



MiroThinker: Pushing the Performance Boundaries of Open-Source Research Agents via Model, Context, and Interactive Scaling

251125

가짜연구소 허의주

MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Scailing 제안

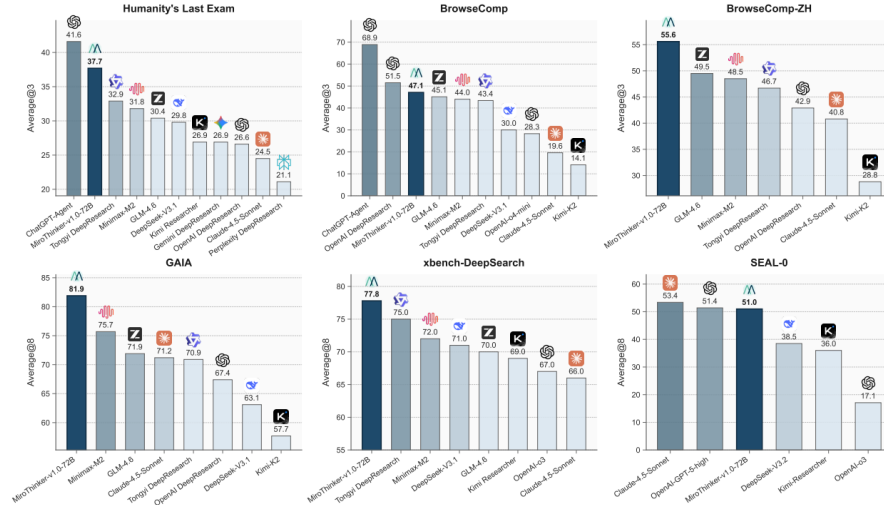
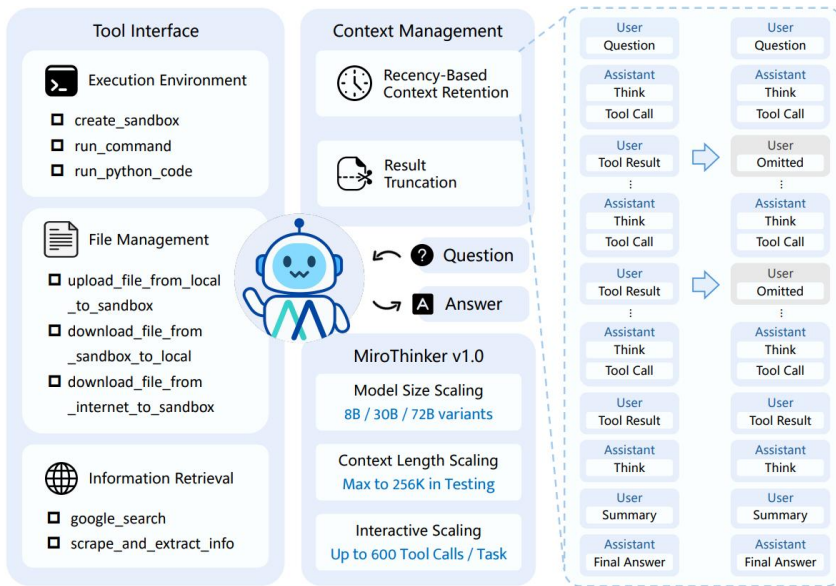


Figure 1: Comparison of MiroThinker with state-of-the-art agents and agentic foundation models.



1. Motivation

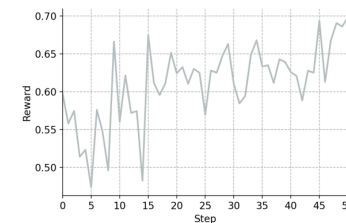
- 연구 에이전트의 격차: ChatGPT Agent나 Claude Research와 같은 독점적인 시스템과 기존 오픈 소스 모델들과 명확한 성능 격차가 존재, 폐쇄적인 상용 Agent로 인한 투명성과 재현성의 제약 존재로 오픈 소스 커뮤니티의 연구가 어려움
- MiroThinker는 모델 크기, 컨텍스트 길이와 함께 상호 작용 깊이를 세 번째 핵심 차원으로 확장, 오픈 소스 시스템의 성능 한계를 극복하고자 함

