 2월 : 연세대학교 컴퓨터공학과(공학박사)
2016년 3월 ~ 현재 : 강남대학교 소프트웨어응용학부 교수
관심분야 : 가상현실, 증강현실, 모바일컴퓨팅, 기계학습, HCI