DeepResearch Arena: The First Exam of LLMs' Research Abilities via Seminar-Grounded Tasks

Haiyuan Wan [1,2] , Chen Yang [3], Junchi Yu [4], Meiqi Tu [5], Jiaxuan Lu [1], Di Yu [1, 2], Jianbao Cao [1, 6], Ben Gao [1, 6], Jiaqing Xie [1], Aoran Wang [1], Wenlong Zhang [1], Philip Torr [4], Dongzhan Zhou [1*]

[1]Shanghai Artificial Intelligence Laboratory
[2]Tsinghua University [3]The Hong Kong University of Science and Technology, Guangzhou [4]University of Oxford
[5]The University of Hong Kong [6]Wuhan University

Abstract

Deep research agents have attracted growing attention for their potential to orchestrate multi-stage research workflows, spanning literature synthesis, methodological design, and empirical verification. Despite these strides, evaluating their research capability faithfully is rather challenging due to the difficulty of collecting frontier research questions that genuinely capture researchers' attention and intellectual curiosity. To address this gap, we introduce ***DeepResearch Arena***, a benchmark grounded in academic seminars that capture rich expert discourse and interaction, better reflecting real-world research environments and reducing the risk of data leakage. To automatically construct DeepResearch Arena, we propose a Multi-Agent Hierarchical Task Generation (MAHTG) system that extracts research-worthy inspirations from seminar transcripts. The MAHTG system further translates research-worthy inspirations into high-quality research tasks, ensuring the traceability of research task formulation while filtering noise. With the MAHTG system, we curate DeepResearch Arena with over 10,000 high-quality research tasks from over 200 academic seminars, spanning 12 disciplines, such as literature, history, and science. Our extensive evaluation shows that DeepResearch Arena presents substantial challenges for current state-of-the-art agents, with clear performance gaps observed across different models.

Introduction

Recent developments in large language models (LLMs) have led to the rise of the deep research agent (Huang et al. 2025; Xu and Peng 2025; Wu et al. 2025), a LLM-powered agentic system designed for research task automation by integrating literature search (Baek et al. 2024), experiment design (Schmidgall et al. 2025), and ideation (Li et al. 2024). Prevailing examples, such as GPT DeepResearch (OpenAI 2025), indicate that deep research agents have great potential to significantly promote research creativity and productivity.

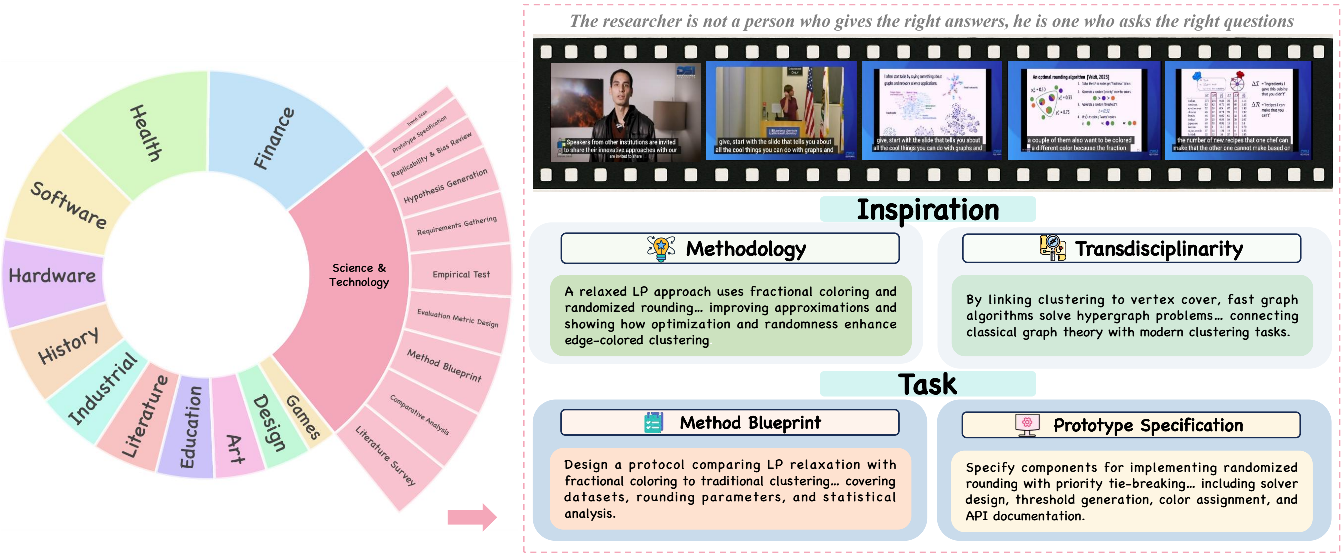
While deep research agents have gained increasing attention (Du et al. 2025), faithfully evaluating their research ability remains a huge challenge. As Einstein once stated, *The formulation of the problem is often more essential than its solution, which may be merely a matter of mathematical or experimental skill* (Einstein and Infeld 1938). This perspective highlights a crucial challenge in formulating high-quality and frontier research tasks to faithfully assess the ability of deep research agents.