2. Core Methodology: Interactive Scailing

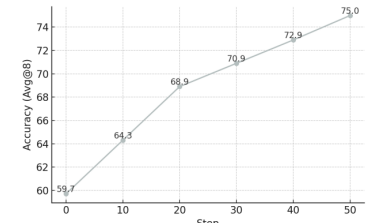
- 모델 수준에서 상호 작용 스케일링 탐구, 환경 피드백과 외부 정보 획득을 활용
- 3단계 훈련 파이프라인
 - 1) SFT로 기본 행동 확립, 2) DPO로 의사 결정 정렬, 3) RL로 창의적 탐색 및 일반화
- 대규모 상호 작용 지원: 256K의 컨텍스트 윈도우로 task 당 최대 600회의 도구 호출
- GRPO를 사용, 롤아웃 궤적을 이용해 정책 모델을 업데이트 (완전 온라인 정책 훈련)
- 보상 설계: 솔루션 정확성을 위주로 측정, 형식 지침을 따르지 않은 경우 패널티 부여

3. Experimental Results

- 오픈 소스 SOTA 달성
- 상용 모델 초월
- 상호작용 깊이 강화
- 스케일링 특성 확립



(a) Training reward across training steps.



(b) Val acc on GAIA-Text-103 across training steps.

MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Introduction

- 대규모 언어 모델(LLMs)의 급속한 발전은 AI 패러다임을 정적인 텍스트 생성기에서 동적이고 도구로 확장된 에이전트로 전환시킴
- 이러한 패러다임 속에서 에이전트의 연구 능력은 새로운 지능의 영역으로 평가받음
- 하지만 오픈소스 에이전트 성능과 독점적 시스템들(ChatGPT, Claude 등)의 격차가 커지며 투명성, 재현성 등 혁신을 제약하게 됨
- 이러한 도전에 대응하여, 오픈소스 시스템의 성능 한계를 세 가지 주요 차원에 따라 확장하는 MiroThinker v1.0을 제안
- 특히 MiroThinker는 모델 수준에서 상호 작용 깊이 스케일링을 체계적으로 훈련하여 성능 개선을 위한 제3의 차원으로 탐구
- 차별점: 긴 추론 과정 중 성능 저하 위험이 있는 일반적 LLM 시간 스케일링과 달리, 환경 피드백과 외부 정보 획득 기능을 활용
- 기술 사양: 모델은 256K 컨텍스트 윈도우를 통해, 작업당 최대 600회의 도구 호출을 수행할 수 있음
 - 기존 오픈 소스 모델의 100회 미만 호출 능력에서 크게 도약
- 주요 성과: 단순한 ReAct 에이전트임에도 몇몇 벤치마크에서 SOTA 성능을 달성
 - 72B 모델은 GAIA에서 81.9%, HLE에서 37.7%, BrowseComp에서 47.1%, BrowseComp-ZH에서 55.6%의 정확도 달성
 - 특히 HLE 벤치마크에서는 GPT-5-high와 같은 상용 모델의 결과를 능가

MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scaling 제안

❖ Introduction

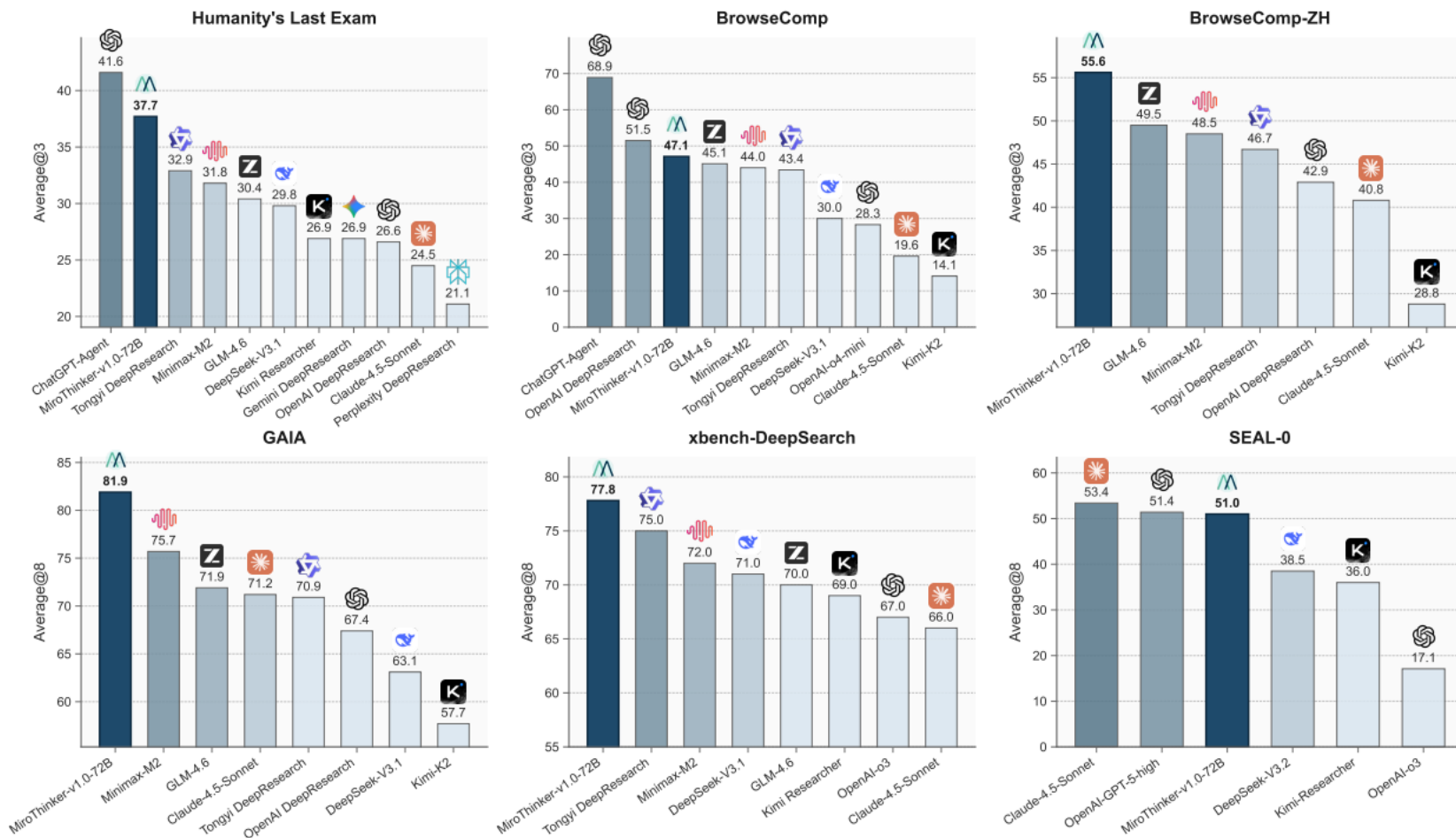


Figure 1: Comparison of MiroThinker with state-of-the-art agents and agentic foundation models.

MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Related Works

1. 에이전트 기반 모델 (Agent Foundation Models, AFMs)

- 배경: 최근 연구는 LLM의 에이전트 역량인 계획, 추론, 자율 행동 등을 강화하는데 중점을 두고 있음
- 특징: AFMs는 일반적 언어 이해를 넘어 의사결정, 도구 사용, 외부 환경과의 상호 작용과 같은 에이전트 지향적 능력을 모델 훈련에 통합
- 동향: 특히 코딩 에이전트와 검색 에이전트에 집중; 도구 기반 문제 해결, 검색 증강 추론 및 자율적 작업 실행 능력 향상이 주 목적
- 모델: ChatGPT, Claude, Grok, MiniMax 등

2. 심층 연구 모델(Deep Research Models)

- 배경: AFMs의 발전 추세를 이어, 복잡한 Multi-hop 추론과 장문 맥락, 검색 집약적 작업을 위해 전문화된 LLM 기반 에이전트로 도입
- 특징: 동적인 정보 탐색과 반복적 계획(iterative planning)을 워크플로우에 통합, 자율적인 지식 획득 및 종합을 통해 포괄적 답변 생성
- 동향: 주요 AI 연구소들은 독점적 심층 연구 시스템을 개발, 오픈 소스 커뮤니티에서도 많은 심층 연구 모델이 발표됨
- 모델: OpenAI Deep Research, Claude Research, Grok DeepSearch, Tongyi DeepResearch 등

