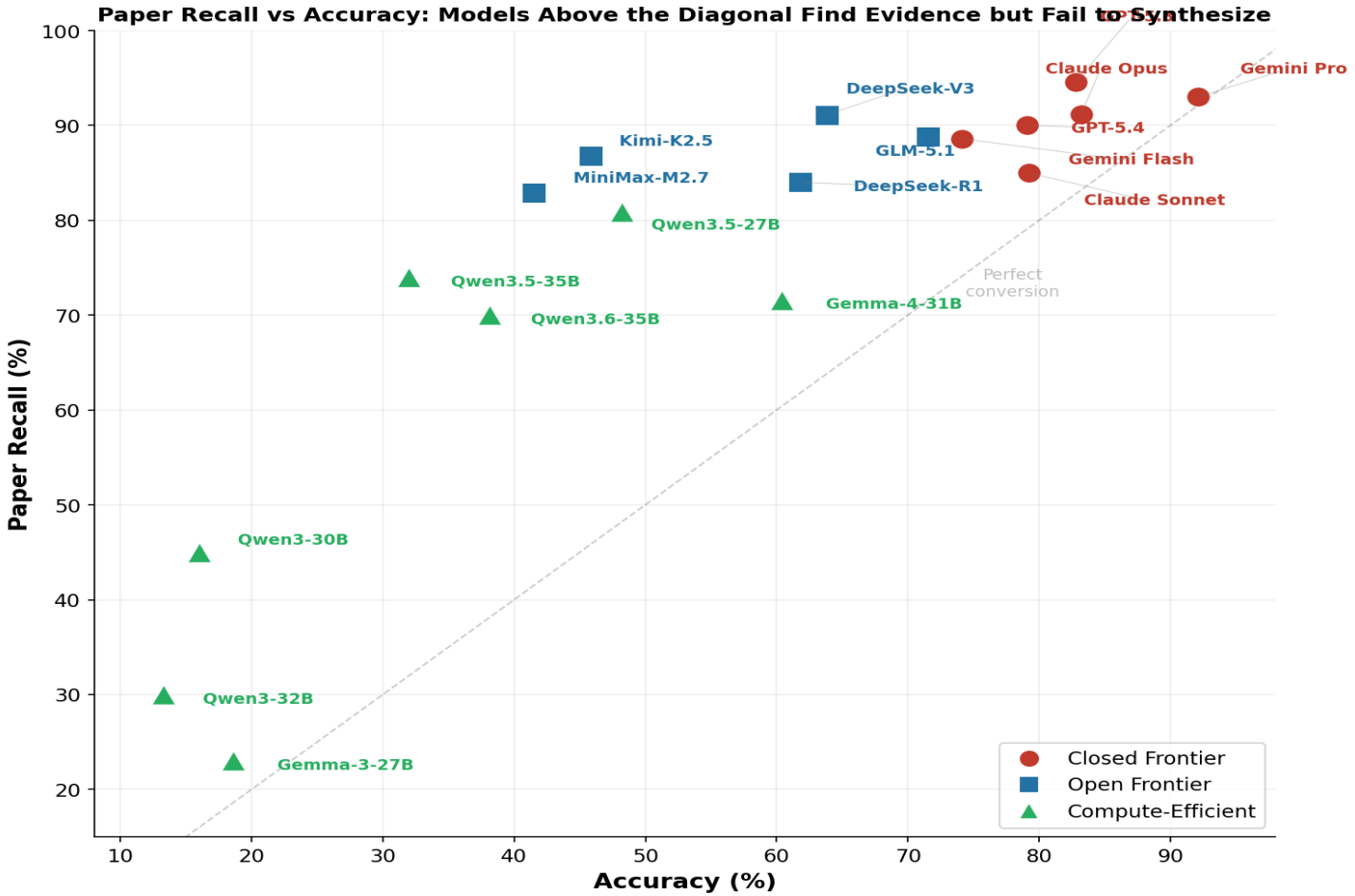
ArxivHop: A Diagnostic Benchmark For Multi-Turn Agent Evaluation



Natural Language Processing & Artificial Intelligence

2026. 04. 23.

