

Atom-Searcher: Enhancing Agentic Deep Research via Fine-Grained Atomic Thought Reward

Yong Deng*, Guoqing Wang*, Zhenzhe Ying*, Xiaofeng Wu*, Jinzhen Lin, Wenwen Xiong, Yuqin Dai, Shuo Yang, Zhanwei Zhang, Qiwen Wang, Yang Qin, Yuan Wang, Quanxing Zha, Sunhao Dai, Changhua Meng

Ant Group

*Core Contributors

Large language models (LLMs) exhibit remarkable problem-solving abilities, but struggle with complex tasks due to static internal knowledge. Retrieval-Augmented Generation (RAG) enhances access to external information, yet remains limited in multi-hop reasoning and strategic search due to rigid workflows. Recent advancements in agentic deep research empower LLMs to autonomously reason, search, and synthesize information. However, current approaches relying on outcome-based reinforcement learning (RL) face critical issues such as conflicting gradients and reward sparsity, limiting performance gains and training efficiency. To address these, we first propose Atomic Thought, a novel LLM thinking paradigm that decomposes reasoning into fine-grained functional units. These units are supervised by Reasoning Reward Models (RRMs), which provide Atomic Thought Rewards (ATR) for fine-grained guidance. Building on this, we propose Atom-Searcher, a novel RL framework for agentic deep research that integrates Atomic Thought and ATR. Atom-Searcher uses a curriculum-inspired reward schedule, prioritizing process-level ATR early and transitioning to outcome rewards, accelerating convergence on effective reasoning paths. Experiments on seven benchmarks show consistent improvements over the state-of-the-art. Key advantages include: (1) Atom-Searcher scales computation at test-time. (2) Atomic Thought provides supervision anchors for RRMs, bridging deep research tasks and RRMs. (3) Atom-Searcher exhibits more interpretable, human-like reasoning patterns.

Code: <https://github.com/antgroup/Research-Venus>

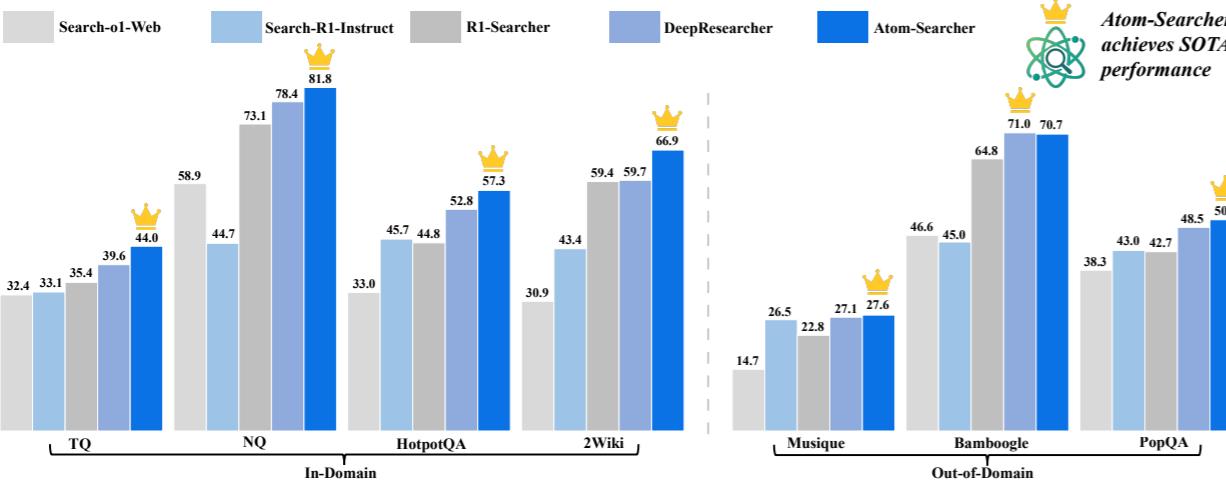


Figure 1 Atom-Searcher achieves SOTA performance on both in-domain and out-of-domain benchmarks.

Atom-Searcher: Ajanik Derin Araştırmayı İnce Taneli Atomik Düşünce Ödülü ile Güçlendirme

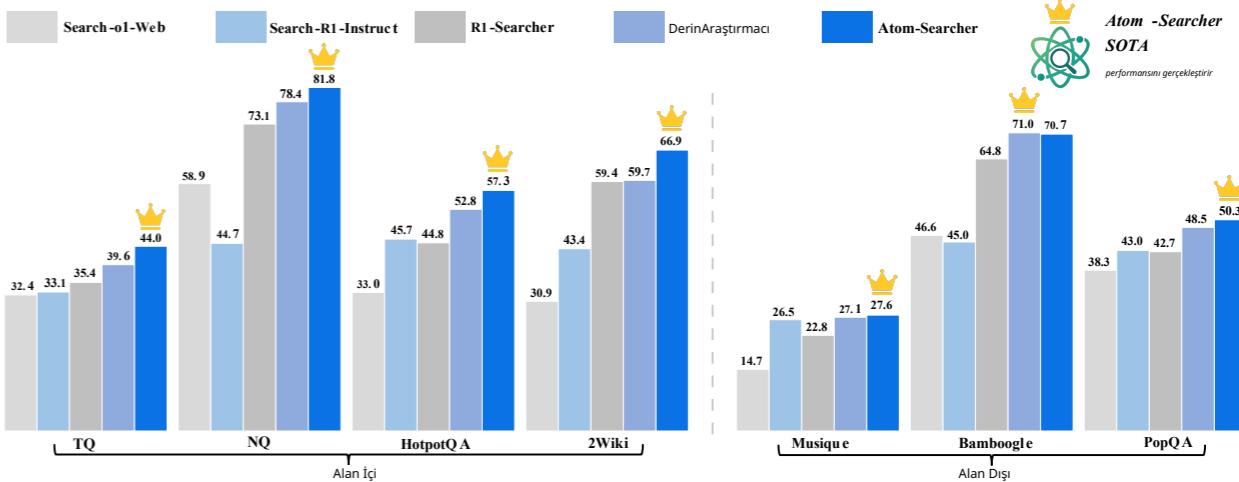
Yong Deng*, Guoqing Wang*, Zhenzhe Ying*, Xiaofeng Wu*, Jinzhen Lin, Wenwen Xiong, Yuqin Dai, Shuo Yang, Zhanwei Zhang, Qiwen Wang, Yang Qin, Yuan Wang, Quanxing Zha, Sunhao Dai, Changhua Meng

Ant Group

*Temel Katkıda Bulunanlar

Büyük dil modelleri (LLM'ler) olağanüstü problem çözme yeteneklerine sahiptir, ancak statik iç bilgiler nedeniyle karmaşık görevlerde zorlanmaktadır. Arama destekli üretim (RAG), dış bilgiye erişimi artırır ancak katı iş akışları nedeniyle çok aşamalı muhakeme ve stratejik aramada sınırlı kalmaktadır. Son gelişmeler, ajanik derin araştırmayı LLM'lerin bağımsız şekilde muhakeme yapmasını, arama gerçekleştirmesini ve bilgi sentezlemesini mümkün kılmıştır. Ancak sonuç odaklı takviyeli öğrenmeye dayanan mevcut yaklaşımlar; çatışan gradyanlar ve ödül seyrekligi gibi kritik sorunlarla karşılaşmaktadır, bu da performans iyileşmelerini ve eğitim verimliliğini sınırlamaktadır. Bunları ele almak için, öncelikle akıl yürütmemi ince taneli fonksiyonel birimlere ayıran yeni bir LLM düşünme paradigmasi olan Atomik Düşünce'yi öneriyoruz. Bu birimler, detaylı rehberlik sağlayan Atomik Düşünce Ödülleri (ADO) sunan Akıl Yürütme Ödül Modelleri (AYÖM) tarafından denetlenmektedir. Bunun üzerine, Atomik Düşünce ve ADO'yu entegre eden ajanik derin araştırma için yeni bir TL çerçevesi olan Atom-Searcher'ı öneriyoruz. Atom-Searcher, süreç düzeyindeki ADO'yu erken aşamada önceliklendirdip sonuç ödüllerine geçiş yaparak, etkili akıl yürütme yollarında yakınsamayı hızlandıran müfredat esinli bir ödül takvimi kullanır. Yedi karşılaştırma ölçütü üzerindeki deneyler, en son teknolojilere kıyasla tutarlı gelişmeler göstermektedir. Ana avantajlar şunlardır: (1) Atom-Searcher, test zamanında hesaplamayı ölçeklendirir. (2) Atomik Düşünce, derin araştırma görevleri ile AYÖM'ler arasında köprü kurarak AYÖM'ler için denetim noktaları sağlar. (3) Atom-Searcher, daha yorumlanabilir ve insan benzeri akıl yürütme modelleri

Kod:<https://github.com/antgroup/Research-Venus>



Şekil 1 Atom-Searcher, alan içi ve alan dışı karşılaştırma ölçütlerinde SOTA performansını gerçekleştirir.

1 Introduction

Although large language models (LLMs) demonstrate impressive language understanding and logical reasoning abilities Yang et al. (2025a); Guo et al. (2025); Hurst et al. (2024), their capacity to solve complex problems ultimately hits a ceiling due to the static nature of their internal knowledge representation Wang et al. (2024a); Jin et al. (2024). Retrieval-Augmented Generation (RAG) Lewis et al. (2020) offers solution by equipping LLMs with external information sources, enhancing the relevance, accuracy, and timeliness of their responses Gao et al. (2023); Fan et al. (2024). However, RAG’s static workflows, making them ineffective at handling real-world questions that require sophisticated multi-hop reasoning and strategic search planning Singh et al. (2025), as they often fail to construct correct search paths for complex problems Yao et al. (2023). To mitigate these limitations, a new search paradigm, termed **Agentic Deep Research** system, has been proposed, which enables autonomous reasoning, on-demand searching, and iterative information synthesis. Demonstrations from recent deep research systems by OpenAI OpenAI (2025) and Google Google (2024) reveal several key advantages of this paradigm: 1) *Comprehensive Understanding*: Effectively handles complex, multi-step queries that challenge traditional methods Wei et al. (2022); 2) *Enhanced Synthesis*: Integrates diverse and even conflicting sources into coherent, informative outputs Cheng et al. (2025); 3) *Reduced User Effort*: Automates tedious search processes, easing users’ cognitive and manual burden Sami et al. (2024).

Early implementations of agentic deep research relied on prompt engineering Song et al. (2024); Kim et al. (2024) and supervised fine-tuning (SFT) Zhang et al. (2024). Yet, prompt-based methods rely heavily on LLMs’ instruction-following and long-context capabilities, whereas SFT tends to generalize poorly across domains Chu et al. (2025). More recently, post-training LLMs via reinforcement learning with outcome-based rewards (outcome-based RL) has yielded notable gains in reasoning performance Guo et al. (2025); OpenAI (2024). Building on this insight, recent advances Dai et al. (2025); Yang et al. (2025b,c) (e.g. Search-R1 Jin et al. (2025) and DeepResearcher Zheng et al. (2025)) treat the search tool as part of the environment and apply outcome-based RL to enable end-to-end optimization of the entire workflow, resulting in more performant and generalizable agentic deep research systems. Although outcome-based RL has shown promise, it remains insufficient in fully advancing agentic deep research, for the following reasons: 1) *Gradients Conflicts*: In the outcome-based RL paradigm, an incorrect final answer results in the entire trajectory being penalized Lightman et al. (2023), even when intermediate reasoning process or research strategies are effective. This coarse-grained reward design introduces potential gradient conflicts between intermediate reasoning steps and final answers, which hinders the model from discovering better reasoning capabilities and research strategies, thereby limiting its generalization ability. 2) *Reward sparsity*: Outcome-based RL relies solely on the final answer to generate rewards Du et al. (2024), resulting in each training sample providing only sparse feedback. This severely limits the efficiency of policy optimization, as it increases the reliance on larger training datasets and prolonged training schedules.

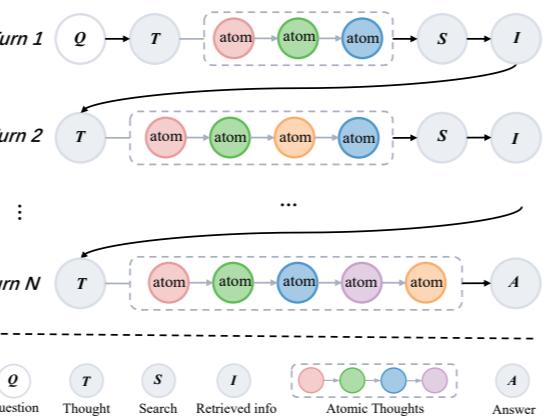


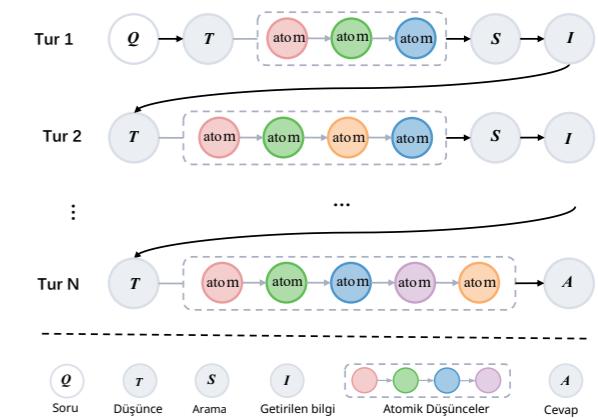
Figure 2 Atomic Thought paradigm automatically decomposes each <think> into finer-grained functional units <atom-think> during the rollout.

1 Giriş

Büyük dil modelleri (LLM’ler), Yang ve ark. (2025a); Guo ve ark. (2025); Hurst ve ark. (2024) tarafından belirtilen şekilde etkileyici dil anlama ve mantıksal akıl yürütme yetenekleri sergilemektedir; ancak karmaşık problemleri çözme kapasiteleri, Wang ve ark. (2024a); Jin ve ark. (2024) tarafından vurgulandığı üzere, içsel bilgi temsillerinin statik doğası nedeniyle nihai bir sınırla karşılaşmaktadır. Arama destekli üretim (RAG) Lewis ve ark. (2020) tarafından önerilmiş olup, LLM’leri dış bilgi kaynaklarıyla donatarak cevapların alaka düzeyini, doğruluğunu ve güncellliğini artırmaktadır Gao ve ark. (2023); Fan ve ark. (2024).

Ancak, RAG’ın statik iş akışları, Singh ve ark. (2025) tarafından ifade edildiği üzere, ileri düzeyde çok adımlı akıl yürütme ve stratejik arama planlaması gerektiren gerçek dünya sorularını etkin şekilde yanıtlandırmada yetersiz kalmaktadır, çünkü karmaşık problemler için doğru arama yollarını oluşturmaktan sıklıkla başarısız olmaktadır Yao ve ark. (2023). Bu kısıtlamaları azaltmak amacıyla, özerk akıl yürütme, talep üzerine arama ve yinelemeli bilgi sentezine olanak tanıyan Agentic Deep Research sistemi adlı yeni bir arama paradigmı önerilmiştir. OpenAI OpenAI (2025) ve Google Google (2024) tarafından geliştirilen son ajanik derin araştırma sistemleri, bu paradigmın birkaç temel avantajını ortaya koymaktadır: 1) *Kapsamlı Anlayış* : Wei ve ark. (2022) tarafından bildirildiği üzere, geleneksel yöntemlerin üstesinden gelemediği karmaşık, çok adımlı sorguları etkin şekilde ele alır; 2) *Geliştirilmiş Sentez* : Cheng ve ark. (2025) tarafından belirtildiği üzere, çeşitli ve hatta çelişen kaynakları tutarlı ve bilgilendirici çıktılar halinde birleştirir; 3) *Azaltılmış Kullanıcı Çabası* : Sami ve ark. (2024) tarafından ifade edildiği üzere, kullanıcıların bilişsel ve manuel yükünü azaltarak zahmetli arama süreçlerini otomatikleştirir.

Ajanik derin araştırmaların erken uygulamaları, Song ve ark. (2024) tarafından belirtildiği gibi prompt mühendisliğine dayanmaktadır; Kim ve ark. (2024) ile Zhang ve ark. (2024) tarafından gerçekleştirilen denetimli ince ayar (SFT) yöntemlerine dayanıyordu. Bununla birlikte, istem tabanlı yöntemler büyük ölçüde LLM’lerin talimat takibi ve uzun bağlam yeteneklerine dayanırken, SFT genellikle alanlar arasında iyi genelleme yapamamaktadır Chu vd. (2025). Daha yakın zamanda, sonuç temelli ödüllerle takviyeli öğrenme kullanılarak post eğitim LLM’lerde (sonuç temelli RL) anlamlı artışlar gözlemlenmiştir Guo vd. (2025); OpenAI (2024). Bu anlayış temelinde, son gelişmeler Dai vd. (2025); Yang vd. (2025b,c) (ör. Search-R1 Jin vd. (2025) ve DerinAraştırmacı Zheng vd. (2025)) arama aracını ortamın bir parçası olarak ele almakta ve tüm iş akışının uçtan uca optimize edilmesini sağlamak için sonuç temelli RL uygulamaktadır; bu da daha performanslı ve genellenebilir ajanik derin araştırma sistemleri ortaya çıkarmaktadır. Sonuç temelli RL umut vaat etse de, ajanik derin araştırmayı tam anlamıyla ilerletmede yetersiz kalmaktadır, bunun sebepleri aşağıda sıralanmaktadır: 1) Gradyan çatışmaları: Sonuç temelli RL paradigmında, hatalı nihai cevap tüm yörüklenen cezalandırılmasına yol açar Lightman vd. (2023); oysa ara aşamalardaki akıl yürütme süreçleri veya araştırma stratejileri etkili olabilir. Bu kaba taneli ödül tasarımları, ara aşama akıl yürütme adımları ile nihai cevaplar arasında potansiyel gradyan çatışmalarına neden olur; bu durum modelin daha iyi akıl yürütme yetenekleri ve araştırma stratejileri keşfetmesini engelleyerek genelleme yetisini sınırlar. 2) Ödül seyrekligi: Sonuç temelli RL yalnızca nihai cevaba dayalı ödüller üretir Du vd. (2024); bu da her eğitim örneğinin sadece seyrek geribildirim sağlanması ile sonuçlanır. Bu durum politika optimizasyonunun verimliliğini ciddi şekilde sınırlar; çünkü daha büyük eğitim veri setlerine ve uzun eğitim programlarına olan bağımlılığı artırır.