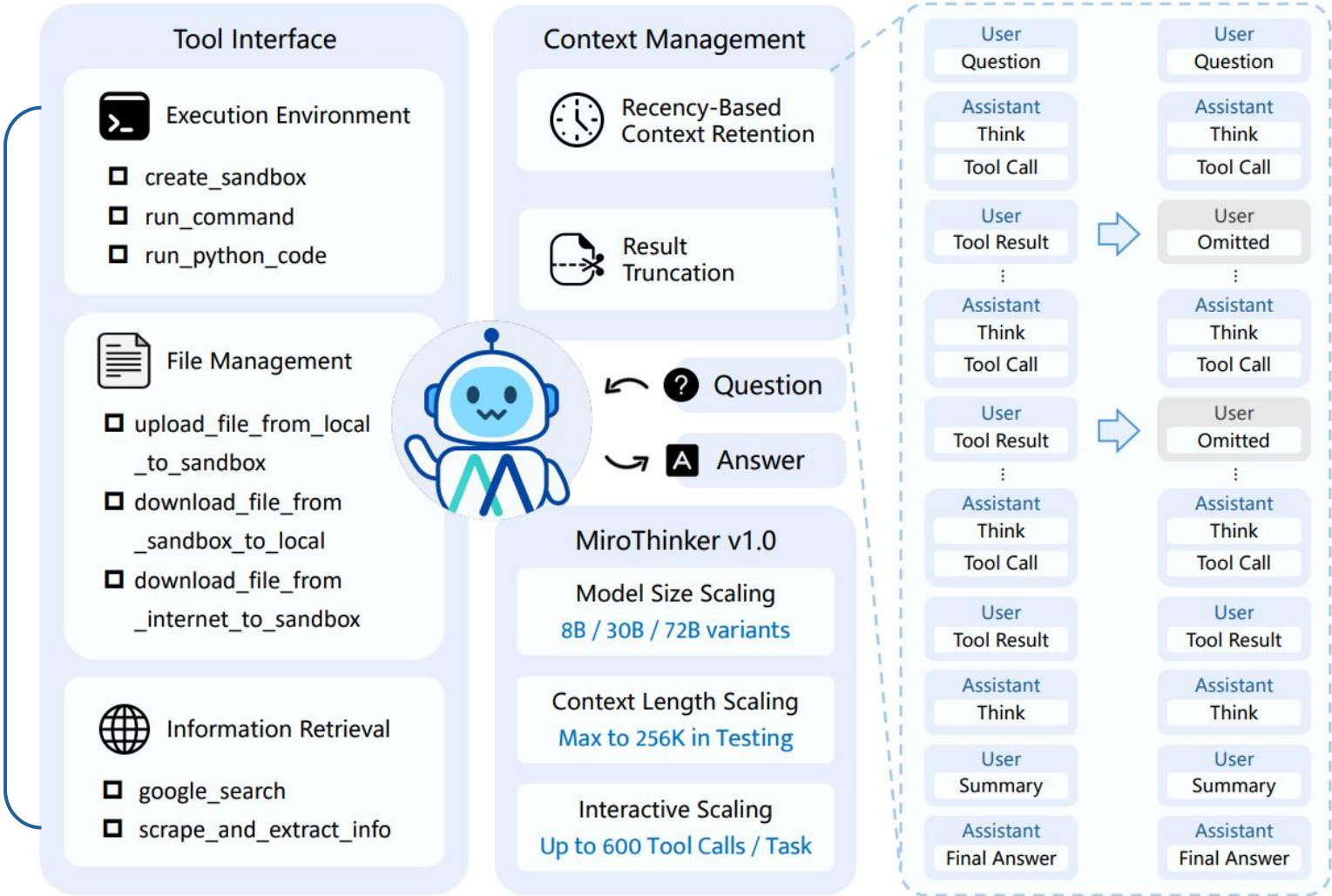
MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Agentic Workflow

ReAct 패러다임 구성 요소: $H_t = \{(T_1, A_1, O_1), \dots, (T_{t-1}, A_{t-1}, O_{t-1})\}$

구조화된
도구 인터페이스



MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

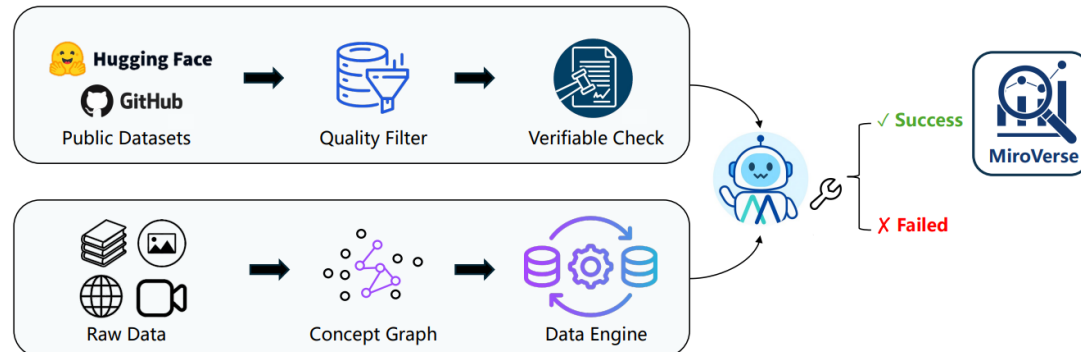
성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Data Construction - 1