박찬희

KOREA UNIVERSITY 고려대학교 KOREA UNIVERSITY

0. Motivation

1. Codex, Claude Code 등 작동 방식

- 파일 시스템에 대한 권한을 얻고, read/search 등의 내부 툴로 동작
- Search Agent의 시대 Ex) BrowseComp … WebSearch API 기반 에이전트 평가
- 그렇다면 검색 풀도 짜고, recall도 보면서, end-task 성능도 측정하는 벤치마크가 필요하지 않을까?

2. 논문 기반 멀티홉 벤치마크

- 논문은 자체적으로 citation이라는 edge로 연결되어 있는 트리 구조를 형성한다.
- 특히 논문이 매년 지수함수 그래프로 쏟아져 나오는 CS 분야는 이런 멀티홉 데이터를 수집하기 최적의 분야임
- 그래프 구조로 서치 트리를 확보하고, 논문으로 QA를 만든다면?

3. Tool 기반 샌드박스 평가 방식

- 툴/코드 등은 텍스트로 짜고, API나 평가 로직으로 샌드박스 평가가 가능.
- 즉, 외부 API가 없는 상태에서 단순 LLM만 집어넣어서 천하제일 무술대회를 펼칠 수 있을 것이다.

=> 지식, 검색, 툴 활용 능력을 모두 평가할 수 있는 벤치마크가 탄생

1. Introduction

문제 인식:

- 연구자 81%가 LLM을 연구 워크플로우에 활용
- 에이전트는 직접 검색, 읽기, 합성을 수행
  (RAG와 달리 검색 자체가 모델의 판단)
- 기존 벤치마크는 pass/fail만 보고
  → 어떤 능력이 부족한지 진단 불가

핵심 갭:

- Knowledge QA: 모델이 아는 것을 측정
- Code 벤치마크: 출력의 정확성을 측정
- Agent 벤치마크: 작업 완료 여부를 측정
- 어떻게 작동하는지는 아무도 측정하지 않음

ArxivHop: 진단 도구 (Diagnostic Instrument)

- 테스트가 아닌, 측정 도구
- 에이전트 행동을 분해:
  1) Paper Recall — 증거를 찾았는가?
  2) Conversion — 증거에서 추론했는가?
  3) Failure Attribution — 왜 틀렸는가?

핵심 사례:

Kimi: 논문을 87% 찾지만 51%만 정답으로 전환
→ 합성(synthesis) 병목
Gemma-4: 72%만 찾지만 74% 전환
→ 탐색(navigation) 병목

→ Pass/fail 벤치마크는 이 차이를 볼 수 없음

2. Related Work

Benchmark	N	Know.	Tool	Multi-hop	Multi-turn	Retrieval	Diagnostic
Knowledge QA							
HotpotQA [10]	113K	✓		✓		✓	
GPQA [13]	448	✓					
SciCode [26]	338	✓					
LitQA2 [1]	248	✓				✓	
Code							
HumanEval [15]	164		✓				
MBPP [16]	974		✓				
LiveCodeBench [17]	880		✓				
SWE-bench [3]	2,294		✓		✓		
Agent							
BFCL v4 [21]	2,000		✓		✓		
τ-bench [22]	240		✓		✓		
WebArena [18]	812		✓	✓	✓	✓	
GAIA [19]	466		✓	✓	✓	✓	
AgentBoard [20]	1,013		✓		✓		
PaperArena [25]	317	✓	✓	✓	✓	✓	
ArxivHop (Ours)	1,011	✓	✓	✓	✓	✓	✓

Table 1: Comparison of ArxivHop with existing benchmarks. N: number of samples. ✓ = supported.