Şekil 2 Atomik Düşünce paradigmı, her <düşün> elemanı <atom-düşün> olarak ayrılmaktadır.

To address these challenges, we begin by introducing **Atomic Thought**, a novel LLM thinking paradigm that decomposes reasoning into fine-grained functional units, called Atomic Thoughts, guiding LLMs to engage in clearer and more in-depth reasoning, as illustrated in Figure 2. For example, reasoning operations like <Reflection> and <Verification> serve as Atomic Thoughts. Their interactions constitute the functional backbone of the reasoning process. To promote generalization, we avoid manual decomposition of Atomic Thoughts and instead encourage the model to autonomously induce them from reasoning processes. Building on this definition, we employ a Reasoning Reward Model (RRM) to score the generated Atomic thoughts and construct fine-grained **Atomic Thought Reward (ATR)**. The ATR serves as an auxiliary signal to calibrate the outcome reward, thereby mitigating gradient conflicts during policy optimization. To aggregate the ATR and outcome reward, we design an curriculum-inspired strategy. During the early stages of training, the model is in a solution path exploration phase: while it may struggle to produce fully correct final answers, it can more easily develop partially correct reasoning traces. Relying solely on outcome rewards at this stage may induce severe gradient conflicts, thus requiring stronger calibration. As training advances, the alignment between reasoning and answers improves, reducing gradient conflicts and necessitating weaker calibration to avoid introducing excessive noise. Accordingly, we employ a linearly decaying weighting scheme, wherein the contribution of the ATR is gradually reduced as training proceeds. In addition, the hybrid reward incorporates process-level signals into the outcome-based reward, alleviating the problem of reward sparsity. Building on the above components, we propose **Atom-Searcher**, a novel RL framework for agentic deep research, aimed at advancing the performance frontier of agentic deep research models.

We conducted experiments on seven benchmarks covering both in-domain and out-of-domain tasks, demonstrating that Atom-Searcher achieves significant performance gains compared to the state-of-the-art (SOTA) baseline. Furthermore, we designed experiments to highlight the following advantages of Atom-Searcher: (1) Atom-Searcher effectively scales computation during test-time. (2) Atomic Thoughts provide supervision anchors for RRMs, effectively bridging deep research tasks and RRMs. (3) Atom-Searcher exhibits more interpretable, human-like reasoning patterns

In summary, our main contributions are as follows:

- We first introduce Atomic Thought, a novel LLM thinking paradigm that decomposes reasoning into fine-grained functional units, effectively guiding LLMs to engage in clearer and more in-depth reasoning.
- Building on Atomic Thought, we design fine-grained Atomic Thought Reward and construct a curriculum-inspired aggregation strategy to integrate ATR with the outcome reward. This reward modeling alleviates gradient conflicts and reward sparsity during policy optimization.
- Building on Atomic Thought paradigm, ATR and the proposed reward aggregation strategy, we introduce Atom-Searcher, a novel RL framework for agentic deep research, aimed at advancing the performance frontier of agentic deep research.
- We demonstrated that Atom-Searcher achieves significant performance improvements over the SOTA baseline on seven benchmarks covering both in-domain and out-of-domain tasks. Additionally, we designed experiments to highlight a range of impressive advantages of Atom-Searcher.

Bu zorlukları ele almak amacıyla, akıl yürütme işlevsel birimlere ayıran ve LLM'lerin daha açık ve derinlemesine akıl yürütmesini sağlayan yeni bir LLM düşünce paradigması olan Atomik Düşünce'yi tanıtıyoruz; bu, Şekil 2'de gösterilmiştir. Örneğin, <Yansıma> ve <Doğrulama> gibi akıl yürütme işlemleri Atomik Düşünceler olarak işlev görür. Bu etkileşimler, akıl yürütme sürecinin işlevsel omurgasını oluşturur. Genelleşmeyi teşvik etmek için, Atomik Düşüncelerin manuel ayrıtırılmasından kaçınıyor ve modelin bunları akıl yürütme süreçlerinden kendi kendine türetmesini destekliyor. Bu tanımlamaya dayanarak, oluşturulan Atomik Düşünceleri puanlamak için bir Akıl Yürütme Ödül Modeli (RRM) kullanıyor ve ince taneli Atomik Düşünce Ödülü (ADO) oluşturuyoruz. ADO, çıktı ödülüne kalibre etmek için yarıdıcı bir sinyal olarak işlev görür ve böylece politik optimizasyon sırasında gradyan çatışmalarını hafifletir. ADO ile çıktı ödülüne birleştirmek amacıyla, müfredat temelli bir strateji tasarladık. Eğitimin erken aşamalarında model, bir çözüm yolu keşif evresindedir: Tamamen doğru nihai cevaplar üretmede zorlanabilir ancak kısmi olarak doğru muhakeme izleri geliştirmesi daha kolaydır. Bu aşamada yalnızca sonuç ödüllerine dayanmak gradyan çatışmalarına yol açabilir; bu nedenle daha güçlü kalibrasyon gerekmektedir. Eğitim ilerledikçe muhakeme ile cevaplar arasındaki uyum gelişir, gradyan çatışmaları azalır ve aşırı gürültü oluşumunu önlemek için daha hafif kalibrasyon yeterli olur. Buna uygun olarak, eğitim ilerledikçe ADO'nun katmasını kademeli olarak azaltan lineer azalan bir ağırlıklendirme şeması kullanıyoruz. Ayrıca, hibrit ödül süreç düzeyindeki sinyalleri sonuç tabanlı ödülü entegre ederek ödül seyrekliği sorununu hafifletir. Bu bileşenler temelinde, ajanik derin araştırma modellerinin performans sınırını ilerletmeyi amaçlayan yeni bir TL çerçevesi olan Atom-Searcher'ı öneriyoruz.

Hem alan içi hem de alan dışı görevleri kapsayan yedi karşılaştırma ölçütünde gerçekleştirildiğimiz deneyler, Atom-Searcher'ın güncel durumun en iyi (SOTA) baz modele kıyasla önemli performans artıları sağladığını ortaya koymaktadır. Buna ek olarak, Atom-Searcher'ın şu avantajlarını vurgulamak için deneyler tasarladık: (1) Atom-Searcher test zamanında hesaplamayı etkili bir şekilde ölçeklendirir. (2) Atomik Düşünceler, Derin Araştırma Modelleri (RRMs) için denetim çapa noktaları sağlar ve derin araştırma görevleri ile RRMs arasında etkili bir köprü kurar. (3) Atom-Searcher daha yorumlanabilir, insan benzeri akıl yürütme kalıpları sergiler. Özette, ana katkılarımız aşağıdaki gibidir:

- İlk olarak Atomic Thought'u tanıtıyoruz; bu, akıl yürütme işlevsel fonksiyonel birimlere ayıran, LLM'lerin daha net ve derinlemesine akıl yürütmesini etkili şekilde yönlendiren yenilikçi bir LLM düşünce paradigmasıdır.
- Atomic Thought temelinde, ince taneli Atomik Düşünce Ödülü tasarlıyor ve ADO'yu sonuç ödülü ile bütünlüğe etkin bir strateji ile sağlıyor. Bu ödül modellemesi, politik optimizasyon sırasında gradyan çatışmaları ve ödül seyrekliğini azaltır.
- Atomik Düşünce paradigması, ADO ve önerilen ödül toplama stratejisi temelinde, ajanik derin araştırmanın performans sınırını ileri taşımayı amaçlayan yenilikçi bir TL çerçevesi olan Atom-Searcher'ı tanıtıyoruz.
- Atom-Searcher'ın, hem alan içi hem de alan dışı görevleri kapsayan yedi karşılaştırma ölçütünde SOTA temel çizgisine kıyasla önemli performans artıları sağladığını gösterdi. Ayrıca, Atom-Searcher'ın çeşitli kayda değer avantajlarını ortaya koyan deneyler tasarladık .

Phase1: Incentivizing LLMs to Generate Atomic Thoughts



Phase2: Reinforcement Learning Guided by Atomic Thought Reward

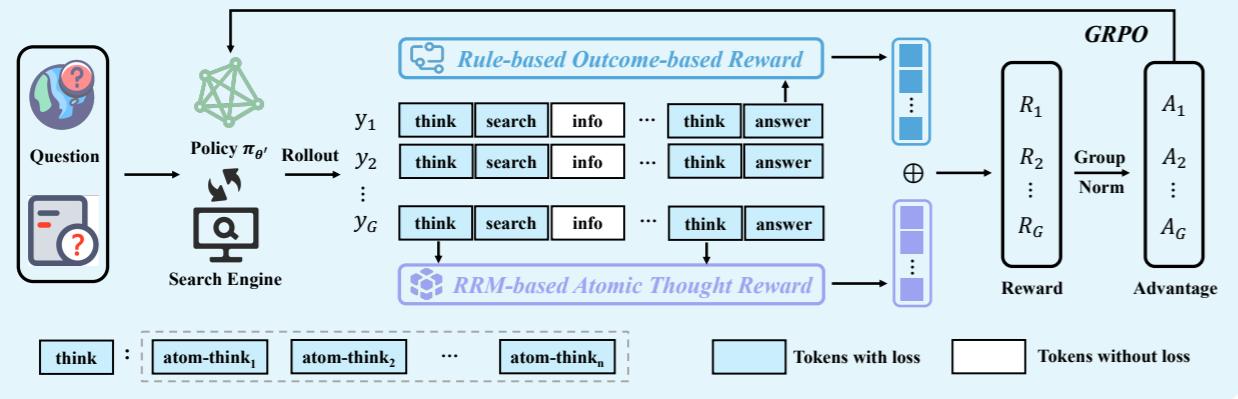


Figure 3 Overview of Atom-Searcher. Within the Atom-Searcher framework, we: 1) construct an atomic thought dataset and apply supervised fine-tuning (SFT) to the policy LLM—serving as the agentic deep research model—to incentivize its capability for generating atomic thoughts; 2) formulate fine-grained atomic thought rewards using a reasoning reward model, aligned with the atomic structure of the reasoning process, and integrate them with existing rule-based outcome rewards to optimize the SFT-initialized policy LLM via reinforcement learning.

2 Atom-Searcher

We propose a novel framework for enhancing agentic deep research models. As illustrated in Figure 3, the framework consists of two phases. In phase1, we construct an atomic thought instruction dataset and perform SFT on the policy model to incentivize its ability to generate atomic thoughts. In phase2, we leverage a Reasoning Reward Model to derive fine-grained rewards based on the generated atomic thoughts, and integrate them with existing rule-based outcome rewards. The resulting hybrid reward is then used to further train the SFT-initialized policy LLM via RL.

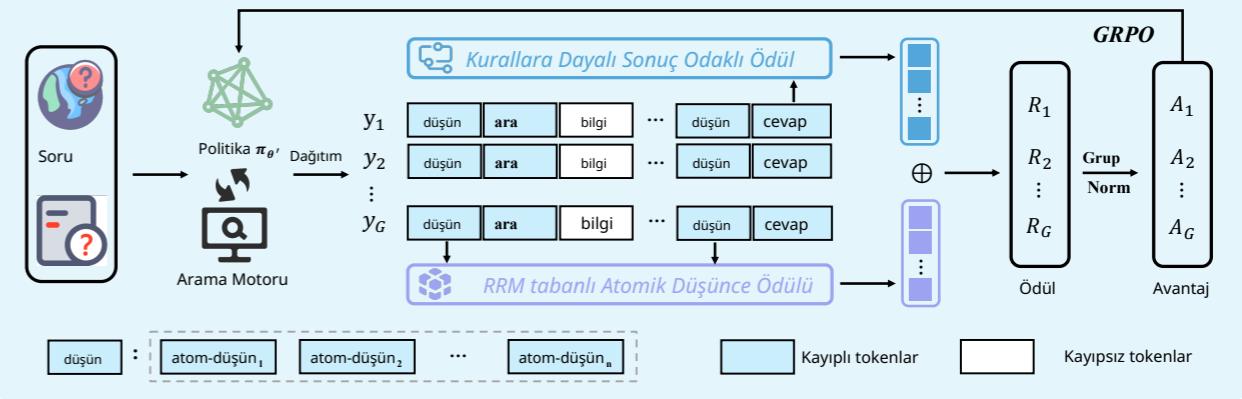
2.1 Preliminary

Atomic Thought. Understanding the fundamental units of thought is critical for simulating intelligence, optimizing decision-making, and extracting actionable knowledge in both cognitive science and computational reasoning Anderson et al. (1997); Ho and Griffiths (2022). Drawing inspiration from philosophical conceptions of thought and the structured decomposition of actions in domains such as football, we propose a principled framework for defining the atomic thought within the reasoning processes of LLMs. A LLM atomic thought is the minimal, functionally coherent unit of reasoning, irreducible in form, yet integral to the model’s reasoning trajectory. The interactions among atomic thoughts collectively form a functionally complete reasoning or behavior process. Take football as an example: when learning the kicking motion of a skilled player, we need to analyze the atomic units that compose this complex behavior, such as step adjustment, leg swing, and point of contact with the ball. Similarly, when shifting to LLMs, assessing the quality of their reasoning requires analyzing the atomic thoughts that compose their thought process. Therefore, in RL settings, designing fine-grained rewards at the atomic thought level can provide valuable intermediate supervision signals for guiding the reasoning trajectory.

Aşama 1: LLM’lerin Atomik Düşünceler Üretmeye Özendirilmesi



Aşama 2: Atomik Düşünce Ödülü ile Yönlendirilen Takviyeli Öğrenme



Şekil 3 Atom-Searcher’ın Genel Görünümü. Atom-Searcher çerçevesinde: 1) atomik düşünce veri seti oluşturuyor ve denetimli ince ayar (SFT) ile ajan olarak görev yapan ajanik derin araştırma modeli niteliğindeki politika LLM’sini atomik düşünceler üretme yeteneğini teşvik edecek şekilde eğitiyoruz; 2) akıl yürütme sürecinin atomik yapısına uygun akıl yürütme ödül modeli kullanarak ince taneli atomik düşünce ödüllerini formüle ediyor ve bunları kurallara dayalı sonuç ödülleri ile birleştirerek SFT ile başlatılmış politika LLM’sini takviyeli öğrenme yoluyla optimize ediyoruz.

2 Atom-Searcher

Ajanik derin araştırma modellerini geliştirmek için yeni bir çerçeve öneriyoruz. Şekil 3’teki gösterildiği gibi, çerçeve iki aşamadan oluşmaktadır. Birinci aşamada, atomik düşünce talimat veri seti oluşturuyor ve politika modeli üzerinde SFT gerçekleştirerek atomik düşünce üretme yeteneğini teşvik ediyoruz. İkinci aşamada, üretilen atomik düşüncelere dayalı ince taneli ödüller elde etmek için bir Akıl Yürütme Ödül Modeli kullanıyor ve bunları mevcut kural tabanlı sonuç ödülleriyle entegre ediyoruz. Ortaya çıkan hibrit ödül, SFT ile başlatılmış politika LLM’ının RL aracılığıyla daha ileri eğitimi için kullanılır.

2.1 Ön Bilgi

Atomik Düşünce. Düşüncenin temel birimlerini anlamak, bilişsel bilim ve hesaplamalı akıl yürütmede zeka simülasyonu, karar verme optimizasyonu ve uygulanabilir bilginin çıkarılması açısından kritik öneme sahiptir Anderson ve ark. (1997); Ho ve Griffiths (2022). Düşünceye ilişkin felsefi kavramlardan ve futbol gibi alanlardaki eylemlerin yapılandırılmış ayrıştırılmasından ilham alarak, LLM’lerin akıl yürütme süreçlerinde atomik düşünceyi tanımlayan prensip temelli bir çerçeve öneriyoruz. Bir LLM atomik düşüncesi, biçim olarak indirgenemeyen ancak modelin akıl yürütme yörungesine ayrılmaz şekilde bağlı olan, minimal ve işlevsel olarak tutarlı akıl yürütme birimidir. Atomik düşünceler arasındaki etkileşimler, işlevsel bakımından tamamlayıcı bir akıl yürütme veya davranış süreci oluşturur. Futbolu örnek alalım: yetenekli bir oyuncunun topa vurma hareketini öğrenirken, bu karmaşık davranışları oluşturan atomik birimler, örneğin adım ayarı, bacak savruluğu ve topa temas noktası analiz edilmelidir. Benzer şekilde, LLM’lere geçildiğinde, akıl yürütmenin kalitesini değerlendirmek için düşünce süreçlerini oluşturan atomik düşünceler analiz edilmelidir. Bu nedenle, Pekşitmeli Öğrenme (RL) ortamlarında, atomik düşünce düzeyinde ince taneli ödüller tasarlamak, akıl yürütme rotasını yönlendirmek için değerli ara denetim sinyalleri sağlayabilir.

In implementation, we encapsulate the LLM’s reasoning process within a $\langle\text{atom-think}_i\rangle$ tag and structure the atomic thoughts as subtags within it, as illustrated in Figure 2. Importantly, the model is not constrained to follow manually defined atomic thoughts. Instead, we incentivize the model to autonomously generate atomic thoughts, enabling it to learn how to decompose reasoning into task-specific atomic thoughts across different scenarios.