• MultiDocQA Synthesis

▶ 상호 연결된 웹 문서를 Multi-hop QA 쌍으로 변환하는 QA 합성 파이프라인 설계

- Document Corpus 구축: Wikipedia, Common Crawl 등 하이퍼링크 구조가 풍부하고 사실적 신뢰도가 높은 문서에서 문서 코퍼스 구성
전처리 과정에서 텍스트를 정리하면서도 하이퍼링크 보존
- 문서 샘플링 및 그래프 구축: 편향되지 않은 표본을 유지하기 위해 문서 샘플링, 각 시드 문서의 내부 하이퍼링크를 따라 연결되는 지식 그래프 구축
- 문서 통합: 구축한 지식 그래프 내부에서만 참조가 유지되도록 link pruning 수행, 통합된 아티클이 현재 컨텍스트 내에서 일관된 참조 유지 보장
- 사실 추출: 구축된 그래프 내 통합 문서에 대해 중앙 주제와 연결되고 cross-document reasoning을 요구하는 핵심 사실을 추출
- 제약 조건 난독화: 추출된 사실들에 대해 심층적 추론을 필요로 하는 간접적인 제약 조건으로 변환하여 난이도를 높임
(예: 2023년 3월 15일 → 2020년대 봄)
- 질문 생성: LLM에게 난독화된 제약 조건들을 조합하여 질문을 합성하도록 프롬프트를 제공하여 진정한 Multi-hop 추론을 요구



MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Data Construction - 2

• Agentic Trajectory Synthesis

- ▶ 모델이 추론 능력을 갖추고 도구를 효과적으로 사용하는 훈련을 하기 위해 **고품질의 다양한 에이전트 궤적 데이터를 생성하는 단계**
 - Agent Paradigms
 1. ReAct Single-Agent: 반복적인 “생각-행동-관찰” 주기를 통해 다단계의 추론과 적응적 의사 결정을 처리
 2. MiroFlow Multi-Agent: 구조화된 프로토콜을 통해 여러 전문 에이전트가 복잡한 워크플로우에 대해 **분업, 조정 및 집단 추론의 협업 궤적 생성**
 - Tool Invocation Mechanisms
 1. Function Calling: 사전 정의된 함수 인터페이스를 통한 전통적이고 구조화된 도구 호출
 2. Model Context Protocol, MCP: 도구와 더 자연스럽게 상호 작용하는 유연한 프로토콜, 더 복잡한 도구 구성 및 **동적 도구 발견**을 지원
 - Diverse Data Synthesis: GPT-OSS, DeepSeek-V3.1 등 여러 선도적인 LLM을 사용하여 궤적 생성
 - 단일 모델 편향 완화, 데이터의 풍부함과 커버리지 확보

• Open-Source Data Collection

- 합성 데이터 외에도 커버리지와 추론 다양성을 넓히기 위해 다양한 오픈소스 QA 데이터셋을 보완적으로 사용
- 사용 데이터셋: MuSiQue, HotpotQA, WebWalkerQA-Silver, MegaScience, TaskCraft, QA-Expert-Multi-Hop-V1.0, WikiTables 등

MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Training Pipeline

- MiroThinker는 Qwen2.5 및 Qwen3 모델을 기반, 세 가지 단계로 구성된 파이프라인으로 훈련

1. Agentic Supervised Fine-tuning, SFT

- 데이터 구성: SFT 데이터셋은 태스크 지침과 생각-행동-관찰로 구성된 전문가 궤적의 쌍으로 이루어짐
- 데이터 품질 관리: Raw trajectories 내부에 발생한 응답 내 반복, 교차 응답 중복, 유효하지 않은 도구 호출과 같은 노이즈 제거
- 훈련 목표: 각 궤적은 사용자와 어시스턴트 간 다중 턴 대화로 처리,
훈련 중에는 실제 도구 실행이 수행되지 않으며 관찰 결과는 데이터 내부에 미리 기록되어 다음 대화의 문맥으로 모델에게 제공

2. Agentic Preference Optimization, DPO

- 데이터 구성: 선호 데이터셋은 태스크 지침, 선호 궤적, 비선호 궤적의 쌍으로 구성
- 선호 기준: 주로 최종 답변의 정확성에 따라 결정됨, 미리 정의된 계획 길이, 단계 수 등의 고정된 에이전트 패턴이나 인위적인 휴리스틱에의 의존을 피함
- 품질 관리: 선택된 궤적은 일관된 추론 과정, 명시적 계획, 명확하고 올바른 최종 답변을 포함해야 함
- 훈련 목표: SFT 모델을 개선하기 위해 DPO 목적함수를 사용

MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

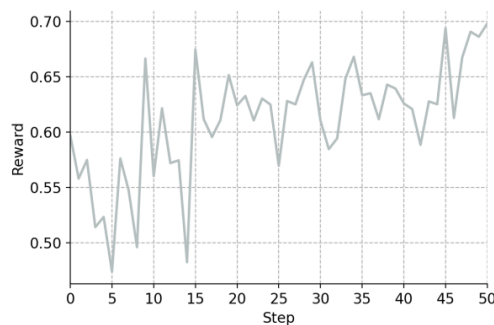
성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Training Pipeline