3. ArxivHop

구축 파이프라인:

953 seeds → 17,983 chains → 1,651 MCQs
→ Answerability Filter (68 제거)
→ Consensus Filter: 3-모델 앙상블 (434 제거)
  · 57%는 no_context 수렴으로 제거
→ Human audit: 7명 전수 검토 (43 drop)
→ 최종 1,011 samples

3-Trap 디스트랙터 설계:

- **seed_only** → 탐색 실패 (인용 미추적)
- **wrong_paper** → 검색 오류 (잘못된 논문)
- **no_context** → 증거 없이 그럴듯한 답변 선호

→ 오답 시 어떤 유형인지로 실패 원인 진단

7-Tool Sandbox (30pt 예산):

- 탐색 (1pt): get_paper_info, get_references, list_sections, keyword_search
- 읽기 (5pt): read_section
- 무료 (0pt): think, submit_answer

데이터셋 통계:

- 7,205 논문, 9개 CS 학회, 2022-2025
- 450K+ 인용 관계
- 568 single-target, 443 multi-target
- 486 depth-1, 525 depth-2

행동 다양성:

- 9,099 궤적 중 6,752 고유 경로 (74%)
- 174개 전 모델 정답, 16개 전 모델 오답

4. Experiment

#	Model	Failure Attribution (% of errors)			Navigation (%)			Tool Management (per sample avg.)			
		NC	WP	SO	Paper	Section	Conv.	Bud.	Turn	Rd.P	Acc.
1	Gemini 3 Pro	55.0	10.0	33.8	93.0	86.6	95.8	19.8	9.4	62.5	**92.1**
2	Claude Opus 4.6	60.6	10.0	29.4	91.2	86.2	88.0	19.2	10.6	60.0	83.2
3	GPT-5.3-codex	81.6	3.4	10.9	94.6	79.6	85.9	25.2	8.5	47.2	82.8
4	GLM-5.1	70.1	3.8	9.4	88.8	84.4	77.5	23.3	11.5	49.6	71.5
5	DeepSeek-V3	74.6	3.3	16.1	91.1	84.4	68.2	26.9	15.6	54.7	63.8
6	Gemma-4-31B	61.0	5.2	14.8	71.6	68.6	73.7	22.6	10.9	42.5	60.4
7	Qwen3.5-27B	59.5	2.1	19.5	80.9	78.0	55.6	23.5	12.1	47.9	48.2
8	Kimi-K2.5	73.0	2.2	13.1	86.8	81.1	51.2	24.7	13.3	51.4	45.8
9	MiniMax-M2.7	71.1	4.6	13.7	82.9	75.7	47.1	24.6	10.5	46.6	41.5

Tier: Closed Open Efficient Best

Table 2: Main results on 9 flagship models, ranked by accuracy. **Failure Attribution**: error distribution—NC (no-context plausibility), WP (wrong paper), SO (seed only). **Navigation**: paper/section recall and conversion rate. **Tool Management**: Bud. = budget points used (of 30), Turn = interaction turns, Rd.P = read precision (gold reads / total reads, %).

4. Experiment – 핵심 분석

Evidence-Ignored Rate (증거 무시율)

골드 논문을 읽고도 오답한 비율:
- Claude Opus: 11.1%
- GPT-5.3: 11.5%
- DeepSeek: 29.3%
- Kimi: 47.7%
- MiniMax: 50.9%

→ Kimi/MiniMax는 증거를 찾아도
절반을 무시하고 그럴듯한 답변 선택
→ 도구 사용 능력은 있으나
증거 기반 추론 훈련이 부족

상위 모델 실패 분석:

Gemini Pro(92%), Opus(83%), GPT-5.3(83%)
→ 공통 오답: 28개 (2.8%)에 불과
→ 각 모델은 ~90개의 고유 오답 보유

오답 시 distractor 프로필:
- GPT-5.3: NC 82% (플러시빌리티 함정 취약)
  → GPT-5.4 생성 distractor와 친숙도 편향?
- Gemini: SO 34% (파라메트릭 지식으로 답변)
- Opus: NC 61%, SO 29% (중간)