Atom-Searcher Trajectory. In an agentic deep research trajectory, the model iteratively performs reasoning and search invocations based on the user question and accumulated observations, as illustrated in Figure 2) *Reasoning*: Following the setup of DeepSeek-R1 Guo et al. (2025), we constrain Atom-Searcher to perform reasoning before taking any other action. Each segment of reasoning is encapsulated between the tags $\langle\text{think}\rangle$ and $\langle/\text{think}\rangle$. Notably, Atom-Searcher further decomposes the reasoning within the $\langle\text{think}\rangle$ tag into a sequence of atomic thoughts, each of which is encapsulated between the tags $\langle\text{atom-think}\rangle$ and $\langle/\text{atom-think}\rangle$ (e.g., $\langle\text{Reflection}\rangle$ and $\langle/\text{Reflection}\rangle$). 2) *Search*: After reasoning, Atom-Searcher may choose to invoke the web search tool by generating a JSON-formatted request with the tool name (`web_search`) and the search queries as arguments. The request is encapsulated between the tags $\langle\text{tool_call}\rangle$ and $\langle/\text{tool_call}\rangle$. 3) *Search Response*: When the system detects the tokens $\langle\text{tool_call}\rangle$ and $\langle/\text{tool_call}\rangle$, a search invocation is triggered. The retrieved results are then wrapped between the tags $\langle\text{tool_response}\rangle$ and $\langle/\text{tool_response}\rangle$ and appended to the current trajectory. 4) *Answer*: Once Atom-Searcher determines that sufficient information has been gathered, it generates the final response enclosed between the tags $\langle\text{answer}\rangle$ and $\langle/\text{answer}\rangle$. This serves as the final answer returned to the user.

Problem Formulation. We model the process of completing the agentic deep research tasks as a finite-horizon Markov Decision Process (MDP), denoted by $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T})$. Given a user instruction I , the agent is required to complete the corresponding task. The state $s \in \mathcal{S}$ is defined as the retrieved content along with the history of previous actions. The action space \mathcal{A} includes three types of actions: 1) a^G (Generate Atomic Thought); 2) a^S (Invoke Search; and) 3) a^A (Answer). At the t -th step, conditioned on the state s_t , the agent takes an action $a_t \in \mathcal{A}$ following the LLM policy π_θ , which can be expressed as:

$$a_t = \pi_\theta(I, s_t) \quad (1)$$

The agent then receives a reward r_t and the state is updated to s_{t+1} . We formalize this process as follows.

$$s_{t+1} = \mathcal{T}(s_t, a_t) \quad (2)$$

$$\mathcal{T}(s_t, a_t) = \begin{cases} \text{concat}(s_t; a_t, d_t) & \text{if } a_t = a_t^S \\ \text{concat}(s_t; a_t) & \text{otherwise} \end{cases} \quad (3)$$

$$r_t = \mathcal{R}(s_t, a_t) \quad (4)$$

where \mathcal{T} and \mathcal{R} denote the deterministic state transition function and deterministic reward function provided by the environment, respectively; $\text{concat}(\cdot)$ denotes the concatenation operation; d_t represents the retrieved external information; and r_t denotes the immediate reward at time step t . In the finite-horizon setting, the trajectory terminates either upon task completion or once the maximum number of interactions is reached. Finally, based on the sampled trajectories, we optimize the policy π_θ using the Group Relative Policy Optimization (GRPO) algorithm Shao et al. (2024).

Uygulamada, LLM’ın akıl yürütme süreci $\langle\text{atom-think}_i\rangle$ etiketi içinde kapsüllenmekte ve atomik düşünceler, Şekil 2’de gösterildiği gibi bu etiketin alt etiketleri olarak yapılandırılmaktadır. Önemle belirtmek gerekir ki, model manuel olarak tanımlanmış atomik düşüncelere uymakla sınırlı değildir. Bunun yerine, modeli atomik düşünceleri otonom olarak üretmeye teşvik ediyoruz; böylece farklı senaryolarda görev özgü atomik düşüncelere akıl yürütmemeyi nasıl parçalayacağını öğrenebilir.

Atom-Searcher Yörünge . Ajanik derin araştırma yörüngeinde, model kullanıcı sorusu ve birikmiş gözlemler temelinde yinelemeli olarak akıl yürütme ve arama çağrıları gerçekleştirir; bu Şekil 2’de gösterilmiştir. 1) Akıl Yürütme : DeepSeek-R1 Guo ve ark. (2025) kurulumunu takip ederek, Atom-Searcher’ı başka herhangi bir EYLEM gerçekleştirmeden önce akıl yürütme yapması konusunda sınırlanıyor. Her akıl yürütme bölümü $\langle\text{think}\rangle$ ve $\langle/\text{think}\rangle$ etiketleri arasında kapsüllenmiştir. Özellikle, Atom-Searcher akıl yürütmemeyi $\langle\text{think}\rangle$ etiketi içinde atomik düşünceler dizisine daha da ayırtır; her atomik düşünce de $\langle\text{atom-think}\rangle$ ve $\langle/\text{atom-think}\rangle$ etiketleri arasında kapsüllenmiştir (örneğin, $\langle\text{Reflection}\rangle$ ve $\langle/\text{Reflection}\rangle$). 2) Arama : Akıl yürütmeden sonra, Atom-Searcher web arama aracını, araç adı (`web_search`) ve arama sorgularını argüman olarak içeren JSON formatında bir istek üretecek çağrımayı tercih edebilir. Bu istek $\langle\text{tool_call}\rangle$ ve $\langle/\text{tool_call}\rangle$ etiketleri arasında kapsüllenmiştir. 3) Arama Yanıtı : Sistem $\langle\text{tool_call}\rangle$ ve $\langle/\text{tool_call}\rangle$ etiketlerini algıladığından bir arama çağrısı tetiklenir. Elde edilen sonuçlar, $\langle\text{tool_response}\rangle$ ve $\langle/\text{tool_response}\rangle$ etiketleri arasına alınarak mevcut yörüngeye eklenir. 4) Yanıt : Atom-Searcher yeterli bilginin toplandığını belirlediğinde, yanıtı $\langle\text{answer}\rangle$ ve $\langle/\text{answer}\rangle$ etiketleri arasında oluşturur. Bu, kullanıcıya döndürülen nihai yanittır.

Problemin Formülasyonu. Ajanik derin araştırma görevlerini tamamlama sürecini, sonlu ufuklu bir Markov Karar Süreci (MDP) olarak modelliyoruz; bu süreç $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{T})$ ile gösterilir. Bir kullanıcı talimatı I verildiğinde, ajan ilgili görevi tamamlamakla yükümlüdür. Durum $s \in \mathcal{S}$ olarak tanımlanır ve önceki eylemlerin geçmişi ile birlikte elde edilen içerikten oluşur. Eylem uzayı A üç tür eylemi içermektedir: 1) a^G (Atomik Düşünce Üret); 2) a^S (Arama Mayı Çağır); ve 3) a^A (Cevap). t -adımında, duruma bağlı olarak s_t , ajan LLM politikası π_θ doğrultusunda bir eylem $a_t \in \mathcal{A}$ seçer ve bu şu şekilde ifade edilebilir:

$$a_t = \pi_\theta(I, s_t) \quad (1)$$

Ardından ajan bir ödül r_t alır ve durum s_{t+1} olarak güncellenir. Bu süreci aşağıdaki gibi formüle ediyoruz.

$$s_{t+1} = \mathcal{T}(s_t, a_t) \quad (2)$$

$$\mathcal{T}(s_t, a_t) = \begin{cases} \text{concat}(s_t; a_t, d_t) & \text{eğer } a_t = a_t^S \text{ ise} \\ \text{concat}(s_t; a_t) & \text{diğer durumlarda} \end{cases} \quad (3)$$

$$r_t = \mathcal{R}(s_t, a_t) \quad (4)$$

burada \mathcal{T} ve \mathcal{R} sırasıyla ortam tarafından sağlanan belirleyici durum geçiş fonksiyonunu ve belirleyici ödül fonksiyonunu ifade eder; $\text{concat}(\cdot)$ birleştirme işlemini belirtir; d_t elde edilen dış bilgiyi temsil eder; ve r_t t zaman adımındaki anlık ödülü ifade eder. Sonlu ufuk ayarında, yörünge ya görev tamamlandığında ya da maksimum etkileşim sayısına ulaşıldığında sonlanır. Son olarak, örneklenen yörüngeye dayanarak politikayı π_θ Group Relative Policy Optimization (GRPO) algoritması Shao ve ark. (2024) kullanarak optimize ederiz.

2.2 Incentivizing LLMs to Generate Atomic Thoughts

To enable LLMs to learn how to reasonably decompose their reasoning processes into atomic thoughts, we construct a high-quality atomic thought dataset D_{atom} consisting of 1,000 annotated examples and perform supervised fine-tuning to impart prior knowledge of atomic thought structures to the model. The details are as follows. The construction of D_{atom} involves two phases: 1) *synthesizing atomic action prompts*: Firstly, we carefully design 10 distinct seed system prompt templates, each containing two atomic thought examples. Each example consists of 3 to 10 common atomic thoughts (e.g., `<plan>`, `<reflection>`, etc.). Secondly, we leverage a powerful teacher model (e.g., Qwen2.5-72B Hui et al. (2024)) to generate approximately 1,000 system prompts based on the seed prompt templates, each containing a distinct combination of atomic thoughts. Finally, we combine each system prompt with different questions and callable search tools (e.g., `web_search`) to obtain 1,000 prompts. 2) *sampling high-quality reasoning trajectories*: Based on these 1,000 prompts, we use Qwen2.5-72B to sample complete reasoning trajectories. To ensure the quality of the generated trajectories, we employ a majority voting strategy during the sampling process. Data samples in D_{atom} follow the reasoning trajectory illustrated in Figure 2. We perform SFT of π_θ on D_{atom} to obtain $\pi_{\theta'}$, which is endowed with prior knowledge of atomic thoughts.

2.3 Reward Modeling

The introduction of atomic thoughts offers a promising perspective for designing fine-grained reward signals to guide agentic deep research models toward developing more intelligent and efficient research strategies. We first construct ATR using a reasoning reward model, and then integrate them with the outcome-level reward through a training-dynamics-aware, linearly decaying aggregation strategy.

Constructing fine-grained atomic thought reward. With the rapid progress of foundation model capabilities and the rise of test-time scaling techniques Snell et al. (2024), Reasoning Reward Models (RRMs) Liu et al. (2025), which leverage large reasoning models (e.g., DeepSeek-R1 Guo et al. (2025)) to generate rewards, have become a promising solution. RRM are particularly effective in settings that require fine-grained supervision, adaptive reasoning, and open-ended tasks without ground truth, making them well aligned with the characteristics of atomic thoughts. Therefore, we use the RRM to score the atomic thoughts generated by the policy model, resulting in the ATR. This process can be formulated as follows:

$$r_{atom}^1, r_{atom}^2, \dots, r_{atom}^n = RRM(I_{score}, y) \quad (5)$$

$$R_{atom} = f(r_{atom}^1, r_{atom}^2, \dots, r_{atom}^n) \quad (6)$$

where, I_{score} denotes the scoring prompt, as illustrated in Figure 6; y refers to the generated trajectory; r_{atom}^i represents the score of the i -th atomic thought, and $f(\cdot)$ denotes the aggregation function that combines individual atomic scores. The choice of $f(\cdot)$ is not fixed—it can be a simple average or a more sophisticated weighting strategy. R_{atom} denotes the ATR of trajectory y .

A Dynamic, Curriculum-Based Approach to Reward Aggregation. A key limitation of outcome-based reward is their coarse credit assignment: it attribute the correctness of intermediate reasoning solely to the final answer, often rewarding or penalizing steps regardless of their actual contribution. This misalignment introduces gradient conflicts during optimization. To address this, we aggregate ATR with the outcome reward, using ATR as an auxiliary signal to calibrate the final reward, thereby mitigating gradient conflicts and improving test-time performance. However, using a static weighting

2.2 LLM’lerin Atomik Düşünceler Üretmesini Özendirmek

LLM’lerin akıl yürütme süreçlerini mantıklı bir şekilde atomik düşüncelere bölmeyi öğrenmelerini sağlamak için, 1.000 etiketlenmiş örnekten oluşan yüksek kaliteli bir atomik düşünce veri seti D_{atom} oluşturuyor ve modele atomik düşünce yapılarına dair ön bilgi kazandırmak amacıyla denetimli ince ayar gerçekleştiriyor. Detaylar aşağıdaki gibidir. D_{atom} ’un oluşumu iki aşamadan oluşur: 1) atomik eylem istemlerinin üretilmesi: Öncelikle, her biri iki atomik düşünce içeren 10 farklı tohum sistemi istemi şablonunu titizlikle tasarlıyor. Her örnek, 3 ila 10 yaygın atomik düşündeden (örneğin, `<plan>`, `<refleksiyon>` vb.) oluşmaktadır. İkinci olarak, güçlü bir öğretmen modeli (örneğin, Qwen2.5-72B Hui et al. (2024)) kullanarak tohum istemi şablonlarına dayalı yaklaşık 1.000 sistem istemi üretiyor; her biri farklı atomik düşünce kombinasyonları içermektedir. Son olarak, her sistem istemini farklı sorular ve çağrılabilecek arama araçlarıyla (örneğin, `web_search`) birleştirerek 1.000 istem elde ediyoruz. 2) yüksek kaliteli akıl yürütme yörüngelarının örneklenmesi: Bu 1.000 istem temelinde, Qwen2.5-72B kullanılarak tam akıl yürütme yörünge örneklenmektedir. Üretilen yörünge kalitesini garanti altına almak için örnekleme sürecinde çoğuluk oylaması stratejisi kullanıyoruz. D_{atom} ’da veri örnekleri, Şekil’de gösterilen akıl yürütme yörüngeini takip etmektedir. 2. π_θ üzerinde D_{atom} üzerinde SFT gerçekleştirerek π_θ' elde ediyoruz; bu, atomik düşüncelerin ön bilgisini içermektedir.

2.3 Ödül Modellemesi

Atomik düşüncelerin tanıtılması, ajanık derin araştırma modellerini daha zeki ve etkin araştırma stratejileri geliştirmeye yönlendirmek için ince taneli ödül sinyalleri tasarlamada umut vadeden bir perspektif sunmaktadır. Öncelikle akıl yürütme ödül modeli kullanarak ADO’yu oluşturuyor ve ardından bunları eğitim dinamiklerine duyarlı, doğrusal azalan bir birleştirme stratejisiyle sonuç seviyesi ödülle entegre ediyoruz.

İnce taneli atomik düşünce ödülünün oluşturulması. Temel model yeteneklerindeki hızlı ilerleme ve test zamanı ölçekleme tekniklerinin yükselişi (Snell ve ark., 2024), büyük akıl yürütme modellerini (örneğin DeepSeek-R1, Guo ve ark.

(2025)) ödül üretmek için kullanılan Akıl Yürütme Ödül Modelleri (RRMs) (Liu ve ark., 2025) umut vadeden bir çözüm haline gelmiştir. RRM, ince taneli denetim, uyarlanabilir akıl yürütme ve gerçek doğrusu olmayan açık uçlu görevlerin gerekliliği söz konusu olduğunda özellikle etkilidir; bu da onları atomik düşüncelerin özellikleriyle uyumlu kılar. Bu nedenle, politika modelinin ürettiği atomik düşünceleri puanlamak için RRM’yi kullanıyoruz ve sonuç olarak ADO ortaya çıkmaktadır. Bu süreç aşağıdaki şekilde formüle edilebilir:

$$r_{atom}^1, r_{atom}^2, \dots, r_{atom}^n = RRM(I_{score}, y) \quad (5)$$

$$R_{atom} = f(r_{atom}^1, r_{atom}^2, \dots, r_{atom}^n) \quad (6)$$

burada, I_{score} puanlama promptunu ifade eder; Şekil 6’da gösterildiği gibidir; y üretilen yörüngeyi ifade eder; r_{atom}^i , i -inci atomik düşüncenin skorunu temsil eder ve $f(\cdot)$ bireysel atomik skorları birleştiren toplama fonksiyonudur. $f(\cdot)$ seçimi sabit değildir; basit bir ortalama veya daha karmaşık bir ağırlıklanırma stratejisi olabilir. R_{atom} yörünge y için ADO’yu ifade eder.

Ödül Toplamaya Dinamik, Müfredat Tabanlı Bir Yaklaşım. Sonuca dayalı ödüllerin temel sınırlaması, kaba kredi dağılımıdır: ara çıkarımların doğruluğunu yalnızca son yanıtla atfeder ve genellikle adımları gerçek katkılarına bakmaksızın ödüllendirir ya da cezalandırır.