Existing benchmarks for deep research agents mainly resort to two approaches to acquire research questions. The first leverages static data corpora such as academic literature and web content, as seen in AcademicBrowse (Zhou et al. 2025a), BrowseComp (Wei et al. 2025), and Researchbench (Liu et al. 2025). The second approach involves manually curated research tasks by domain experts, exemplified by Humanity's Last Exam (Phan et al. 2025), DeepResearchBench (Du et al. 2025), and ExpertLongBench (Ruan et al. 2025). However, both approaches are hindered by critical limitations. Benchmarks derived from static corpora risk data leakage, as the underlying content may already be included in the model pertaining. Meanwhile, datasets curated by experts face scalability bottlenecks and often lack the diversity and spontaneity found in authentic research settings. More fundamentally, both sources tend to abstract away from the situated, evolving nature of real-world research inquiry, where questions emerge dynamically through discourse, ambiguity, and interdisciplinary exploration. A detailed comparison of these benchmarks across key dimensions, including scalability, automation, data leakage risk, and research realism, is provided in Table 1.

To bridge this gap, we introduce a novel benchmark, ***Deep Research Arena***, designed to evaluate deep research agents under authentic, cognitively demanding research scenarios. Unlike static corpora that present information without context, or expert-curated benchmarks that rely on handcrafted tasks detached from actual discovery processes, the proposed benchmark is grounded in academic seminars, where real researchers pose open-ended questions, explore uncertain ideas, and build shared understanding through live discussion. This source captures how real research problems naturally emerge, making Deep Research Arena a more faithful proxy of real-world inquiry. Furthermore, seminar videos are rarely included in model pretraining, which significantly reduces the risk of data leakage that commonly affects benchmarks derived from literature or web corpora.

To capture the nature of such authentic inquiry, Deep Research Arena formulates tasks as open-ended, under-defined problems, drawn by the theory of Ill-Structured Problem

[*] Corresponding author
Under Review


Health
Finance
Software
Hardware
History
Industrial
Literature
Education
Art
Design
Games
Science & Technology
Trend Scan
Prototype Specification
Replicability & Bias Review
Hypothesis Generation
Requirements Gathering
Empirical Test
Evaluation Metric Design
Method Blueprint
Comparative Analysis
Literature Survey
The researcher is not a person who gives the right answers, he is one who asks the right questions
Speakers from other institutions are invited to share their innovative approaches with our
give, start with the slide that tells you about all the cool things you can do with graphs and
give, start with the slide that tells you about all the cool things you can do with graphs and
a couple of them also want to be colored a different color because the fraction
the number of new recipes that one chef can make that the other one cannot make based on
Inspiration
Methodology
Transdisciplinarity
Task
Method Blueprint
Prototype Specification

A relaxed LP approach uses fractional coloring and randomized rounding… improving approximations and showing how optimization and randomness enhance edge-colored clustering

By linking clustering to vertex cover, fast graph algorithms solve hypergraph problems… connecting classical graph theory with modern clustering tasks.

Design a protocol comparing LP relaxation with fractional coloring to traditional clustering… covering datasets, rounding parameters, and statistical analysis.