→ 동일 정확도에서도 실패 양상이 다름
→ 정확도만으로는 볼 수 없는 진단 신호

4. Experiment – 행동 분석

Think 도구: 비추론 모델의 계획 도구

모델 내 비교 (think 사용 vs 미사용):
- MiniMax: +26.1pp ($p<0.0001$)
- Kimi: +17.0pp ($p=0.0004$)
- DeepSeek: +10.0pp ($p=0.028$)

추론 모델 (내부 CoT 보유):
- GPT-5.4: +0.7pp (유의하지 않음)
- DeepSeek-R1: -2.3pp (유의하지 않음)

Opus의 선택적 think 사용:
- get_references 후: 91% (계획)
- read_section 후: 75% (합성)
- get_paper_info 후: 14% (생략)

→ 반사적이 아닌, 의도적 사용

회복 전략: 모델 패밀리별 특성

잘못된 논문 방문 후:
- Claude: think_replan (55%) → 73.9%
- GPT: list_sections (42%) → 75.7%
- DeepSeek: navigate (35%) → 54.8%
- Kimi: navigate (41%) → 41.8%

→ Claude는 반성, GPT는 재구성,
DeepSeek/Kimi는 그냥 전진

최종 제출 전략:

think → submit이 최고 정확도:
- Sonnet: 95.1% (vs read_more 76%)
- MiniMax: 72.6% (vs read_more 48%)

→ 같은 모델, 같은 논문, 다른 행동 → 다른 결과

4. Experiment – Ablation

Condition	GPT-5.3-codex	DeepSeek-chat	Gemma-4-31B
Zero-tool	48.6%	18.3%	31.7%
No-search	79.4%	56.5%	(pending)
Full	82.8%	63.8%	60.4%

Table 4: Tool ablation: accuracy across three conditions (zero-tool, no-search, full).

안정성 (3회 실행, n=1,011):
- ±0.8-1.3pp 표준편차
- 모델 순위 안정적, 역전 없음
- 샘플 당 ~90% 일치율

Closed-Frontier, Open-Frontier, Open-Efficient 모델군 중 하나를 선정하여 ablation 진행

1) zero-tool 2) no search 3) 3-run robustness

- 도구 기여도와 파라메트릭 지식은 역상관: 약한 모델일수록 도구 혜택 큼
- Search 기여: +3-7pp, depth-2에 집중 (search 없으면 -11pp)
- No-think ablation (DeepSeek, GLM): -2~4pp → 인과적 효과는 미미
- Qwen thinking 모드: 최대 ±3pp (vLLM 설정 이슈 가능성)
- Gemini Pro zero-tool 74.8%: 가장 높은 파라메트릭 지식 → 오염 우려