Bu uyumsuzluk, optimizasyon sırasında gradyan tartışmalarına yol açar. Bunu gidermek için, ADO’yu sonuç ödülü ile birleştiriyoruz; ADO’yu nihai ödülü kalibre etmek amacıyla yardımcı bir sinyal olarak kullanarak gradyan tartışmalarını azaltıyor ve test zamanı performansını iyileştiriyoruz. Ancak, statik bir ağırlık

coefficient for reward aggregation fails to align with training dynamics. Specifically, early in training, the model—still limited in its deep research capability—struggles to generate fully correct answers but is more likely to explore useful atomic thoughts that contribute toward a correct solution. If training relies solely on outcome-based rewards at this stage, these beneficial atomic thoughts may be unjustly penalized due to the incorrect final answer; conversely, harmful atomic thoughts may also be mistakenly reinforced, resulting in severe gradient conflict and necessitating strong calibration from ATR. As training progresses and the model’s deep research ability improves, its reasoning trajectories become increasingly aligned with correct answers. Consequently, gradient conflicts diminish, and excessive calibration from ATR may introduce unnecessary noise, potentially harming final accuracy. To accommodate this, we adopt a training-dynamics-aware weighting scheme that linearly reduces the contribution of ATR as training progresses, formulated mathematically as follows:

$$\alpha = 0.5 \times \left(1 - \frac{T}{T_{MAX}}\right) \quad (7)$$

$$R = \begin{cases} \alpha R_{atom} + (1 - \alpha) R_{f1} & \text{if format is correct} \\ -1 & \text{if format is incorrect} \end{cases} \quad (8)$$

$$R_{f1} = \frac{2 \times IN}{PN + RN} \quad (9)$$

where, T denotes the current training step, and T_{MAX} denotes the maximum number of training steps. R denotes the final reward used for RL training, and R_{f1} represents the outcome-based reward computed from the F1 score. The coefficient $\alpha \in [0, 1]$ is a hyperparameter that balances the influence of ATR and the outcome reward during training. PN denotes the word count of the predicted answer, RN denotes the word count of the reference answer, and IN denotes the word count of their intersection.

2.4 RL Training Framework

Policy Optimization. In this work, we adopt the GRPO algorithm Shao et al. (2024) to optimize the SFT policy $\pi_{\theta'}$ using the hybrid reward R that aggregates final answer correctness and reasoning quality. GRPO improves the current policy $\pi_{\theta'}$ by leveraging a reference policy $\pi_{\theta'_{ref}}$ and a set of rollouts generated by a previous policy $\pi_{\theta'_{old}}$. The objective is extended and formulated as follows:

$$r_1, r_2, \dots, r_G = R(y_1, y_2, \dots, y_G) \quad (10)$$

$$A_i = \frac{r_i - \text{mean}(r_1, r_2, \dots, r_G)}{\text{std}(r_1, r_2, \dots, r_G)} \quad (11)$$

$$\begin{aligned} \mathcal{J}_{GRPO}(\theta') = \mathbb{E}_{x \sim \mathcal{D}, \{y_i\}_{i=1}^G \sim \pi_{\theta'_{old}}(\cdot|x)} \left[\frac{1}{G} \sum_{i=1}^G \min \left(\frac{\pi_{\theta'}(y_i|x)}{\pi_{\theta'_{old}}(y_i|x)} A_i, \right. \right. \\ \left. \left. \text{clip} \left(\frac{\pi_{\theta'}(y_i|x)}{\pi_{\theta'_{old}}(y_i|x)}, 1 - \epsilon, 1 + \epsilon \right) A_i \right) - \beta \mathbb{D}_{KL}(\pi_{\theta'} \parallel \pi_{\theta'_{ref}}) \right] \quad (12) \end{aligned}$$

where x denotes an input sampled from the experience distribution D , y_i represents a trajectory generated by $\pi_{\theta'_{old}}$, G is the number of trajectories sampled per training example, r_i is the reward of y_i , A_i is the advantage of y_i , \mathbb{D}_{KL} denotes the unbiased estimate of KL divergence Shao et al. (2024), and β is a tunable hyperparameter. In addition, to mitigate entropy collapse during policy optimization, we adopt a sliding-window-based entropy regulation mechanism, as detailed in Appendix A.1.

kat sayısının ödül toplamasında kullanılması eğitim dinamikleriyle uyumlu değildir. Özellikle, eğitimin erken aşamalarında model — henüz derin araştırma yeteneği sınırlı — tamamen doğru cevaplar üretmekte zorlanır ancak doğru çözüme katkıda bulunan faydalı atomik düşünceleri keşfetme olasılığı daha yüksektir. Eğitim bu aşamada yalnızca sonuç odaklı ödülüle dayanırsa, bu faydalı atomik düşünceler yanlış nihai cevap sebebiyle haksız yere cezalandırılabilir; aksine, zararlı atomik düşünceler de yanlışlıkla takviye edilebilir, bu durum ciddi gradyan tartışmalarına neden olur ve ADO’dan güçlü kalibrasyon gerektirir. Eğitim ilerledikçe ve modelin derin araştırma yeteneği gelişikçe akıl yürütme yolları giderek doğru cevaplarla uyumlu hale gelir. Sonuç olarak gradyan tartışmaları azalır ve ADO’dan aşırı kalibrasyon gereksiz gürültü yaratır, bu da nihai doğruluğa zarar verebilir. Bunu karşılamak için, eğitim ilerledikçe ATR’ın katkısını doğrusal olarak azaltan, eğitim dinamiklerine duyarlı bir ağırlıklandırma şeması benimsemektedir; bu, matematiksel olarak aşağıdaki gibi formüle edilmiştir

$$\alpha = 0.5 \times \left(1 - \frac{T}{T_{MAX}}\right) \quad (7)$$

$$R = \begin{cases} \alpha R_{atom} + (1 - \alpha) R_{f1} & \text{eğer format doğru ise} \\ -1 & \text{Eğer format yanlış ise} \end{cases} \quad (8)$$

$$R_{f1} = \frac{2 \times IN}{PN + RN} \quad (9)$$

Burada, T mevcut eğitim adımı, T_{MAX} ise maksimum eğitim adımları sayısını ifade eder. R RL eğitimi için kullanılan nihai ödül, R_{f1} ise F1 skorundan hesaplanan sonuç-temelli ödülü belirtir. $\alpha \in [0, 1]$ koefisiyenti, eğitim sırasında ATR ile sonuç ödülünün etkisini dengeleyen bir hiperparametredir. $P N$ tahmin edilen cevabın kelime sayısını, RN referans cevabın kelime sayısını ve IN bunların kesişimindeki kelime sayısını gösterir.

2.4 RL Eğitim Çerçeve

Politik optimizasyon . Bu çalışmada, Shao ve ark. (2024) tarafından önerilen GRPO algoritmasını benimseyerek, son yanıt doğruluğu ve akıl yürütme kalitesini birleştiren karma ödül R ile SFT politikası $\pi_{\theta'}$ optimize edilmektedir. GRPO, bir referans politika olan $\pi_{\theta'}$ ve $\pi_{\theta'_{eski}}$ bir dizi önceki bir politika tarafından üretilen rolloutları $\pi_{\theta'_{eski}}$. Amaç genişletilerek aşağıdaki şekilde formüle edilmiştir:

$$r_1, r_2, \dots, r_G = R(y_1, y_2, \dots, y_G) \quad (10)$$

$$A_i = \frac{r_i - \text{mean}(r_1, r_2, \dots, r_G)}{\text{std}(r_1, r_2, \dots, r_G)} \quad (11)$$

$$\begin{aligned} \mathcal{J}_{GRPO}(\theta') = \mathbb{E}_{x \sim \mathcal{D}, \{y_i\}_{i=1}^G \sim \pi_{\theta'_{old}}(\cdot|x)} \left[\frac{1}{G} \sum_{i=1}^G \min \left(\frac{\pi_{\theta'}(y_i|x)}{\pi_{\theta'_{eski}}(y_i|x)} A_i, \right. \right. \\ \left. \left. \text{clip} \left(\frac{\pi_{\theta'}(y_i|x)}{\pi_{\theta'_{eski}}(y_i|x)}, 1 - \epsilon, 1 + \epsilon \right) A_i \right) - \beta \mathbb{D}_{KL}(\pi_{\theta'} \parallel \pi_{\theta'_{ref}}) \right] \quad (12) \end{aligned}$$

Burada x , deneyim dağılımı D ’den örneklenen bir girdiyi, y_i ise $\pi_{\theta'}$ tarafından oluşturulan bir yörüngeyi ifade eder. $\pi_{\theta'}$, G her eğitim örneği için örneklenen yörünge sayısını, r_i y_i ’nin ödülünü, A_i y_i ’nin avantajını, D \mathbb{D}_{KL} KL ayırmalarının tarafsız tahminini (Shao ve ark., 2024) gösterir ve β ayarlanabilir bir hiperparametredir. Ayrıca, politik optimizasyon sırasında entropi çöküşünü azaltmak için, Ek A.1’de ayrıntıları verilen kaydırma pencere tabanlı entropi düzenleme mekanizmasını benimsiyoruz.

Loss Masking. In the original GRPO framework, loss is computed over all tokens in the trajectory. However, in Atom-Searcher, trajectories include retrieval results that are externally fetched by the environment rather than generated by the policy itself. To prevent biasing the policy update toward non-trainable, static content, we apply loss masking to exclude these retrieved segments from the optimization objective. Specifically, in the computation of Equation 12, only tokens corresponding to the model’s reasoning (i.e., text-based thinking) and search queries are included, while tokens originating from retrieval results are masked out.

3 Experiments

3.1 Implementation Details

We use Qwen2.5-7B-Instruct [Qwen et al. \(2025\)](#) as the backbone models. The training is conducted using the verl framework [Sheng et al. \(2024\)](#). At each training step, we sample 32 prompts and generate 16 rollouts per prompt. Each rollout consists of up to 10 tool calls, followed by a final answer step. The training is performed with a mini-batch size of 512, meaning that one rollout stage corresponds to a single backpropagation step. By default, we use Qwen3-30B-A3B [Yang et al. \(2025a\)](#) as the reasoning reward model in Atom-Searcher.

3.2 Benchmarks

To comprehensively assess model performance in both in-domain (ID) and out-of-domain (OOD) scenarios, we construct a diverse evaluation benchmark spanning a wide range of open-domain QA tasks. For ID evaluation, we include the development sets of NQ [Kwiatkowski et al. \(2019\)](#), TQ [Joshi et al. \(2017\)](#), HotpotQA [Yang et al. \(2018\)](#), and 2Wiki [Ho et al. \(2020\)](#). To evaluate OOD generalization, we incorporate three datasets that differ substantially in question format and information distribution: MuSiQue [Trivedi et al. \(2022\)](#), Bamboogle [Press et al. \(2022\)](#), and PopQA [Mallen et al. \(2022\)](#). These datasets are chosen to challenge the model’s ability to generalize beyond its training distribution.

To ensure fair comparison and balanced evaluation, we randomly sample 512 examples from the development sets of NQ, TQ, HotpotQA, 2Wiki, MuSiQue, and PopQA, along with all 125 examples from the Bamboogle development set. This evaluation setup enables a rigorous assessment of model robustness across diverse topics and reasoning demands.

3.3 Baselines

To evaluate the effectiveness of Atom-Searcher, we compare it against the following baseline methods:

- **CoT**: This baseline performs Chain-of-Thought (CoT) reasoning to generate answers without access to any external reference context.
- **Cot+RAG**: This baseline integrates CoT reasoning with retrieved reference context to guide the answer generation process.
- **Search-01**: This baseline performs multi-step reasoning by generating search queries or intermediate answers. For each query, the model receives only a snippet retrieved by a retriever, rather than the full document content.

Kayıp Maskesi. Orijinal GRPO çerçevesinde, kayıp yörüngegedeki tüm tokenlar üzerinden hesaplanır. Ancak, Atom-Searcher’da yörünge, politikanın kendisi tarafından değil, çevre tarafından harici olarak getirilen geri çağrıma sonuçlarını içerir. Politika güncellemesini eğitilemeyen, statik içeriğe karşı önyargılandırmamak için, bu getirilen segmentleri optimizasyon hedefinden çıkarmak üzere kayıp maskesi uygularız. Özellikle, Denklem 12’nin hesaplanması yalnızca modelin muhakemesi (yani metin tabanlı düşünme) ve arama sorgularına ait tokenlar dahil edilir; geri çağrıma sonuçlarından gelen tokenlar ise maskelenir.

3 Deneyler

3.1 Uygulama Detayları

Qwen et al. (2025) tarafından geliştirilen Qwen2.5-7B-Instruct temel modeller olarak kullanılmaktadır. Eğitim, Sheng et al. (2024) tarafından sunulan verl çerçevesi kullanılarak gerçekleştirilmiştir. Her eğitim adımımda 32 istek örneklenir ve her istek için 16 çalışma yürütülür. Her çalışma, en fazla 10 araç çağrılarından oluşur ve bunu son bir cevap adımı izler. Eğitim, 512 mini-batch büyülüğu ile gerçekleştirilir; bu da bir çalışma aşamasının tek bir geri yayılım adımına karşılık geldiği anlamına gelir. Varsayılan olarak, Atom-Searcher’da akıl yürütme ödül modeli olarak Qwen3-30B-A3B Yang ve ark. (2025a) kullanılır.

3.2 Karşılaştırma Ölçütleri

Model performansını hem alan-içi (ID) hem de alan-dışı (OOD) senaryolarda kapsamlı şekilde değerlendirmek amacıyla, geniş bir açık alan SSS görev yelpazesini içeren çeşitli bir değerlendirme karşılaştırma ölçüyü oluşturduk. ID değerlendirme için NQ Kwiatkowski ve ark. (2019), TQ Joshi ve ark. (2017), HotpotQA Yang ve ark. (2018) ile 2Wiki Ho ve ark. (2020) geliştirme setlerini dahil ettik. OOD genellemesini değerlendirmek için, soru formatı ve bilgi dağılımı açısından önemli ölçüde farklılık gösteren üç veri setini dahil ediyoruz: MuSiQue (Trivedi et al., 2022), Bamboogle (Press et al., 2022) ve PopQA (Mallen et al., 2022). Bu veri setleri, modelin eğitim dağılımının ötesinde genelleme yeteneğini sınamak amacıyla seçilmişdir.

Adil karşılaştırma ve dengeli değerlendirme sağlamak için, NQ, TQ, HotpotQA, 2Wiki, MuSiQue ve PopQA gelişim setlerinden rastgele seçilen 512 örnek ile Bamboogle gelişim setindeki tüm 125 örnek kullanılmıştır. Bu değerlendirme yapısı, modelin farklı konular ve akıl yürütme gereksinimleri karşısındaki dayanıklılığını dikkatle ölçmeye olanak tanır.

3.3 Temel Yöntemler

Atom-Searcher’ın etkinliğini değerlendirmek için aşağıdaki temel yöntemlerle karşılaştırma yapılmıştır:

- **CoT** : Bu temel yöntem, herhangi bir dış referans bağlamına erişmeden cevapları Zincir Düşünce (Chain-of-Thought, CoT) akıl yürütmesi ile üretir.
- **Cot+RAG** : Bu temel model, CoT muhakemesini, yanıt oluşturma sürecini yönlendirmek için alınan referans bağlamla bütünlleştirir.
- **Search-01** : Bu temel model, arama sorguları veya ara yanıtlar üreterek çok adımlı muhakeme gerçekleştirir. Her sorgu için model, tam belge içeriği yerine yalnızca bir arama motoru tarafından bulunan bir parçayı alır.

- **Search-o1-Web**: Unlike Search-o1, this setting allows the model to interact with the open web by issuing real-time queries through APIs and browsing webpages via URLs. This capability supports more dynamic and comprehensive information acquisition, laying the groundwork for deep research.
- **Search-r1**: This is a reinforcement learning approach for question answering that utilizes a retriever to search Wikipedia during both training and inference. It includes two variants—Search-r1-base and Search-r1-instruct—which are initialized from either the base model or the instruct-tuned model, respectively.
- **R1-Seaearcher**: This is a two-stage, outcome-driven RL baseline that equips LLMs with autonomous search: the model learns to invoke external search tools and incorporate retrieved evidence on the fly to improve reasoning.
- **DeepResearcher**: This is an end-to-end trained LLM agent for deep research tasks, leveraging reinforcement learning in real-world web environments. It interacts with the open web via real-time search and browsing, enabling dynamic information acquisition.

Table 1 Performance comparison of Atom-Searcher and baselines on in-domain and out-of-domain benchmarks, evaluated by F1 score; the best and second-best results are marked in **bold** and underlined, respectively.

Type	Method	In-domain				Out-of-domain		
		NQ	TQ	HotpotQA	2Wiki	Musique	Bamboogle	PopQA
<i>Prompt Based</i>	CoT	19.8	45.6	24.4	26.4	8.5	22.1	17.0
	CoT+RAG	42.0	68.9	37.1	24.4	10.0	25.4	46.9
	Search-o1	34.5	52.6	31.6	28.6	16.8	35.8	36.9
	Search-o1-Web	32.4	58.9	33.0	30.9	14.7	46.6	38.3
<i>Training Based</i>	Search-r1-base	45.4	71.9	<u>55.9</u>	44.6	26.7	56.5	43.2
	Search-r1-Instruct	33.1	44.7	45.7	43.4	26.5	45.0	43.0
	R1-Searcher	35.4	73.1	44.8	59.4	22.8	64.8	42.7
	DeepResearcher	39.6	<u>78.4</u>	52.8	<u>59.7</u>	<u>27.1</u>	71.0	48.5
	Atom-Searcher	<u>44.0</u>	81.8	57.3	66.9	27.6	<u>70.7</u>	50.3

Table 2 Ablation study of Atom-Searcher on seven QA benchmarks. We analyze the contribution of each component (RRM and Atom Thought). The **bold** indicates the best performance, and underline indicates the second-best performance.

Method	In-domain				Out-of-domain		
	NQ	TQ	HotpotQA	2Wiki	Musique	Bamboogle	PopQA
Base	39.6	<u>78.4</u>	52.8	59.7	<u>27.1</u>	71.0	48.5
+ RRM	<u>40.1</u>	78.2	<u>53.5</u>	<u>60.0</u>	25.7	70.5	<u>48.8</u>
Atom-Searcher	44.0	81.8	57.3	66.9	27.6	<u>70.7</u>	50.3

3.4 Main Result

Our main result, presented in Table 1, show that Atom-Searcher achieves significant performance gains over both prompt-based and training-based baselines on in-domain and out-of-domain benchmarks.