Specify components for implementing randomized rounding with priority tie-breaking… including solver design, threshold generation, color assignment, and API documentation.

Figure 1: **Overview of seminar domains and task structures in MAHTG. Left:** Distribution of academic seminars across diverse domains such as Science & Technology, Health, Finance, and others. The outer arc further decomposes each domain into representative research tasks. For instance, Science & Technology includes tasks such as *Hypothesis Generation*, *Empirical Test*, *Prototype Specification*, and *Trend Scan*. **Right:** Illustration of MAHTG's multi-agent pipeline, where seminar content is transformed into structured research tasks via intermediate inspirations (e.g., *Methodology*, *Transdisciplinarity*). Example outputs are shown for both stages.

Solving (Jonassen 1997), which describes real-world problems as "poorly defined, with no single correct formulation and no objective evaluation criteria". To construct the Deep Research Arena, we develop a Multi-Agent Hierarchical Task Generation (MAHTG) system that automatically extracts research-worthy inspirations and systematically transforms them into high-quality, traceable research tasks through a multi-stage filtering and structuring pipeline. This design ensures both the authenticity and reproducibility of task construction, while reducing noise and preserving the intellectual context of original expert discourse.

We curate a large-scale, multidisciplinary seminar dataset, constructing over 10,000 structured tasks spanning core research competencies. Building on this, we develop a hybrid evaluation framework that jointly measures factual grounding and higher-order reasoning, with examples shown in Figure 1. Together, these contributions provide a rigorous and theory-aligned foundation for assessing deep research competence in deep research agents.

Our contributions are threefold:

- **Seminar-grounded data collection.** We curate a corpus of over 200 academic seminars across 12 disciplines, encompassing real-world expert discourse across science, engineering, humanities, and the arts.
- **Hierarchical task generation.** A multi-stage agent framework extracts research-worthy inspirations from seminar transcripts, categorized into *Limitation*, *Methodology*, *Transdisciplinarity*, and *Hypothesis*, and transforms them into over 10,000 open-ended tasks aligned with the canonical research stages of *Synthesis*, *Design*, and *Evaluation*.
- **Hybrid evaluation framework.** We employ two complementary metrics to quantify factual alignment via extracted keypoints and evaluate open-ended reasoning using adaptively generated, rubric-based checklists.

Related Works

Deep Research Agents. The emergence of DR agents builds upon recent advances in LLMs equipped with tool-use capabilities (Li et al. 2025; Qu et al. 2025; Tang et al. 2023), which allow models to interface with search engines, code interpreters, and external APIs to extend their reasoning horizon. On this foundation, systems such as GPT Deep Research (OpenAI 2025), Gemini Deep Research (Google 2025), and Grok DeepSearch (xAI 2025) have been developed to support multi-stage research workflows. GPT's system focuses on outline-driven long-form synthesis with citation grounding, Gemini emphasizes multimodal retrieval and synthesis, while Grok prioritizes real-time web summarization for dynamic topics. These agents reflect a shift from retrieval-based assistants to goal-directed, tool-augmented agents capable of supporting exploratory, open-ended inquiry (Yu, He, and Ying 2023).

Benchmarks for Deep Research Agents. Existing benchmarks for deep research agents mainly resort to two approaches to acquire research questions: automatically deriving tasks from static corpora or manually curating them through expert design. The first leverages static data corpora such as papers, and web documents to construct benchmarks represented by multi-hop reasoning or simplified scientific queries. Examples include MuSiQue (Trivedi et al.

Benchmark	**Data Source**	**Scalability**	**Risk of Data Leakage**	**Task Automation**	**Research Realism**
ScholarSearch	Literature	✓	✓	✗	✗
BrowseComp	Web Corpus	✓	✓	✓	✗
ResearchBench	Literature	✓	✓	✓	✗
Humanity's Last Exam	Expert	✗	✓	✗	✓
DeepResearchBench	Expert	✗	✓	✗	✓
ExpertLongBench	Expert	✗	✓	✗	✓
DeepResearch Arena (Ours)	Seminar Discourse	✓	✗	✓	✓

Table 1: Comparison of existing deep research benchmarks and our ***DeepResearch Arena*** along key dimensions.