4. Cross-Benchmark Correlation

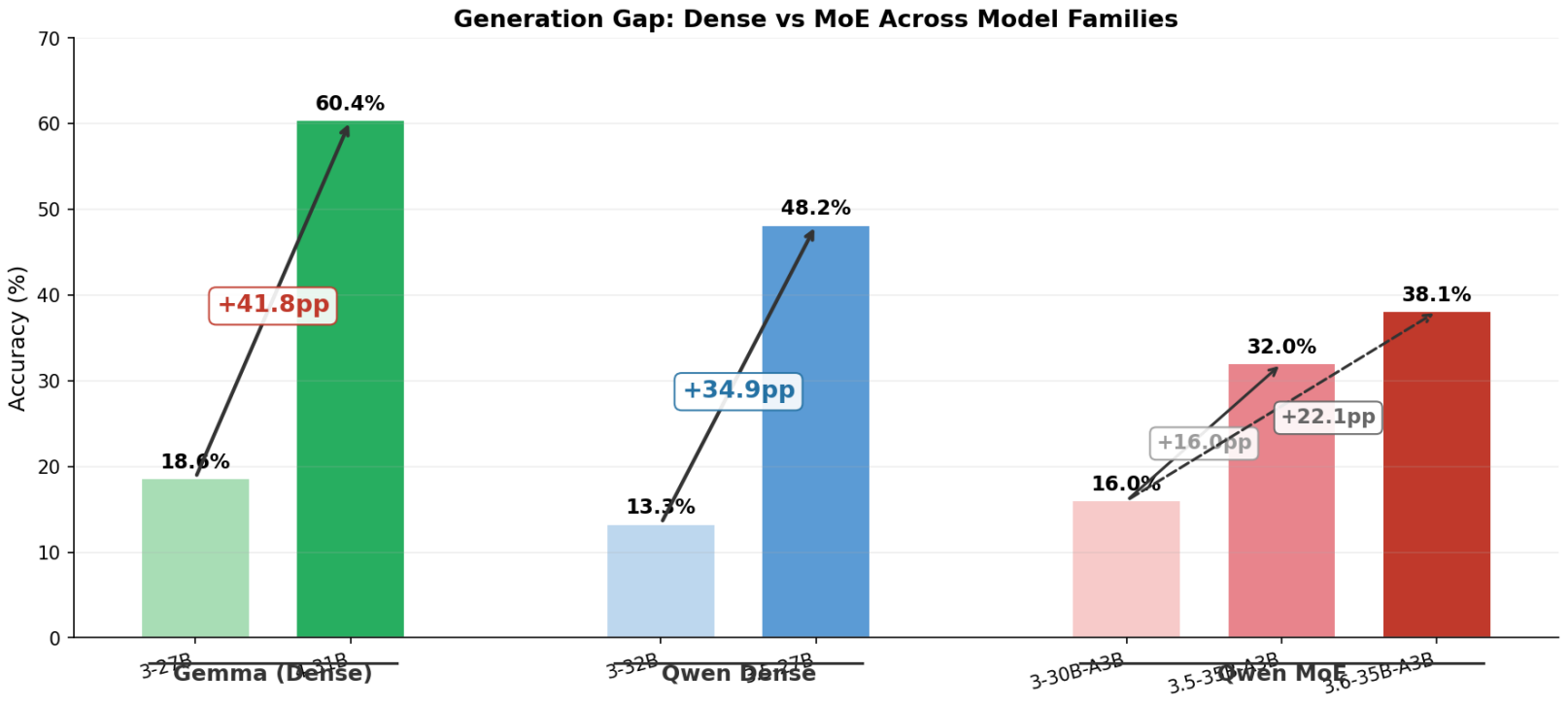
ArxivHop correlates strongly with agent/tool benchmarks

Benchmark	N	Spearman ρ	Category
BrowseComp	9	0.895***	Agent
tau2-bench	8	0.738*	Tool-use
HLE (no tools)	12	0.811**	Hard reasoning
Terminal-Bench	13	0.802***	Code agent
GPQA-D	13	0.775**	Reasoning

기존 벤치마크와 모델 랭킹 비교

- 기존 벤치마크에서 잘하는 모델은 ArxivHop에서도 잘한다.
- ArxivHop은 기존 벤치마크처럼 모델 우열을 충분히 구분하면서 동시에 유의미한 분석 툴을 제공해준다.

4. Generation Gap



Gemma, Qwen으로 보는 모델 변천사

- Gemma4, Qwen3.6 모두 나온지 한 달이 안 된 최신 모델
- 기존 3세대 모델과의 가장 큰 차이점은
  '싱글턴' 상황에서 어려운 문제를 잘 푸냐,
  '멀티턴' 상황에서 여러 결과물을 종합하는 능력이 있느냐
- 즉, 에이전트는 다름 아닌, 기존의 성능을 멀티턴에서도 유지하는 게 핵심 아닐까?