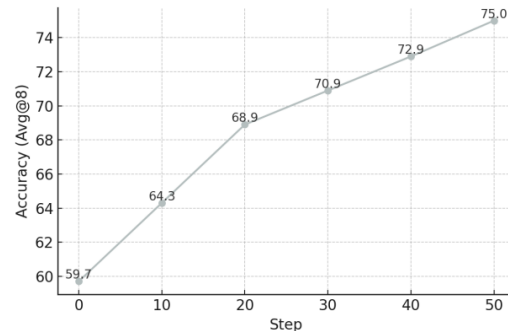
- MiroThinker는 Qwen2.5 및 Qwen3 모델을 기반, 세 가지 단계로 구성된 파이프라인으로 훈련

3. Agentic Reinforcement Learning, RL

- 학습 방법: GRPO를 사용, Rollout trajectory를 사용하여 각 궤적마다 Policy 모델을 한 번만 업데이트하는 완전 온라인 정책 훈련 수행
- 환경 설정: 수천 건의 동시 에이전트 롤아웃을 지원할 수 있는 확장 가능한 환경 모음 구축
예시) 실시간 다중 출처 검색, 웹 스크래핑 및 요약, Python 코드 실행, LLM 채점 시스템 등
- Streaming Rollout Acceleration: Agent RL은 LLM과 환경 간 다중 왕복이 필요하므로, 다양한 궤적의 완료 시간이 과한 Long-tail의 양상을 띌 수 있어 Streaming Rollout Acceleration을 적용, 모든 미완료 Task를 다음 Task queue로 넘겨 효율성 증가
- 보상 설계: 솔루션 정확성과 형식 지침 준수 여부를 이용하여 보상을 설계 $R(x, H) = \alpha_c R_{\text{correct}}(H) - \alpha_f R_{\text{format}}(H),$
- Trajectory Curation: RL 학습 신호 품질 보장을 위한 필터링 파이프라인
- 훈련 목표: GRPO는 프롬프트 당 여러 궤적을 샘플링, 그룹 평균에 대한 이점을 계산해 정책을 최적화



(a) Training reward across training steps.



(b) Val acc on GAIA-Text-103 across training steps.

MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Experiments

• Experimental Setup

- 훈련 후 평가를 위해 Qwen2.5 및 Qwen3 모델을 기반으로 MiroThinker v1.0을 초기화
- Temperature: 1.0, Top-p: 0.95, 최대 턴 수: 600, 컨텍스트 길이: 256K 토큰, 컨텍스트 보존 예산: 5
- 모든 벤치마크 성능은 LLM-as-Judge로 평가
- 분산이 높은 벤치마크는 K회 독립 실행 후 평균점수(avg@k)로 보고
(HLE, BrowseComp, BrowseComp-ZH, WebWalkerQA, FRAMES는 avg@3, GAIA, xbench-DeepSearch, SEAL-0은 avg@8 사용)

MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Experiments

• Experimental Result

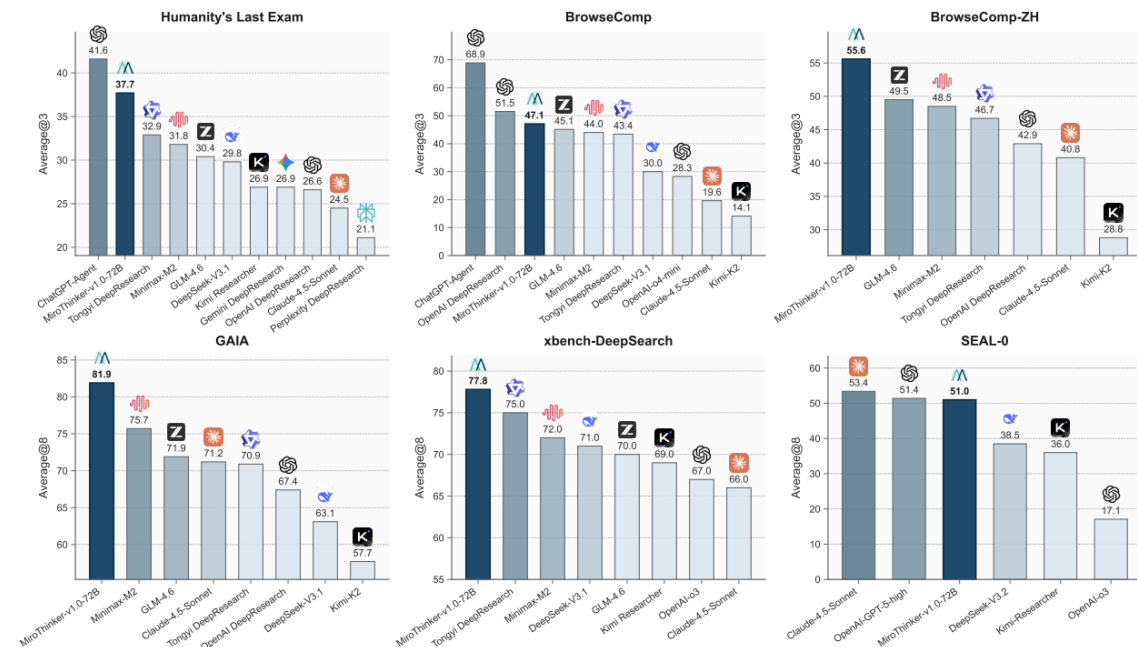


Figure 1: Comparison of MiroThinker with state-of-the-art agents and agentic foundation models.

Benchmarks	Humanity's Last Exam	Browse Comp	Browse Comp-ZH	GAIA	xbench DeepSearch	WebWalker QA	FRAMES	SEAL-0
<i>Foundation Models with Tools</i>								
GLM-4.6 [4]	30.4	45.1	49.5	71.9	70.0	—	—	—
Minimax-M2 [3]	31.8	44.0	48.5	75.7	72.0	—	—	—
DeepSeek-V3.1 [5]	29.8	30.0	49.2	63.1	71.0	61.2	83.7	—
DeepSeek-V3.2 [5]	27.2	40.1	47.9	63.5	71.0	—	80.2	38.5
Kimi-K2-0905 [2]	21.7	7.4	22.2	60.2	61.0	—	58.1	25.2
Claude-4-Sonnet [7]	20.3	12.2	29.1	68.3	64.6	61.7	80.7	—
Claude-4.5-Sonnet [7]	24.5	19.6	40.8	71.2	66.0	—	85.0	53.4
OpenAI-o3 [49]	24.9	49.7	58.1	—	67.0	71.7	84.0	17.1
OpenAI-GPT-5-high [1]	35.2	54.9	65.0	76.4	77.8	—	—	51.4
<i>Research Agents</i>								
OpenAI DeepResearch [24]	26.6	51.5	42.9	67.4	—	—	—	—
ChatGPT-Agent [8]	41.6	68.9	—	—	—	—	—	—
Kimi-Researcher [23]	26.9	—	—	—	69.0	—	78.8	36.0
WebExplorer-8B-RL [21]	17.3	15.7	32.0	50.0	53.7	62.7	75.7	—
DeepMiner-32B-RL [17]	—	33.5	40.1	58.7	62.0	—	—	—
AFM-32B-RL [16]	18.0	11.1	—	55.3	—	63.0	—	—
SFR-DeepResearch-20B [50]	28.7	—	—	66.0	—	—	82.8	—
Tongyi-DeepResearch-30B [11]	32.9	43.4	46.7	70.9	75.0	72.2	90.6	—
MiroThinker-v1.0-8B	21.5±0.4	31.1±1.6	40.2±2.9	66.4±3.2	60.6±3.8	60.6±0.8	80.6±0.5	40.4±2.6
MiroThinker-v1.0-30B	33.4±0.2	41.2±1.3	47.8±1.1	73.5±2.6	70.6±2.2	61.0±0.2	85.4±0.8	46.8±3.2
MiroThinker-v1.0-72B	37.7±0.5	47.1±0.7	55.6±1.1	81.9±1.5	77.8±2.6	62.1±0.6	87.1±0.9	51.0±2.0

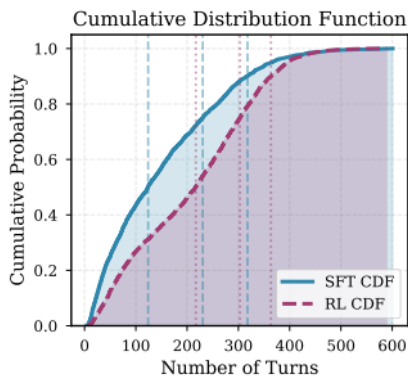
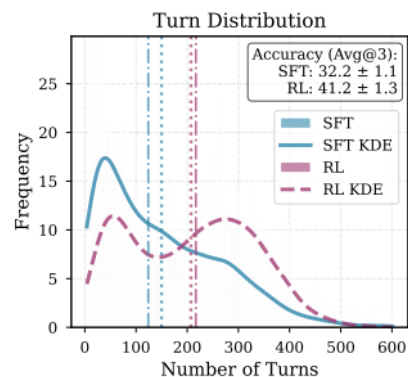
MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