- **Search-o1-Web** : Search-o1'den farklı olarak, bu ayar modelin API'ler aracılığıyla gerçek zamanlı sorgular yaparak ve URL'ler üzerinden web sayfalarını gezerek açık web ile etkileşimde bulunmasına olanak tanır. Bu yetenek, daha dinamik ve kapsamlı bilgi edinmeyi destekleyerek derin araştırma için temel oluşturur.
- **Search-r1** : Bu, takviyeli öğrenme tabanlı bir soru yanıtımıdır ve eğitim ile çıkarmı sırasında Wikipedia'yı aramak için bir arama motoru kullanır. İki çeşidi vardır—Search-r1-base ve Search-r1-instruct—bunlar sırasıyla temel modelden veya instruct-tuned modelden başlatılır.
- **R1-Seaearcher** : Bu, LLM'leri otonom arama ile donatan, sonuç odaklı iki aşamalı bir takviyeli öğrenme temel modelidir: Model, muhakemeyi geliştirmek için harici arama araçlarını çağrırmayı ve elde edilen kanıtları anlık olarak dahil etmeyi öğrenir.
- **DerinAraştırmacı** : Derin araştırma görevleri için uçtan uca eğitilmiş bir LLM ajanıdır ve gerçek dünya web ortamlarında takviyeli öğrenmeyi kullanmaktadır. Gerçek zamanlı arama ve gezinme yoluyla açık web ile etkileşim kurarak dinamik bilgi edinimini sağlar.

Tablo 1 Atom-Searcher ve temel yöntemlerin alan-içi ve alan-dışı karşılaştırma ölçütlerindeki performans karşılaştırması , F1 skoru ile değerlendirilmiştir; En iyi ve ikinci en iyi sonuçlar sırasıyla kalın ve altı çizili olarak işaretlenmiştir.

TürYöntem	Alan-içi				Alan-dışı		
	NQ	TQ	HotpotQA	2Wiki	Musique	Bamboogle	PopQA
<i>İstem Tabanlı</i>	CoT	19.8	45.6	24.4	26.4	8.5	22.1
	CoT+RAG	42.0	68.9	37.1	24.4	10.0	25.4
	Search-o1	34.5	52.6	31.6	28.6	16.8	35.8
	Search-o1-Web	32.4	58.9	33.0	30.9	14.7	46.6
<i>Eğitim Tabanlı</i>	Search-r1-base	45.4	71.9	<u>55.9</u>	44.6	26.7	56.5
	Search-r1-Instruct	33.1	44.7	45.7	43.4	26.5	45.0
	R1-Searcher	35.4	73.1	44.8	59.4	22.8	64.8
	DerinAraştırmacı	39.6	<u>78.4</u>	52.8	<u>59.7</u>	<u>27.1</u>	71.0
	Atom-Searcher	<u>44.0</u>	81.8	57.3	66.9	<u>27.6</u>	<u>70.7</u>

Tablo 2 Atom-Searcher'in yedi SSS karşılaştırma ölçütü üzerindeki ablation çalışması. Her bileşenin katkısını analiz ediyoruz (RRM ve Atom Thought). Kalın yazı en iyi performansı, altı çizili ise ikinci en iyi performansı göstermektedir.

Yöntem	Alan-içi				Alan-dışı		
	NQ	TQ	HotpotQA	2Wiki	Musique	Bamboogle	PopQA
Temel	39.6	<u>78.4</u>	52.8	59.7	<u>27.1</u>	71.0	48.5
+ RRM	<u>40.1</u>	78.2	<u>53.5</u>	<u>60.0</u>	25.7	70.5	<u>48.8</u>
Atom-Searcher	44.0	81.8	57.3	66.9	27.6	<u>70.7</u>	50.3

3.4 Ana Sonuç

Tablo 1'de sunulan ana sonucumuz, Atom-Searcher'in alan içi ve alan dışı karşılaştırma ölçütlerinde hem prompt-temelli hem de eğitim-temelli temel çizgilere kıyasla önemli performans kazanımları elde ettiğini göstermektedir.

3.4.1 Atom-Searcher outperforms baselines on in-domain benchmarks

In the in-domain results, Atom-Searcher achieved the best performance on the TQ, HotpotQA and 2Wiki benchmarks, showing significant improvements over the second-best results, with increases of 4.3%, 2.5% and 12.1%, respectively. On average, Atom-Searcher outperformed the SOTA baseline (DeepResearcher) by 8.5% across the four in-domain benchmarks. Notably, while Search-r1-base achieved optimal performance on NQ, it was trained and evaluated using a local RAG system with direct access to the relevant Wikipedia corpus. In contrast, Atom-Searcher navigates the entire Internet to find relevant information, presenting a more realistic and challenging scenario, despite both models ultimately sourcing answers from Wikipedia.

3.4.2 Atom-Searcher demonstrates optimal out-of-domain generalization

In the out-of-domain results, Atom-Searcher achieved the best performance on the Musique and PopQA benchmarks, improving over the second-best performance by 1.8% and 3.7%, respectively. On Bamboogle, it achieved second-best performance, but was only 0.4% lower than the optimal result. On average, Atom-Searcher outperformed the SOTA baseline (DeepResearcher) by 2.5% across the three out-of-domain benchmarks. This demonstrates that Atom-Searcher effectively generalizes the skills learned during RL to unseen scenarios.

3.5 Atom-Searcher Effectively Scales Computation at Test Time

To analyze whether Atom-Searcher can effectively scale computation at test time, we compared the average number of tokens generated during the testing phase between Atom-Searcher and the SOTA baseline DeepResearcher. As

shown in Table 3, Atom-Searcher generates 3.2 times more tokens in the average response length (**avg.# response tokens**) compared to DeepResearcher. In terms of the average length of a single think process within the response (**avg.# think tokens**), Atom-Searcher generates 2.6 times more tokens. Additionally, Atom-Searcher performs 1.24 times more tool calls per response (**avg.# tool calls**) than DeepResearcher. This demonstrates that the Atom-Searcher architecture effectively achieves Test-Time Scaling without the introduction of additional incentives for generating more tokens, highlighting its stronger exploration and discovery capabilities when handling complex and challenging deep research tasks.

3.6 Ablation Study

We conduct an ablation study to evaluate the impact of the Atomic Thought and the fine-grained rewards generated by RRM on Atom-Searcher. To assess their contributions, we compare **Atom-Searcher** with two alternative frameworks: (1) **Base** refers to the DeepResearcher [Zheng et al. \(2025\)](#) setting, indicating Atom-Searcher w/o Atomic Thought & fine-grained rewards generated by RRM. (2) **+RRM** refers to the incorporation of fine-grained rewards generated by RRM (with the same implementation details as Atom-Searcher) on top of the Base setting, indicating Atom-Searcher w/o Atomic Thought. As shown in Table 2, the results across seven benchmarks, including both in-domain and out-of-domain, indicate that **+RRM** does not yield a significant performance improvement over **Base**. This suggests that directly using RRM for fine-grained supervision provides minimal benefits. However, **Atom-Searcher** significantly outperforms **+RRM**, achieving an average performance improvement of 6.1% across four in-domain benchmarks and 2.5% across

3.4.1 Atom-Searcher alan içi kıyaslamalarda temel çizgilerin üzerinde performans göstermektedir

Alan içi sonuçlarda, Atom-Searcher TQ, HotpotQA ve 2Wiki karşılaştırma ölçütlerinde en iyi performansı elde etmiş ve ikinci en iyi sonuçlara sırasıyla %4,3, %2,5 ve %12,1 oranlarında anlamlı iyileşmeler kaydetmiştir. Ortalama olarak, Atom-Searcher dört alan içi karşılaştırma ölçütünde SOTA temel çizgisi (DerinAraştırmacı) üzerinde %8,5'lük bir üstünlük sağlamıştır. Özellikle, Search-r1-base NQ üzerinde optimal performans elde etmiş olmakla birlikte, ilgili Wikipedia korpusuna doğrudan erişimi olan yerel bir RAG sistemi kullanılarak eğitilmiş ve değerlendirilmiştir. Buna karşılık, Atom-Searcher tüm İnternet'i tarayarak ilgili bilgiyi bulmakta ve her iki model de nihai olarak cevaplarını Wikipedia'dan sağlasa da, daha gercekçi ve zorlu bir senaryoyu temsil etmektedir.

3.4.2 Atom-Searcher optimal alan dışı genelleme yeteneği göstermektedir

Alan dışı sonuçlarda, Atom-Searcher Musique ve PopQA karşılaştırma ölçütlerinde en iyi performansı sergileyerek, ikinci en iyi performansa sırasıyla %1,8 ve %3,7 oranında iyileşme sağlamıştır. Bamboogle üzerinde ikinci en iyi performansı elde etmiş ancak optimal sonuca yalnızca %0,4 oranında daha düşük kalmıştır. Genel olarak, Atom-Searcher üç alan dışı karşılaştırma ölçütünde SOTA temel çizgisi olan DerinAraştırmacı'dan %2,5 daha iyi performans göstermiştir. Bu durum, Atom-Searcher'in pekiştirmeli öğrenme sırasında edinen becerileri henüz karşılaşılmamış senaryolara etkili şekilde genelleyebildiğini ortaya koymaktadır.

3.5 Atom-Searcher Test Zamanında Hesaplama Etkin Şekilde Ölçeklendirir

Atom-Searcher'in test zamanında hesaplama etkin şekilde ölçeklendirdip ölçeklendiremeyeceğini analiz etmek için, test aşamasında üretilen token sayısının ortalamasını Atom-Searcher ile SOTA temel çizgisi DerinAraştırmacı arasında karşılaştırıldı.

Tablo 3 Atom-Searcher ile DerinAraştırmacı Arasında Test Zamanında Token Üretim İstatistikleri			
Yöntem	ortalama # cevap tokenleri	ortalama # düşünme tokenleri	ortalama # araç çağrıları
DerinAraştırmacı	176	55	2.13
Atom-Searcher	565	143	2.65

Tablo 3'te gösterildiği üzere, Atom-Searcher ortalama cevap uzunluğu (ortalama # cevap tokenleri) açısından DerinAraştırmacı'ya kıyasla 3,2 kat daha fazla token üretmektedir. Cevap içerisindeki tek bir düşünme sürecinin ortalama uzunluğu (ortalama # düşünme tokenleri) bakımından Atom-Searcher 2,6 kat daha fazla token üretmektedir. Ek olarak, Atom-Searcher yanıt başına DeepResearcher'dan 1,24 kat daha fazla araç çağrıları gerçekleştirir (avg.# araç çağrıları). Bu, Atom-Searcher mimarisinin daha fazla token üretimi için ek teşvikler olmadan Test-Zamanı Ölçeklendirmesini etkin biçimde sağladığını ortaya koymakta ve karmaşık, zorlu derin araştırma görevlerinde daha güçlü keşif ve bulma yeteneklerini vurgulamaktadır.

3.6 Yalıtım Çalışması

Atom-Searcher üzerinde Atomik Düşünce ve RRM tarafından üretilen ince taneli ödüllerin etkisini değerlendirmek amacıyla bir yalıtım çalışması gerçekleştirdik. Katkılarını değerlendirmek için Atom-Searcher'ı iki alternatif çerçeve ile karşılaştırdık: (1) Base Zheng ve ark. (2025) tarafından önerilen DerinAraştırmacı ayarını ifade eder; bu, Atom-Searcher'in Atomik Düşünce ve RRM tarafından üretilen ince taneli ödüller olmadan hali anlamına gelir. (2) +RRM Base ayarına, Atom-Searcher ile aynı uygulama detaylarına sahip RRM tarafından üretilen ince taneli ödüllerin eklenmesini ifade eder; yani Atom-Searcher Atomik Düşünce olmadan. Tablo 2'de gösterildiği gibi, hem alan içi hem de alan dışı üzere yedi karşılaştırma ölçütündeki sonuçlar, +RRM 'nin Base 'e kıyasla anlamlı bir performans artışı sağlamadığını göstermektedir. Bu durum, RRM'nin ince taneli denetim için doğrudan kullanılmasının sınırlı faydalalar sunduğunu göstermektedir. Ancak, Atom-Searcher +RRM 'yi önemli ölçüde geride bırakarak, dört alan içi kıyaslamada ortalama %6,1 ve üç alan dışı kıyaslamada %2,5 performans artışı elde etmiş, bu da Atomik Düşünce'nin katkısını göstermektedir

three out-of-domain benchmarks, demonstrating the contribution of Atomic Thought. The above results raise an interesting question: **why does direct supervision using RRM have minimal effect on the reasoning process, while its effectiveness significantly improves after decomposing the reasoning process into Atom Thoughts?** We speculate that this is because **Atom Thoughts provide supervision anchors for RRM, helping it focus on the effective functional modules in the reasoning process, thereby generating meaningful fine-grained reward signals** (ATR in Atom-Searcher).

3.7 Case Study

Figure 4 analyzes the behavioral differences between Atom-Searcher and the SOTA baseline DeepResearcher in completing a deep research task. It demonstrates the following advantages of Atom-Searcher: (1) Atom-Searcher employs Atomic Thoughts in its reasoning, which leads to more human-like cognitive behaviors, such as problem analysis, solution hypotheses, error prediction, and next-step planning, making its reasoning process deeper and clearer. (2) Atom-Searcher triggers more search calls, allowing it to obtain richer external information to ensure the correctness of the answer. These advantages indicate that Atom-Searcher has great potential in more complex deep research tasks.

Additionally, we analyzed the token frequency statistics for Atom-Searcher and DeepResearcher during the testing phase. The word cloud, shown in Figure 5, illustrates the most frequently occurring tokens. The top-5 most frequent tokens in Atom-Searcher are <observation>, <action>, hypothesis, risk, and <risk_analysis>, whereas the top-5 most frequent tokens in DeepResearcher are I, search, need, find, and from. This disparity suggests that, compared to DeepResearcher, Atom-Searcher better aligns with human-like efficient cognitive patterns when performing deep research tasks, with a stronger focus on in-depth problem analysis, hypothesis evaluation, risk assessment, and strategic planning.

4 Related Work

4.1 Prompt and SFT-based Agentic Deep Research

Early prompt-based paradigms rely on human-authored workflows to specify the interaction between LLMs and external knowledge sources. Wang et al. (2024b). For example, OpenResearcher Zheng et al. (2024), AirRAG Feng et al. (2025), IterDRAG Yue et al. (2024), Plan*RAG Verma et al. (2025), Search-o1 Li et al. (2025a), and Open Deep Search Alzubi et al. (2025) have advanced search capabilities via carefully designed workflows. However, their reliance on human-engineered prompts and interaction patterns imposes rigid behavior constraints, limiting adaptability. These limitations motivate a shift toward SFT-based approaches that support more flexible and adaptive search strategies Yu et al. (2024); Wang et al. (2024c). For example, CoRAG Wang et al. (2024c) employs Monte Carlo Tree Search (MCTS) to dynamically select document blocks under budget constraints. However, it suffers from high computational overhead and limited generalization to unseen scenarios due to its reliance on supervised signals.

4.2 RL-based Agentic Deep Research

As LLMs have achieved remarkable breakthroughs in reasoning through outcome-based RL Guo et al. (2025); Team et al. (2025), this paradigm is emerging as a promising direction for enhancing agentic deep research via end-to-end optimization, attracting growing interest and active exploration

Yukarıdaki sonuçlar ilginç bir soruyu gündeme getirmektedir: Neden doğrudan RRM denetimi akıl yürütme sürecinde minimal etkiye sahipken, işlem Atomik Düşüncelere ayrıldıktan sonra etkinliği önemli ölçüde artmaktadır? Bunun nedeni olarak, Atomik Düşüncelerin RRM için denetim dayanakları sağlama, böylece akıl yürütme sürecindeki etkili fonksiyonel modüllere odaklanmasıına yardımcı olması ve anlamlı ince taneli ödül sinyalleri oluşturmaması (Atom-Searcher'da ADO) olduğunu düşünüyoruz.

3.7 Vaka Çalışması

Şekil 4, Atom-Searcher ile SOTA temel çizgisi DerinAraştırmacı'nın derin bir araştırma görevini tamamlarken sergilediği davranışsal farklılıklarını analiz etmektedir. Atom-Searcher'in şu avantajları ortaya konulmaktadır: (1) Atom-Searcher, akıl yürütmesinde Atomik Düşünceler kullanmakta ve bu sayede problem analizi, çözüm hipotezleri, hata tahmini ve sonraki adım planlaması gibi insan benzeri bilişsel davranışlara yol açarak akıl yürütme sürecini daha derin ve daha net hale getirmektedir. (2) Atom-Searcher, daha fazla arama çağrıları başlatarak cevabın doğruluğunu sağlamak için daha zengin dış bilgileri elde etmesine olanak tanır. Bu avantajlar, Atom-Searcher'in daha karmaşık derin araştırma görevlerinde büyük potansiyele sahip olduğunu göstermektedir.

Ayrıca, test aşamasında Atom-Searcher ve DerinAraştırmacı'nın token sıklık istatistiklerini analiz etti. Şekil 5'te gösterilen kelime bulutu, en sık geçen tokenleri ortaya koymaktadır. Atom-Searcher'daki en sık geçen ilk 5 token <observation>, <action>, hypothesis, risk ve <risk_analysis> iken, DerinAraştırmacı'daki en sık geçen ilk 5 token I, search, need, find ve from 'dur. Bu farklılık, Atom-Searcher'in DerinAraştırmacı'ya kıyasla derin araştırma görevlerinde insan benzeri ve ritimli bilişsel kalıplarla daha iyi uyum sağladığını, derinlemesine sorun analizi, hipotez değerlendirmesi, risk analizi ve stratejik planlamaya daha güçlü odaklılığını göstermektedir.