5. Conclusion

1. ArxivHop: 진단 도구 — 테스트가 아닌 측정 도구
1,011 samples, 7 tools, 3-trap 디스트랙터, zero parsing failures

2. 분해 프레임워크: paper recall × conversion × failure attribution
탐색, 합성, 파라메트릭 의존도를 단일 실행에서 분리

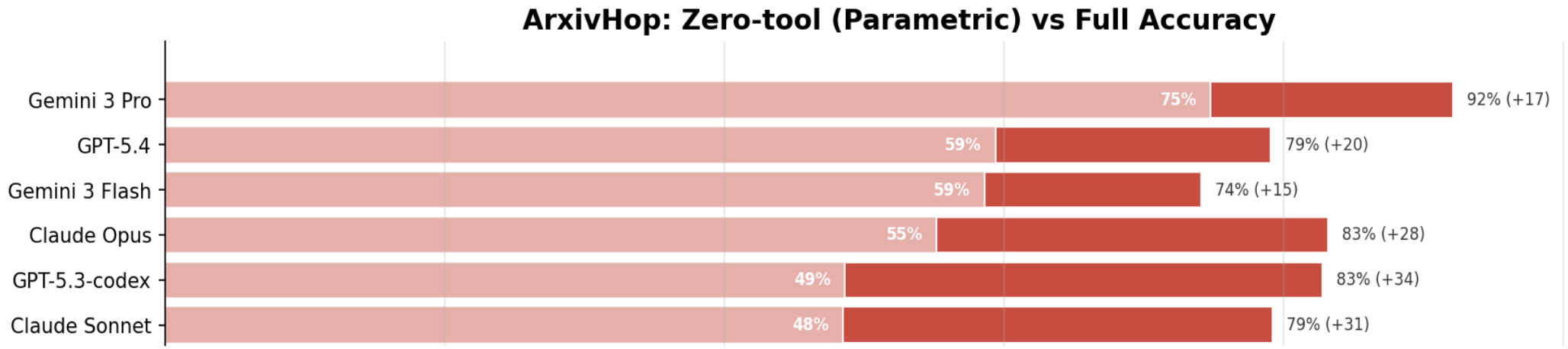
3. 18 모델 실증 검증:
- 동일 정확도에서의 전략 다양성 (Claude vs GPT at 83%)
- 보편적 플러시빌리티 함정 (오답의 60-80%)
- Conversion gap: Kimi 87% recall → 51% conversion (실행 가능한 진단)
- 상위 모델 간 실패 비중첩: 28/1,011만 공통 실패 (2.8%)
- 세대 간 격차 +25-42pp (에이전트 특화 훈련에 의한)

ArxivHop은 어떤 능력을 개선해야 하는지를 식별하게 해준다.

6. 고민 지점

1. Closed Frontier의 독주 : zero-tool 성능이 이미 저 너머로

- 기존의 벤치마크는 항상 SOTA 모델들이 2-30점으로 허덕이는 모습을 보여주며 시작함
- 반면에 ArxivHop은 Gemini-3-pro가 92점을 달성하고 시작함
- 사람들이 주로 사용하는 Opus, Codex, Gemini가 모두 80점 이상이라는 데에서
  '과연 난이도가 적절했는가?'



6. 고민 지점

2. 못하는 모델은 분석이 가능하다. 그런데 잘하는 모델은?

- 이미 90점을 맞는 모델에 분석이 필요한가?
- 무슨 기준을 들이밀어도 recall 93, conversion 96 찍은 모델을 어떻게 분석할 것인가?

3. Causality와 Correlation의 차이

- 모델 A와 모델 B의 tool 사용 전략을 비교한다고 했을 때,
  1) 모델 A가 성능이 좋은 이유는 특정 tool 패턴을 많이 보였기 때문이다.
  2) 모델 A가 성능이 좋은 이유는 그냥 모델 A가 B보다 더 똑똑하기 때문이다.

- Findings와 Good analysis의 차이는 뭘까?
  실험 결과를 충실히 설명함과 동시에 유의미한 모델 간 차이를 설명할 수 있으려면?

Thank you