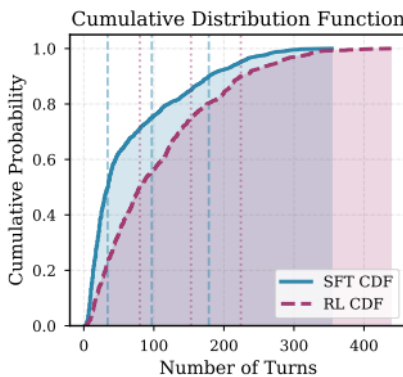
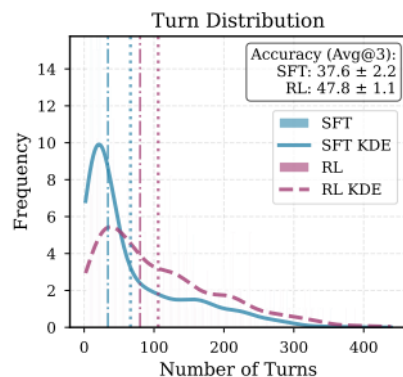
❖ Experiments

• Experimental Result

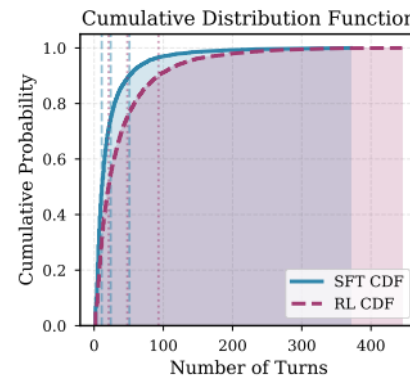
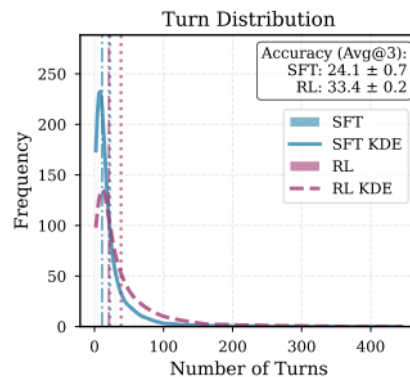
- 분석을 통해 강화 학습 훈련이 에이전트의 행동 패턴을 어떻게 변화시키고 성능을 향상시키는지 입증



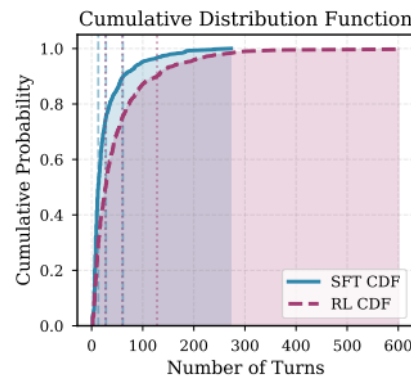
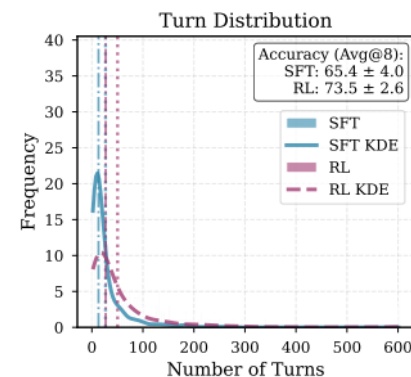
(a) BrowseComp



(b) BrowseComp-ZH



(c) HLE



(d) GAIA

MiroThinker: 모델, 컨텍스트, 상호 작용 스케일링을 통한 오픈 소스 에이전트 성능 한계 확장

성능 개선을 위한 제3의 차원으로서 Interactive Depth Scailing 제안

❖ Experiments

• Limitation

- 도구 사용 품질 저하: RL 튜닝된 모델이 SFT모델보다 도구를 더 자주 호출하는 경향이 있으며, 이 중 일부는 미미하거나 중복된 기여를 하는 것을 발견
- 과도하게 긴 추론 CoT: RL은 정확도를 높이기 위해 모델이 긴 응답을 생성하도록 장려하는 경향이 존재
→ 지나치게 길고 반복적이며 읽기 어려운 CoT를 초래하여 작업 속도를 늦춤
- 언어 혼용: 비영어권 입력 시 모델의 내부 추론이나 중간 출력에 영어가 혼합됨
- 제한적 샌드박스 기능: 모델이 코드 실행 및 파일 관리 도구에 완전히 숙련되지 못함