4 İlgili Çalışmalar

4.1 Prompt ve SFT Tabanlı Ajanık Derin Araştırma

Erken dönem prompt tabanlı paradigmalar, LLM'ler ile dış bilgi kaynakları arasındaki etkileşimi belirlemek için insan tarafından oluşturulmuş iş akışlarına dayanır (Wang ve ark., 2024b). Örneğin, OpenResearcher (Zheng ve ark., 2024), AirRAG (Feng ve ark., 2025), IterDRAG (Yue ve ark., 2024), Plan*RAG (Verma ve ark., 2025), Search-o1 (Li ve ark., 2025a) ve Open Deep Search (Alzubi ve ark., 2025) titizlikle tasarlanmış iş akışları aracılığıyla arama yeteneklerini geliştirmiştir. Ancak, bu yaklaşımın insan tarafından tasarlanmış promptlara ve etkileşim modellerine olan bağımlılığı, katı davranış kısıtlamaları getirerek uyarlanabilirliği sınırlar. Bu sınırlamalar, daha esnek ve uyarlanabilir arama stratejilerini destekleyen SFT tabanlı yaklaşımlara doğru kayış teşvik etmektedir (Yu ve ark., 2024; Wang ve ark., 2024c). Örneğin, CoRAG (Wang ve ark., 2024c), bütçe kısıtlamaları altında belge bloklarını dinamik olarak seçmek için Monte Carlo Tree Search (MCTS) yöntemini kullanmaktadır. Ancak, denetimli sinyallere olan bağımlılığı nedeniyle yüksek hesaplama yükü ve görülmemiş senaryolara sınırlı genelleme sorunu yaşamaktadır.

4.2 Sonuç Temelli RL Tabanlı Ajanık Derin Araştırma

LLM'ler, sonuç temelli RL aracılığıyla Guo vd. (2025); Team vd. (2025) tarafından kayda değer atılımlar gerçekleştirdiğinden, bu paradyigma ajanık derin araştırmayı uçtan uca optimizasyon yoluyla geliştirme için umut vadeden bir yön olarak ortaya çıkmakta ve artan ilgi ile aktif keşfi çekmektedir.

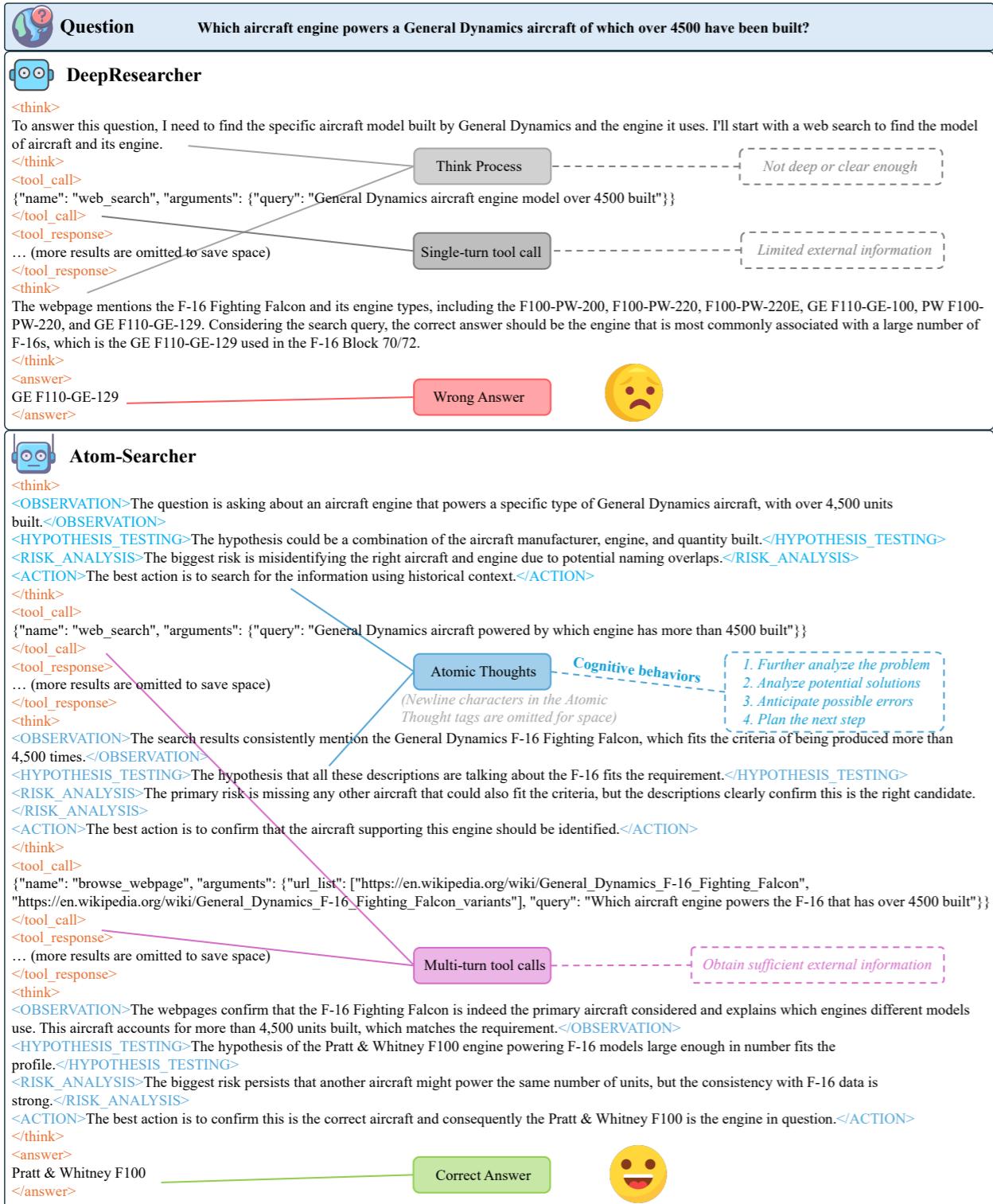


Figure 4 The case study demonstrates a comparison of the reasoning behavior between Atom-Searcher (below) and the SOTA baseline DeepResearcher (above).

from the research community. Recent works, such as ReSearch Chen et al. (2025), Search-R1 Jin et al. (2025), R1-Searcher Song et al. (2025), DeepResearcher Zheng et al. (2025), WebRL Qi et al. (2024), WebThinker Li et al. (2025b), ZeroSearch Sun et al. (2025) and WebAgent-RL Wei et al. (2025) have extended outcome-supervised reinforcement learning to the agentic deep research

araştırma topluluğundan. Chen vd. (2025) tarafından ReSearch, Jin vd. (2025) tarafından Search-R1, Song vd. (2025) tarafından R1-Searcher, Zheng vd. (2025) tarafından DerinAraştırmacı, Qi vd. (2024) tarafından WebRL, Li vd. (2025b) tarafından WebThinker, Sun vd. (2025) tarafından ZeroSearch ve Wei vd. (2025) tarafından WebAgent-RL gibi güncel çalışmalar, çıktı denetimli takviyeli öğrenmeyi ajanık derin araştırma alanına genişletmiştir.



Figure 5 Word cloud: Token frequency statistics of the responses during the testing phase for Atom-Searcher (a) and DeepResearcher (b).

setting, enabling LLMs to autonomously leverage search engines for complex reasoning tasks. Although enhancing agentic deep research with outcome-supervised reinforcement learning has led to performance gains, the coarse-grained reward signals provide limited guidance for learning efficient and intelligent search strategies, often resulting in suboptimal search calls. To overcome this, we propose an atomic thought-aware fine-grained reward to guide the model toward more efficient and intelligent search behaviors, while mitigating the training inefficiency caused by reward sparsity. Building on this, we further introduce a novel agentic deep research framework, Atom-Searcher, that integrates this reward formulation into a reinforcement learning paradigm.

5 Conclusion

In this work, we first introduce Atomic Thought, a novel LLM thinking paradigm designed to guide LLMs in clearer and more in-depth reasoning. We then supervise Atomic Thoughts using a Reasoning Reward Model to generate fine-grained Atomic Thought Reward and aggregate them with outcome reward through a training-dynamics-aware strategy. Based on this, we propose Atom-Searcher, a novel RL framework for agentic deep research, which advances the performance frontier of agentic deep research models by addressing the conflicting gradients and reward sparsity issues present in existing outcome-based deep research frameworks. Experimental results demonstrate the outstanding performance of Atom-Searcher and a range of impressive advantages.

References

- Salaheddin Alzubi, Creston Brooks, Purva Chiniya, Edoardo Contente, Chiara von Gerlach, Lucas Irwin, Yihan Jiang, Arda Kaz, Windsor Nguyen, Sewoong Oh, et al. Open deep search: Democratizing search with open-source reasoning agents. *arXiv preprint arXiv:2503.20201*, 2025.

John R Anderson, Michael Matessa, and Christian Lebiere. Act-r: A theory of higher level cognition and its relation to visual attention. *Human–Computer Interaction*, 12(4):439–462, 1997.

Mingyang Chen, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z Pan, Wen Zhang, Huajun Chen, Fan Yang, et al. Learning to reason with search for llms via reinforcement learning. *arXiv preprint arXiv:2503.19470*, 2025.

Mingyue Cheng, Yucong Luo, Jie Ouyang, Qi Liu, Huijie Liu, Li Li, Shuo Yu, Bohou Zhang, Jiawei Cao, Jie Ma, et al. A survey on knowledge-oriented retrieval-augmented generation. *arXiv preprint arXiv:2503.10677*, 2025.

Tianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine, and Yi Ma. Sft memorizes, rl generalizes: A comparative study of foundation model post-training. *arXiv preprint arXiv:2501.17161*, 2025.

Yuqin Dai, Shuo Yang, Guoqing Wang, Yong Deng, Zhanwei Zhang, Jun Yin, Pengyu Zeng, et al. Careful queries, credible results: Teaching rag models advanced web search tools with reinforcement learning, 2025. <https://arxiv.org/abs/2508.07956>.



Şekil 5 Kelime bulutu: Atom Searcher (a) ve PerinArşırmacı (b) için test aşamasındaki yanıtların token frekans istatistikleri.

yarı, LLM'lerin karmaşık akıl yürütme görevlerinde arama motorlarını otonom şekilde kullanmasını mümkün kılar. İkti denetimli takviyeli öğrenme ile ajanık derin araştırmaların güçlendirilmesi performans artışları sağlamlıksız olsa da, kaba taneli ödül sinyalleri etkin ve akıllı arama stratejilerinin öğrenilmesi için sınırlı haberler sunmakta ve sıkılıkla altoptimal arama çağrılarına yol açmaktadır. Bunu aşmak için, modeli daha verimli ve akıllı arama davranışlarına yönlendiren, atomik düşünceye duyarlı ince taneli bir ödül öneriyoruz; böylece ödül seyrekliğinin neden olduğu eğitim verimsizliği hafifletilmiş olur. Bunun üzerine inşa ederek, bu ödül formülasyonunu takviyeli öğrenme paradigmاسına entegre eden yeni bir ajanlık derin araştırma çerçevesi olan Atom-Searcher'ı tanıtıyoruz.

5 Sonuç

Bu çalışmada öncelikle LLM'leri daha net ve derin akıl yürütme süreçlerine yönlendirmek için tarihanmış yeni bir LLM düşünme paradigması olan Atomik Düşünce'yi tanıtıyoruz. Ardından, Atomik Düşünceleri yönetmek üzere bir Akıl Yürütme Ödül Modeli kullanıyor, ince taneli Atomik Düşünce Ödülü üretiyor ve bu ödülleri eğitim dinamiklerine duyarlı bir strateji kullanarak sonuç ödülüyle birleştiriyoruz. Buna dayanarak, mevcut sonuç tabanlı derin araştırma çerçevelerinde yaşanan çeşitlilik gradyanlar ve ödül seyrekliği sorunlarını çözerek ajanlık derin araştırma modellerinin performans sınırını ileri taşıyan yeni bir TL çerçevesi olan Atom-Searcher'ı öneriyoruz. Deneysel sonuçlar, Atom-Searcher'ın üstün performansını ve çeşitli etkileyici avantajlarını ortaya koymaktadır.

Kaynaklar

- alaheddin Alzubi, Creston Brooks, Purva Chiniya, Edoardo Contente, Chiara von Gerlach, Lucas Irwin, Yihan Jiang, Arda Kaz, Windsor Nguyen, Sewoong Oh ve diğerleri. Open deep search: Açık kaynaklı akıl yürütünen ajanlarla aramayı demokratikleştirmek. *arXiv* ön baskısı *arXiv:2503.20201*, 2025.

ohn R Anderson, Michael Matessa ve Christian Lebiere. Act-r: Daha yüksek düzey biliş teorisi ve bunun görsel dikkat ile ilişkisi. *Human–Computer Interaction*, 12(4):439–462, 1997.

Mingyang Chen, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z Pan, Wen Zhang, Huajun Chen, Fan Yang ve diğerleri. Takviyeli öğrenme yoluyla LLM'ler için aramaya akıl yürütmemeyi öğrenmek. *arXiv* ön baskısı *arXiv:2503.19470*, 2025.

Mingyue Cheng, Yucong Luo, Jie Ouyang, Qi Liu, Huijie Liu, Li Li, Shuo Yu, Bohou Zhang, Jiawei Cao, Jie Ma ve diğerleri. Bilgi odaklı arama destekli üretim üzerine bir derleme. *arXiv* ön baskısı *arXiv:2503.10677*, 2025.

ianzhe Chu, Yuexiang Zhai, Jihan Yang, Shengbang Tong, Saining Xie, Dale Schuurmans, Quoc V Le, Sergey Levine ve Yi Ma. Sft ezberler, rl genelleştirir: Temel model eğitim sonrası karşılaştırmalı bir çalışma. *arXiv* ön baskısı *arXiv:2501.17161*, 2025.

uqin Dai, Shuo Yang, Guoqing Wang, Yong Deng, Zhanwei Zhang, Jun Yin, Pengyu Zeng ve diğerleri. Titiz sorgular, güvenilir sonuçlar: Takviyeli öğrenme ileraq modellerine gelişmiş web arama araçlarının öğretilmesi, 2025. <https://arxiv.org/abs/2508.07956>.

- Yuqing Du, Alexander Havrilla, Sainbayar Sukhbaatar, Pieter Abbeel, and Roberta Raileanu. A study on improving reasoning in language models. In *I Can't Believe It's Not Better Workshop: Failure Modes in the Age of Foundation Models*, 2024.
- Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6491–6501, 2024.
- Wenfeng Feng, Chuzhan Hao, Yuwei Zhang, Jingyi Song, and Hao Wang. Airrag: Activating intrinsic reasoning for retrieval augmented generation via tree-based search. *arXiv preprint arXiv:2501.10053*, 2025.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1), 2023.
- Google. Gemini deep research. Technical report, 2024.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.
- Mark K Ho and Thomas L Griffiths. Cognitive science as a source of forward and inverse models of human decisions for robotics and control. *Annual Review of Control, Robotics, and Autonomous Systems*, 5(1):33–53, 2022.
- Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*, 2020.
- Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.
- Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.
- Bowen Jin, Jinsung Yoon, Jiawei Han, and Sercan O Arik. Long-context llms meet rag: Overcoming challenges for long inputs in rag. *arXiv preprint arXiv:2410.05983*, 2024.
- Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*, 2025.
- Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.
- Jaehyung Kim, Jaehyun Nam, Sangwoo Mo, Jongjin Park, Sang-Woo Lee, Minjoon Seo, Jung-Woo Ha, and Jinwoo Shin. Sure: Summarizing retrievals using answer candidates for open-domain qa of llms. *arXiv preprint arXiv:2404.13081*, 2024.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.
- Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. Search-o1: Agentic search-enhanced large reasoning models. *arXiv preprint arXiv:2501.05366*, 2025a.
- Xiaoxi Li, Jiajie Jin, Guanting Dong, Hongjin Qian, Yutao Zhu, Yongkang Wu, Ji-Rong Wen, and Zhicheng Dou. Webthinker: Empowering large reasoning models with deep research capability. *arXiv preprint arXiv:2504.21776*, 2025b.
- Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2023.
- Zijun Liu, Peiyi Wang, Runxin Xu, Shirong Ma, Chong Ruan, Peng Li, Yang Liu, and Yu Wu. Inference-time scaling for generalist reward modeling. *arXiv preprint arXiv:2504.02495*, 2025.
- Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*, 7, 2022.
- OpenAI. Learning to reason with llms. Technical report, 2024.
- OpenAI. Deep research system card. Technical report, 2025.
- Yuqing Du, Alexander Havrilla, Sainbayar Sukhbaatar, Pieter Abbeel ve Roberta Raileanu. Dil modellerinde muhakemeyi geliş-tirmeye yönelik bir çalışma. In *I Can't Believe It's Not Better Workshop: Failure Modes in the Age of Foundation Models*, 2024.
- Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua ve Qing Li. RAG ve LLM'ler üzerine bir anket: Arama destekli büyük dil modellerine doğru. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, sayfa 6491–6501, 2024.
- Wenfeng Feng, Chuzhan Hao, Yuwei Zhang, Jingyi Song ve Hao Wang. Airrag: Ağaç tabanlı arama yoluyla arama destekli üretimde içsel muhakemenin etkinleştirilmesi. *arXiv ön baskısı arXiv:2501.10053*, 2025.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang ve Haofen Wang. Büyük dil modelleri için Arama destekli üretim: Bir derleme. *arXiv ön baskısı arXiv:2312.10997*, 2(1), 2023.
- Google. Gemini derin araştırması. Teknik rapor, 2024.
- Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi ve diğerleri. Deepseek-r1: Takviyeli öğrenme yoluyla LLM'lerde akıl yürütme yeteneğinin teşvik edilmesi. *arXiv ön baskısı arXiv:2501.12948*, 2025.
- Mark K. Ho ve Thomas L. Griffiths. Robotik ve kontrol için insan kararlarının ileri ve ters modellerinin kaynağı olarak bilişsel bilim. *Annual Review of Control, Robotics, and Autonomous Systems*, 5(1):33–53, 2022.
- Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara ve Akiko Aizawa. Akıl yürütme adımlarının kapsamlı değerlendirilmesi amacıyla çoklu adımlı Soru-Cevap veri seti oluşturulması. *arXiv ön baskısı arXiv:2011.01060*, 2020.
- Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu ve diğerleri. Qwen2. 5-coder teknik raporu. *arXiv ön baskısı arXiv:2409.12186*, 2024.
- Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford ve diğerleri. Gpt-4o sistem kartı. *arXiv ön baskısı arXiv:2410.21276*, 2024.
- Bowen Jin, Jinsung Yoon, Jiawei Han ve Sercan O. Arik. Uzun-bağlam LLM'ler rag ile buluşuyor: rag'de uzun girdiler için karşılaşılan zorlukların aşılması. *arXiv ön baskısı arXiv:2410.05983*, 2024.
- Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani ve Jiawei Han. Search-r1: Takviyeli öğrenme ile LLM'leri arama motorlarını kullanarak akıl yürütme ve faydalananma konusunda eğitmek. *arXiv ön baskısı arXiv:2503.09516*, 2025.
- Mandar Joshi, Eunsol Choi, Daniel S. Weld ve Luke Zettlemoyer. Triviaqa: Okuma-anlama için büyük ölçekli uzak denetimli zorlu bir veri seti. *arXiv ön baskısı arXiv:1705.03551*, 2017.
- Jaehyung Kim, Jaehyun Nam, Sangwoo Mo, Jongjin Park, Sang-Woo Lee, Minjoon Seo, Jung-Woo Ha ve Jinwoo Shin. Elbette: LLM'lerin açık alan soru-cevaplamasında yanıt adayları kullanılarak aramaların özetlenmesi. *arXiv ön baskısı arXiv:2404.13081*, 2024.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee ve diğerleri. Natural Questions: Soru-cevap araştırmaları için bir benchmark. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel ve diğerleri. Bilgi yoğun NLP görevleri için arama destekli üretim. *Sınırsız Bilgi İşleme Sistemlerinde Gelişmeler*, 33:9459–9474, 2020.
- Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang ve Zhicheng Dou. Search-o1: Ajanık arama destekli büyük muhakeme modelleri. *arXiv ön baskısı arXiv:2501.05366*, 2025a.
- Xiaoxi Li, Jiajie Jin, Guanting Dong, Hongjin Qian, Yutao Zhu, Yongkang Wu, Ji-Rong Wen ve Zhicheng Dou. Webthinker: Büyük muhakeme modellerini derin araştırma yeteneğiyle güçlendirmek. *arXiv ön baskısı arXiv:2504.21776*, 2025b.
- Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever ve Karl Cobbe. Adım adım doğrulayalım. On İkinci Uluslararası Öğrenme Temsilleri Konferansı, 2023.
- Zijun Liu, Peiyi Wang, Runxin Xu, Shirong Ma, Chong Ruan, Peng Li, Yang Liu ve Yu Wu. Genel amaçlı ödül modellemeye çıkarım-zamanı ölçeklendirmesi. *arXiv ön baskısı arXiv:2504.02495*, 2025.
- Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi ve Daniel Khashabi. Dil modellerine ne zaman güvenilmemeli: Parametrik ve parametrik olmayan belleklerin etkinliği ve sınırlamalarının araştırılması. *arXiv ön baskısı arXiv:2212.10511*, 7, 2022.
- OpenAI. LLM'lerle akıl yürütme öğrenme. Teknik rapor, 2024.
- OpenAI. Derin araştırma sistem kartı. Teknik rapor, 2025.

- Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*, 2022.
- Zehan Qi, Xiao Liu, Iat Long Iong, Hanyu Lai, Xueqiao Sun, Wenyi Zhao, Yu Yang, Xinyue Yang, Jiadai Sun, Shuntian Yao, et al. Webrl: Training llm web agents via self-evolving online curriculum reinforcement learning. *arXiv preprint arXiv:2411.02337*, 2024.
- Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianshi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. <https://arxiv.org/abs/2412.15115>.
- Abdul Malik Sami, Zeeshan Rasheed, Kai-Kristian Kemell, Muhammad Waseem, Terhi Kilamo, Mika Saari, Anh Nguyen Duc, Kari Systä, and Pekka Abrahamsson. System for systematic literature review using multiple ai agents: Concept and an empirical evaluation. *arXiv preprint arXiv:2403.08399*, 2024.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*, 2024.
- Aditi Singh, Abul Ehtesham, Saket Kumar, and Tala Talaei Khoei. Agentic retrieval-augmented generation: A survey on agentic rag. *arXiv preprint arXiv:2501.09136*, 2025.
- Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.
- Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *arXiv preprint arXiv:2503.05592*, 2025.
- Maojia Song, Shang Hong Sim, Rishabh Bhardwaj, Hai Leong Chieu, Navonil Majumder, and Soujanya Poria. Measuring and enhancing trustworthiness of llms in rag through grounded attributions and learning to refuse. *arXiv preprint arXiv:2409.11242*, 2024.
- Hao Sun, Zile Qiao, Jiayan Guo, Xuanbo Fan, Yingyan Hou, Yong Jiang, Pengjun Xie, Yan Zhang, Fei Huang, and Jingren Zhou. Zerosearch: Incentivize the search capability of llms without searching. *arXiv preprint arXiv:2505.04588*, 2025.
- Kimi Team, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao, et al. Kimi k1. 5: Scaling reinforcement learning with llms. *arXiv preprint arXiv:2501.12599*, 2025.
- Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.
- Prakhar Verma, Sukruta Prakash Midigesi, Gaurav Sinha, Arno Solin, Nagarajan Natarajan, and Amit Sharma. Plan rag: Efficient test-time planning for retrieval augmented generation. In *Workshop on Reasoning and Planning for Large Language Models*, 2025.
- Shuting Wang, Jiejun Tan, Zhicheng Dou, and Ji-Rong Wen. Omnieval: An omnidirectional and automatic rag evaluation benchmark in financial domain. *arXiv preprint arXiv:2412.13018*, 2024a.
- Xiaohua Wang, Zhenghua Wang, Xuan Gao, Feiran Zhang, Yixin Wu, Zhibo Xu, Tianyuan Shi, Zhengyuan Wang, Shizheng Li, Qi Qian, et al. Searching for best practices in retrieval-augmented generation. *arXiv preprint arXiv:2407.01219*, 2024b.
- Ziting Wang, Haitao Yuan, Wei Dong, Gao Cong, and Feifei Li. Corag: A cost-constrained retrieval optimization system for retrieval-augmented generation. *arXiv preprint arXiv:2411.00744*, 2024c.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.
- Zhepei Wei, Wenlin Yao, Yao Liu, Weizhi Zhang, Qin Lu, Liang Qiu, Changlong Yu, Puyang Xu, Chao Zhang, Bing Yin, et al. Webagent-r1: Training web agents via end-to-end multi-turn reinforcement learning. *arXiv preprint arXiv:2505.16421*, 2025.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengan Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.
- Shuo Yang, Yuqin Dai, Guoqing Wang, Xinran Zheng, Jinfeng Xu, Jinze Li, Zhenzhe Ying, Weiqiang Wang, and Edith C. H. Ngai. Realfactbench: A benchmark for evaluating large language models in real-world fact-checking, 2025b.
- Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith ve Mike Lewis. Dil modellerindeki bileşimsel uyum-suzluğu ölçme ve azaltma. *arXiv ön baskısı arXiv:2210.03350* , 2022.
- Zehan Qi, Xiao Liu, Iat Long Iong, Hanyu Lai, Xueqiao Sun, Wenyi Zhao, Yu Yang, Xinyue Yang, Jiadai Sun, Shuntian Yao ve diğerleri. Webrl: LLM web ajanlarını kendi kendini geliştiren çevrimiçi müfredatlı takviyeli öğrenme ile eğitmek. *arXiv ön baskısı arXiv:2411.02337* , 2024.
- Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianshi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang ve Zihan Qiu. Qwen2.5 teknik raporu, 2025. <https://arxiv.org/abs/2412.15115>.
- Abdul Malik Sami, Zeeshan Rasheed, Kai-Kristian Kemell, Muhammad Waseem, Terhi Kilamo, Mika Saari, Anh Nguyen Duc, Kari Systä ve Pekka Abrahamsson. Çoklu yapay zeka ajanları kullanarak sistematik literatür inceleme sistemi: Kavram ve ampirik değerlendirme. *arXiv ön baskısı arXiv:2403.08399* , 2024.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu ve diğerleri. Deepseekmath: Açık dil modellerinde matematiksel akıl yürütmenin sınırlarını zorlamak. *arXiv ön baskısı arXiv:2402.03300* , 2024.
- Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin ve Chuan Wu. Hybridflow: Esnek ve verimli bir takviyeli öğrenme çerçevesi. *arXiv ön baskısı arXiv: 2409.19256*, 2024.
- Aditi Singh, Abul Ehtesham, Saket Kumar ve Tala Talaei Khoei. Ajanık arama destekli üretim: Ajanık RAG üzerine bir inceleme. *arXiv ön baskısı arXiv:2501.09136*, 2025.
- Charlie Snell, Jaehoon Lee, Kelvin Xu ve Aviral Kumar. LLM test zamanı hesaplama kaynaklarının ölçeklendirilmesi, model parametrelerini ölçeklendirmekten daha etkili olabilir. *arXiv ön baskısı arXiv:2408.03314* , 2024.
- Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang ve Ji-Rong Wen. R1-searcher: LLM'lerde arama yeteneğinin takviyeli öğrenme ile teşvik edilmesi. *arXiv ön baskısı arXiv:2503.05592*, 2025.
- Maojia Song, Shang Hong Sim, Rishabh Bhardwaj, Hai Leong Chieu, Navonil Majumder ve Soujanya Poria. Yerleşik atıflar ve reddetmeyi öğrenme yoluyla rag'de LLM'lerin güvenilirliğinin ölçülmesi ve artırılması. *arXiv ön baskısı arXiv:2409.11242* , 2024.
- Hao Sun, Zile Qiao, Jiayan Guo, Xuanbo Fan, Yingyan Hou, Yong Jiang, Pengjun Xie, Yan Zhang, Fei Huang ve Jingren Zhou. Zerosearch: Araştırma yapmadan LLM'lerin arama yeteneğini teşvik etmek. *arXiv ön baskısı arXiv:2505.04588*, 2025.
- Kimi Takımı, Angang Du, Bofei Gao, Bowei Xing, Changjiu Jiang, Cheng Chen, Cheng Li, Chenjun Xiao, Chenzhuang Du, Chonghua Liao ve diğerleri. Kimi k1.5: LLM'lerle takviyeli öğrenmenin ölçeklendirilmesi. *arXiv ön baskısı arXiv:2501.12599* , 2025.
- Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot ve Ashish Sabharwal. Musique: Tek adımlı soru bileşimi yoluyla çok adımlı sorular. *İşlemsel Dilbilim Derneği Yayınları* , 10:539–554, 2022.
- Prakhar Verma, Sukruta Prakash Midigesi, Gaurav Sinha, Arno Solin, Nagarajan Natarajan ve Amit Sharma. Plan rag: Arama destekli üretim için etkili test zamanı planlaması. *Büyük Dil Modelleri için Muhakeme ve Planlama Çalıştayı'nda* , 2025.
- Shuting Wang, Jiejun Tan, Zhicheng Dou ve Ji-Rong Wen. Omnieval: Finansal alanda çok yönlü ve otomatik rag değerlendirme kıştası. *arXiv ön baskısı arXiv:2412.13018* , 2024a.
- Xiaohua Wang, Zhenghua Wang, Xuan Gao, Feiran Zhang, Yixin Wu, Zhibo Xu, Tianyuan Shi, Zhengyuan Wang, Shizheng Li, Qi Qian ve diğerleri. Arama destekli üretimde en iyi uygulamaların araştırılması. *arXiv ön baskısı arXiv:2407.01219* , 2024b.
- Ziting Wang, Haitao Yuan, Wei Dong, Gao Cong ve Feifei Li. Corag: Arama destekli üretim için maliyet kısıtlı arama optimizasyon sistemi. *arXiv ön baskısı arXiv:2411.00744* , 2024c.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou ve diğerleri. Zincir-düşünce yönlendirmesi, büyük dil modellerinde muhakemeyi tetikler. *Sınırsız Bilgi İşleme Sistemlerinde Gelişmeler* , 35:24824–24837, 2022.
- Zhepei Wei, Wenlin Yao, Yao Liu, Weizhi Zhang, Qin Lu, Liang Qiu, Changlong Yu, Puyang Xu, Chao Zhang, Bing Yin ve diğerleri. Webagent-r1: Uçtan uca çok tur takviyeli öğrenme ile web ajanlarının eğitimi. *arXiv ön baskısı arXiv:2505.16421* , 2025.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengan Huang, Chenxu Lv ve diğerleri. Qwen3 teknik raporu. *arXiv ön baskısı arXiv:2505.09388* , 2025a.
- Shuo Yang, Yuqin Dai, Guoqing Wang, Xinran Zheng, Jinfeng Xu, Jinze Li, Zhenzhe Ying, Weiqiang Wang ve Edith C. H. Ngai. Realfactbench: Büyük dil modellerinin gerçek dünya bilgi doğrulamasında değerlendirilmesi için bir kıyaslama, 2025b.

- Shuo Yang, Zijian Yu, Zhenzhe Ying, Yuqin Dai, Guoqing Wang, Jun Lan, Jinfeng Xu, Jinze Li, and Edith C. H. Ngai. Rama: Retrieval-augmented multi-agent framework for misinformation detection in multimodal fact-checking, 2025c.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*, 2018.
- Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.
- Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu, et al. Dapo: An open-source llm reinforcement learning system at scale. *arXiv preprint arXiv:2503.14476*, 2025.
- Tian Yu, Shaolei Zhang, and Yang Feng. Auto-rag: Autonomous retrieval-augmented generation for large language models. *arXiv preprint arXiv:2411.19443*, 2024.
- Zhenrui Yue, Honglei Zhuang, Ajun Bai, Kai Hui, Rolf Jagerman, Hansi Zeng, Zhen Qin, Dong Wang, Xuanhui Wang, and Michael Bendersky. Inference scaling for long-context retrieval augmented generation. *arXiv preprint arXiv:2410.04343*, 2024.
- Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. When scaling meets llm finetuning: The effect of data, model and finetuning method. *arXiv preprint arXiv:2402.17193*, 2024.
- Yuxiang Zheng, Shichao Sun, Lin Qiu, Dongyu Ru, Cheng Jiayang, Xuefeng Li, Jifan Lin, Binjie Wang, Yun Luo, Renjie Pan, et al. Openresearcher: Unleashing ai for accelerated scientific research. *arXiv preprint arXiv:2408.06941*, 2024.
- Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. Deepresearcher: Scaling deep research via reinforcement learning in real-world environments. *arXiv preprint arXiv:2504.03160*, 2025.
- Shuo Yang, Zijian Yu, Zhenzhe Ying, Yuqin Dai, Guoqing Wang, Jun Lan, Jinfeng Xu, Jinze Li ve Edith C. H. Ngai. Rama: Multimodal bilgi doğrulamada yanlış bilgi tespiti için geri çağrıma destekli çok ajanlı çerçeve, 2025c.
- Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov ve Christopher D. Manning. Hotpotqa: Çeşitli ve açıklanabilir çok adımlı soru-cevaplama için bir veri seti. *arXiv ön baskısı arXiv:1809.09600*, 2018.
- Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao ve Karthik Narasimhan. Tree of thoughts: Büyük dil modelleri ile bilinçli problem çözümü. *Sinirsel Bilgi İşleme Sistemlerinde Gelişmeler*, 36:11809–11822, 2023.
- Qiying Yu, Zheng Zhang, Ruofei Zhu, Yufeng Yuan, Xiaochen Zuo, Yu Yue, Weinan Dai, Tiantian Fan, Gaohong Liu, Lingjun Liu ve diğerleri. Dapo: Ölçekli açık kaynaklı bir llm takviyeli öğrenme sistemi. *arXiv ön baskısı arXiv:2503.14476*, 2025.
- Tian Yu, Shaolei Zhang ve Yang Feng. Auto-rag: Büyük dil modelleri için otomatik arama destekli üretim. *arXiv ön baskısı arXiv:2411.19443*, 2024.
- Zhenrui Yue, Honglei Zhuang, Ajun Bai, Kai Hui, Rolf Jagerman, Hansi Zeng, Zhen Qin, Dong Wang, Xuanhui Wang ve Michael Bendersky. Uzun bağlılı arama destekli üretim için çıkarım ölçeklendirme. *arXiv ön baskısı arXiv:2410.04343*, 2024.
- Biao Zhang, Zhongtao Liu, Colin Cherry ve Orhan Firat. Ölçeklendirme ile LLM ince ayarının kesişimi: Veri, model ve ince ayar yönteminin etkisi. *arXiv ön baskısı arXiv:2402.17193*, 2024.
- Yuxiang Zheng, Shichao Sun, Lin Qiu, Dongyu Ru, Cheng Jiayang, Xuefeng Li, Jifan Lin, Binjie Wang, Yun Luo, Renjie Pan ve diğerleri. Openresearcher: Hızlandırılmış bilimsel araştırmalar için yapay zekanın etkinleştirilmesi. *arXiv ön baskısı arXiv:2408.06941*, 2024.
- Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu ve Pengfei Liu. DerinAraştırmacı: Gerçek dünya ortamlarında takviyeli öğrenme ile derin araştırmaların ölçeklendirilmesi. *arXiv ön baskısı arXiv:2504.03160*, 2025.

A Training Details

A.1 Sliding-Window-based Entropy Regulation Mechanism

A major obstacle in scaling reinforcement learning for LLMs is the occurrence of entropy collapse [Yu et al. \(2025\)](#), characterized by a rapid sharp drop in policy entropy at the early training stage, which results in an overconfident policy and severely impairs exploration. To mitigate policy entropy collapse, we introduce a Sliding-Window-based dynamic Entropy Regulation Mechanism (**SWERM**) applied at the granularity of training steps. Before introducing SWERM, we first define policy entropy \mathcal{H} as the average token-level entropy of the policy model $\pi_{\theta'}$ over the current batch \mathcal{B} , which can be formulated as follows:

$$\mathcal{H}(\pi_{\theta'}, \mathcal{B}) = -\mathbb{E}_{\mathcal{B}, \pi_{\theta'}}[\log \pi_{\theta'}(y_t | \mathbf{y}_{<t})] = -\frac{1}{|\mathcal{B}|} \sum_{x \in \mathcal{B}} \frac{1}{|\mathbf{y}|} \sum_{t=1}^{|\mathbf{y}|} \mathbb{E}_{y_t \sim \pi_{\theta'}}[\log \pi_{\theta'}(y_t | \mathbf{y}_{<t}, x)] \quad (13)$$

where x represents an input sampled from \mathcal{B} , y_t denotes the token generated at time step t , and $\mathbf{y}_{<t}$ denotes the prefix sequence consisting of the first $t-1$ tokens. In SWERM, a sliding window of size k is employed to track the average policy entropy over the latest k training steps, and is defined as:

$$\bar{\mathcal{H}}_T = \frac{1}{k} \sum_{i=T-k+1}^T \mathcal{H}_i \quad (14)$$

where T denotes the current training step, \mathcal{H}_i denotes the policy entropy at training step i and $\bar{\mathcal{H}}_T$ is the average entropy computed over the sliding window at step T . To monitor the stability of entropy reduction during training, we quantify the drop in $\bar{\mathcal{H}}$ from step $T-1$ to step T as follows:

$$\Delta \bar{\mathcal{H}}_T = \bar{\mathcal{H}}_{T-1} - \bar{\mathcal{H}}_T \quad (15)$$

$\Delta \bar{\mathcal{H}}_T$ serves as an effective indicator for measuring the smoothness of entropy drop. When $\Delta \bar{\mathcal{H}}_T > \tau$ (where τ is a threshold hyperparameter), it indicates a collapse in \mathcal{H}_T , which significantly pulls down

A Eğitim Detayları

A.1 Kaydırmalı Pencere Temelli Entropi Düzenleme Mekanizması

Takviyeli öğrenmenin LLM'ler için ölçeklendirilmesinde önemli bir engel entropi çöküşünün meydana gelmesidir [Yu ve ark. \(2025\)](#), başlangıç eğitim aşamasında politik entropide ani ve keskin bir düşüşle karakterize edilen, bu durumun aşırı kendinden emin bir politika ile sonuçlanıp keşfi ciddi şekilde engellediği bildirildi. Politik entropi çöküşünü azaltmak için, eğitim adımlarının inceliğinde uygulanan Kaydırmalı Pencere Temelli Dinamik Entropi Düzenleme Mekanizması (**SWERM**) tanıtıyoruz. SWERM'i tanıtmadan önce, politika entropisi H' yi geçerli B partisi üzerindeki politika modeli $\pi^{\theta'}$ 'nin ortalama token seviyesi entropisi olarak tanımlıyoruz; bu şekilde formüle edilebilir:

$$\mathcal{H}(\pi_{\theta'}, \mathcal{B}) = -\mathbb{E}_{\mathcal{B}, \pi_{\theta'}}[\log \pi_{\theta'}(y_t | \mathbf{y}_{<t})] = -\frac{1}{|\mathcal{B}|} \sum_{x \in \mathcal{B}} \frac{1}{|\mathbf{y}|} \sum_{t=1}^{|\mathbf{y}|} \mathbb{E}_{y_t \sim \pi_{\theta'}}[\log \pi_{\theta'}(y_t | \mathbf{y}_{<t}, x)] \quad (13)$$

Burada x , \mathcal{B} 'den seçilen bir girdi örneğini; y_t , t zaman adımıda oluşturulan tokeni; ve $\mathbf{y}_{<t}$, ilk $t-1$ tokeni içeren önek dizisini temsil etmektedir. SWERM'de, k büyülüüğünde bir kayan pencere kullanılarak son k eğitim adımlarının ortalama politik entropisi izlenmekte ve şu şekilde tanımlanmaktadır:

$$\bar{\mathcal{H}}_T = \frac{1}{k} \sum_{i=T-k+1}^T \mathcal{H}_i \quad (14)$$

Burada T , mevcut eğitim adımı; \mathcal{H}_i , i aşamasındaki politik entropiyi; ve $\bar{\mathcal{H}}_T$, T aşamasında kayan pencere üzerinden hesaplanan ortalama entropiyi ifade etmektedir. Eğitim sırasında entropi azalmasının kararlılığını izlemek için, $\bar{\mathcal{H}}_T$ değerindeki azalmayı $T-1$ adımdan T adıma kadar aşağıdaki şekilde niceleştiriyoruz:

$$\Delta \bar{\mathcal{H}}_T = \bar{\mathcal{H}}_{T-1} - \bar{\mathcal{H}}_T \quad (15) \Delta$$

$\bar{\mathcal{H}}_T$, entropi azalmasının düzgünlüğünü ölçmek için etkili bir gösterge olarak işlev görür. Eğer $\Delta \bar{\mathcal{H}}_T > \tau$ (burada τ bir eşik hyperparametredir), bu \mathcal{H}_T değerinde çokmeyi gösterir ve bu durum entropinin ölçüde düşmesine yol açar.

Background Knowledge:
The following is a deep-research scenario:
Lines beginning with user indicate user questions.
Lines beginning with assistant represent the deep-research agent's reasoning content.
The content inside <think>xxx</think> represents the agent's reasoning process.
The segments enclosed within <xxx>...</xxx> indicate an atomic thought action. For example, <PLAN> represents the atomic action of planning, and <PLAN>xxx</PLAN> indicates a specific plan for solving the problem.
<tool_call>xxx</tool_call> shows how the deep-research agent decides to invoke a tool after reasoning (mainly including web_search [search tool] and browse_webpage [a tool to retrieve webpage content from a URL]).
<tool_response>xxx</tool_response> indicates the result returned from the tool (e.g., search result or webpage content).
<answer>xxx</answer> represents the final answer.

Task [TASK]:
You are a superintelligent expert agent, more capable than the deep-research agent. You are now required to evaluate the agent's reasoning and tool usage. The scoring rules are as follows:
You must first explain the meaning of each atomic thought action. For example, <PLAN> represents the planning atomic action, and <PLAN>xxx</PLAN> indicates the agent's specific plan for this problem.
For each atomic action, you must define a scoring rubric based on its actual result—what constitutes good or poor performance. Minimum score: -3; Maximum score: 5; The score should remain between -3 and 5.

Based on the rules in step 2, assign scores to each atomic action's performance. You must evaluate step by step, and finally give a specific score. Return the scoring results from step 3 in JSON format. The key should be the atomic action name. The value should be the corresponding score. For example: {"PLAN": 0, "xxx": 3}

Reference Answer Example:
example:
Explanation of each atomic thought action: xxxxxxx
Scoring rules for each action: xxxxxxx
Step-by-step evaluation for each atomic action:
Final output: The final scoring result is:
{
 "XXX": 0,
 "XXX": 0,
 "XXX": 0,
 "analysis": "Brief explanation of the score..."
}

Arka Plan Bilgisi:
Aşağıda derin araştırma senaryosu verilmiştir:
user ile başlayan satırlar kullanıcı sorularını belirtir.
assistant ile başlayan satırlar derin araştırma ajanının akıl yürütme içeriğini temsil eder.
<think>xxx</think> içindeki içerik ajanın akıl yürütme sürecini temsil eder.
<xxx>...</xxx> ile çevrilmiş bölümler atomik düşünce eylemini gösterir. Örneğin, <PLAN> planlama atomik eylemini temsil eder ve <PLAN>xxx</PLAN> problemin çözümü için belirli bir planı ifade eder.
<tool_call>xxx</tool_call>, derin araştırma ajanının bir aracı kullanmaya karar vermeden önceki muhakemesini göstermektedir (başlıca web_search [arama aracı] ve browse_webpage [bir URL'den web sayfası içeriğini alan araç] içerir). <tool_response>xxx</tool_response>, araçtan dönen sonucu (örneğin arama sonucu veya web sayfası içeriği) belirtir.
<answer>xxx</answer> nihai cevabı temsil eder.

Görev [TASK]:
Siz, derin araştırma ajanından daha yetkin, süperzekâ uzman bir ajansınız. Şimdi, ajanın muhakemesini ve araç kullanımını değerlendirmeniz gerekmektedir. Puanlama kuralları aşağıdaki gibidir:
Her atomik düşünce eyleminin anlamını önce açıklamalısınız. Örneğin, <PLAN> planlama atomik eylemini temsil eder ve <PLAN>xxx</PLAN> ajanının bu probleme yönelik spesifik planını gösterir.
Her atomik eylem için, gerçek sonucu temel alan bir puanlama ölçütü tanımlamalısınız; iyi veya kötü performans neyi oluşturur belirtmelisiniz. Minimum puan: -3; Maksimum puan: 5; Skor -3 ile 5 arasında kalmalıdır.

Adım 2'deki kurallara dayanarak, her atomik eylemin performansına puan atayın. Adım adım değerlendirme yapmalı ve sonunda belirli bir puan vermelisiniz. Adım 3'ten puanlama sonuçlarını JSON formatında döndürün.
Anahtar, atomik eylemin ismi olmalıdır. Değer ise karşılık gelen puan olmalıdır. Örneğin: {"PLAN": 0, "xxx": 3}

Referans Cevap Örneği:
örnek:
Her atomik düşünce eyleminin açıklaması: xxxxxxx
Her eylem için puanlama kuralları: xxxxxxx Her atomik eylem için adım adım değerlendirme:
Nihai çıktı: Nihai puanlama sonucu şudur:
{
 "XXX": 0,
 "XXX": 0,
 "XXX": 0,
 "analysis": "Puanın kısa açıklaması..."
}

Şekil 6 RRM'nin Atomik Düşünceleri değerlendirmesi için istem.

$\Delta \bar{\mathcal{H}}_T$. In contrast, $\Delta \bar{\mathcal{H}}_T \leq \tau$ suggests that the policy entropy is dropping smoothly. Accordingly, to mitigate entropy collapse, we increase the policy temperature and resample the outputs on the current batch whenever $\Delta \bar{\mathcal{H}}_T > \tau$ is detected.

B Prompts Employed in Atom-Searcher

B Atom-Searcher'da Kullanılan İstemler

Background Knowledge:
The following is a deep-research scenario:
Lines beginning with **user** indicate user questions.
Lines beginning with **assistant** represent the thinking process of the deep-research agent.
The content enclosed in <think>...</think> reflects the agent's internal reasoning.
The content enclosed in <tool_call>...</tool_call> indicates how the deep-research agent, after reasoning, invokes external tools (mainly: web_search [search engine] or browse_webpage [to retrieve webpage content from a URL]).
The content within <tool_response>...</tool_response> shows the result returned by the tool (e.g., search results or retrieved webpage content).
The content within <answer>...</answer> is the final answer generated by the agent.

Task [TASK]:
You are a superintelligent agent expert—smarter than the deep-research agent.
Your task is to evaluate the deep-research agent's reasoning and tool usage based on the following scoring criteria:

【Evaluation Dimensions】

1. **Search Strategy Intelligence** (0–5 points):

- Evaluate diversity of sources, use of advanced search syntax, and appropriateness of time filters.
- 5 points: Cross-platform/multilingual queries, use of Boolean logic, quotation marks for exact matches, etc.
- 4 points: Keyword variation and improvement across search rounds that meaningfully enhance information retrieval.
- 3 points: Basic keyword search with no advanced filtering.
- 0 points: Repeated or irrelevant sources; invalid tool usage.

2. **Logical Reasoning Quality** (0–5 points):

- Evaluate hypothesis formulation, evidence use, and consistency of conclusions.
- 5 points: Fully deductive, tightly justified reasoning chains with evidence support.
- 3 points: Acceptable logical leaps, but not rigorously justified.
- 0 points: Broken chains, circular reasoning, or major logical flaws.

3. **Answer Accuracy** (0–5 points):

- Compare generated results to the reference answer on key factual elements.
- + Fully correct: 5 points
- + Partially correct: Score proportionally based on semantic and factual match (e.g., 80% match = 4 points)
- Factually incorrect or misdirected: 0 points (e.g., wrong conclusion, irrelevant content)

【Input Data】

Research Process Record: {process_str}
Generated Answer: {result_str}
Reference Answer: {reference_str}

【Output Requirements】

1. Output must be in JSON format with three evaluation scores.
2. Use the following keys:
 - "Search_Intelligence"
 - "Reasoning_Intelligence"
 - "Result_Accuracy"
3. Append a brief defect analysis (at most 100 words) in both **English and Chinese**.

【Correct Example】

Scoring shows limited search coverage (3/5), reasoning includes unverified assumptions (4/5), and result contains dosage inconsistency (-1).

Defect Analysis:

The main limitations include:

- 1) Lack of recent clinical trials after 2023
- 2) Unverified assumptions about pharmacokinetic parameters

Final Score Output:

```
\boxed{
  "Search_Intelligence": 0,
  "Reasoning_Intelligence": 0,
  "Result_Accuracy": 0,
  "analysis": "Explanation in both English and Chinese..."
}
```

Arka Plan Bilgisi:

Aşağıda derin araştırma senaryosu verilmiştir:

user ile başlayan satırlar, kullanıcı sorularını gösterir.

assistant ile başlayan satırlar, derin araştırma ajanının düşünme sürecini temsil eder.

<think>...</think> ile çevrelenmiş içerik, ajanın içsel muhakemesini yansıtır.

<tool_call>...</tool_call> ile çevrelenmiş içerik, derin araştırma ajanının muhakeme sonrası harici araçları (başlıca: web_search [arama motoru] veya browse_webpage [bir URL'den web sayfası içeriği almak]) nasıl çağırıldığını gösterir.

<tool_response>...</tool_response> ile çevrelenmiş içerik, aracın döndürdüğü sonucu (örneğin arama sonuçları ya da alınan web sayfası içeriği) gösterir.

<answer>...</answer> ile çevrelenmiş içerik, ajanın oluşturduğu nihai cevaptır.

Görev [TASK]:

Sen, derin araştırma ajanından daha zeki olan süperzékâ bir ajan uzmanısın.

Göreviniz, derin araştırma ajanının muhakeme ve araç kullanımını aşağıdaki puanlama kriterlerine göre değerlendirmektir:

【Değerlendirme Boyutları】

1. **Arama Stratejisi Zekâsı** (0–5 puan):

- Kaynak çeşitliliğini, gelişmiş arama sözdizimi kullanımını ve zaman filtresi uygunluğunu değerlendirin.
- 5 puan: Çok platformlu/çok dilli sorgular, Boole mantığı kullanımı, tam eşleşme için tırnak işaretleri vb.
- 4 puan: Anahtar kelime çeşitliliği ve arama turlarında bilgiyi anlamlı şekilde iyileştiren gelişmeler.
- 3 puan: Gelişmiş filtreleme olmaksızın temel anahtar kelime araması.
- 0 puan: Tekrarlayan veya alakasız kaynaklar; geçersiz araç kullanımı.

2. **Mantıksal Muhakeme Kalitesi** (0–5 puan):

- Hipotez oluşturma, kanıt kullanımı ve sonuçların tutarlığını değerlendirin.
- 5 puan: Tam çıkarımsal, kanıt destekli ve sıkıca gereklendirilmiş muhakeme zincirleri.
- 3 puan: Kabul edilebilir mantıksal sıçramalar, ancak titizlikle gereklendirilmemiş.
- 0 puan: Kırık zincirler, döngüsel muhakeme veya ciddi mantık hataları.

3. **Cevap Doğruluğu** (0–5 puan):

- Üretilen sonuçları temel gerçekler açısından referans cevapla karşılaştırın.
- + Tam doğru: 5 puan
- + Kısmen doğru: Anlamsal ve gerçek uyuma göre orantılı puanlama (örneğin, %80 uyum = 4 puan)
- Gerçekten yanlış veya yönlendirici olmayan: 0 puan (örneğin, yanlış sonuç, alakasız içerik)

【Girdi Verisi】

Araştırma Süreci Kaydi: {process_str}
Oluşturulan Yanıt: {result_str}
Referans Yanıt: {reference_str}

【Çıktı Gereksinimleri】

1. Çıktı, üç değerlendirme puanıyla JSON formatında olmalıdır.

2. Aşağıdaki anahtarları kullanın:

- "Search_Intelligence"
- "Reasoning_Intelligence"
- "Result_Accuracy"

3. Kısa bir hata analizi ekleyin (en fazla 100 kelime) hem **İngilizce hem Çince**.

【Doğru Örnek】

Puanlama, sınırlı arama kapsamı gösteriyor (3/5), muhakeme doğrulanmamış varsayımlar içeriyor (4/5) ve sonuç doz tutarsızlığı içeriyor (-1).

Hata Analizi:

Başlıca sınırlamalar şunlardır:

- 1) 2023 sonrası güncel klinik deneylerin olmaması
- 2) Farmakokinetik parametreler hakkında doğrulanmamış varsayımlar

Nihai Skor Çıktısı:

```
\boxed{
  "Search_Intelligence": 0,
  "Reasoning_Intelligence": 0,
  "Result_Accuracy": 0,
  "analysis": "İngilizce ve Çince açıklama..."
}
```

Şekil 7 Düşünce Sürecini değerlendirmek için RRM'ye İstek.

Figure 7 Prompt for RRM to assess the Thought Process.