2022), which automatically generates multi-hop questions by linking single-hop QA pairs from existing datasets, and HotpotQA (Yang et al. 2018), where annotators write questions guided by system-selected Wikipedia article pairs, making the process closer to extraction than genuine question generation. Other benchmarks in this category include StrategyQA (Geva et al. 2021), ThoughtSource (Ott et al. 2023), AcademicBrowse (Zhou et al. 2025a), and BrowseComp (Wei et al. 2025). Despite their emphasis on multi-step reasoning, these benchmarks rely on manually constructed logic chains with predefined paths. They primarily test factual retrieval and compositional reasoning capabilities, yet fail to capture how research questions naturally emerge, evolve, and iterate in real-world research contexts. ScienceQA (Lu et al. 2022) is a large-scale multimodal multiple-choice science QA benchmark (21K questions across STEM and social/language science) that includes lecture and explanation-level CoT annotations to support interpretable multi-step reasoning.

The second category consists of expert-authored benchmarks, where researchers collaborate with domain specialists to construct high-quality, PhD-level evaluation tasks. Compared to benchmarks built from static corpora, these datasets typically feature more original, conceptually challenging, and discipline-specific questions that better reflect expert-level reasoning. Representative examples include LAB-Bench (Laurent et al. 2024), ARC (Clark et al. 2018), GPQA (Rein et al. 2024), FrontierMath (Glazer et al. 2024), and Humanity's Last Exam (Phan et al. 2025). GPQA provides graduate-level multiple-choice questions in biology, physics, and chemistry, curated and verified by domain PhDs to ensure they cannot be solved via surface-level heuristics or web search. Humanity's Last Exam comprises a collection of open-ended, expert-written research questions across disciplines such as history, philosophy, and theoretical science, designed to probe creative, integrative thinking under minimal structural constraints. DeepResearch Bench (Du et al. 2025) moves toward more realistic simulation by requiring long-form research reports across disciplines. However, this entire class of expert-authored benchmarks faces several limitations: their prompts are manually constructed, which restricts scalability and diversity, and the datasets remain relatively small in size. More fundamentally, they also fail to capture how research questions emerge dynamically through discourse, ambiguity, and interdisciplinary exploration—core characteristics of authentic research practice.

Multi-Agent Hierarchical Task Generation.

Data Collection. To support the construction of research tasks grounded in authentic scholarly practice, we curated a diverse corpus of over 200 academic seminar videos spanning 12 disciplines, contributed by PhD-level researchers and sourced from publicly accessible academic seminar recordings spanning multiple disciplines. Each video is knowledge-dense and typically lasts around or over 1 hour, and the disciplinary distribution of this corpus is illustrated in Figure 2. Seminar recordings preserve the full contextual flow of expert discourse, encompassing how researchers synthesize prior knowledge, design new approaches, and evaluate outcomes. In this way, they offer a rich context for task generation. Compared to static corpora such as Wikipedia or scientific articles, seminar data captures dynamic and authentic interactions among scholars, reflecting the iterative and evolving nature of real-world research.

As a first step in processing the raw seminar videos, we extract the audio and convert it into textual transcripts with automatic speech recognition. The resulting transcripts retain the full semantic content of the original recordings while remaining absent from existing LLM pretraining corpora, thereby reducing the risk of data contamination and ensuring the integrity of task construction.

Inspiration Extraction. Based on seminar transcripts, **Inspira Agent** automatically extracts *inspirations* (as illustrated in Table 2) from seminar transcripts, transforming unstructured expert discourse into structured units suitable for downstream research task construction. To identify academically valuable content, the agent evaluates candidate segments along four dimensions: Novelty, Explorability, Challenge, and Verifiability. Each selected inspiration must satisfy at least two of these criteria. This multi-dimensional filtering process enables the agent to effectively suppress irrelevant or redundant material, reorganize latent research signals, and produce outputs with clearer logical structure and sharper thematic focus, thereby enhancing their suitability for subsequent task generation. In addition, the agent categorizes each item based on its informational focus into one of four types: *Limitation*, *Methodology*, *transdisciplinarity*, *Hypothesis*, as illustrated in Table 2, representing testable claims that can be empirically verified.

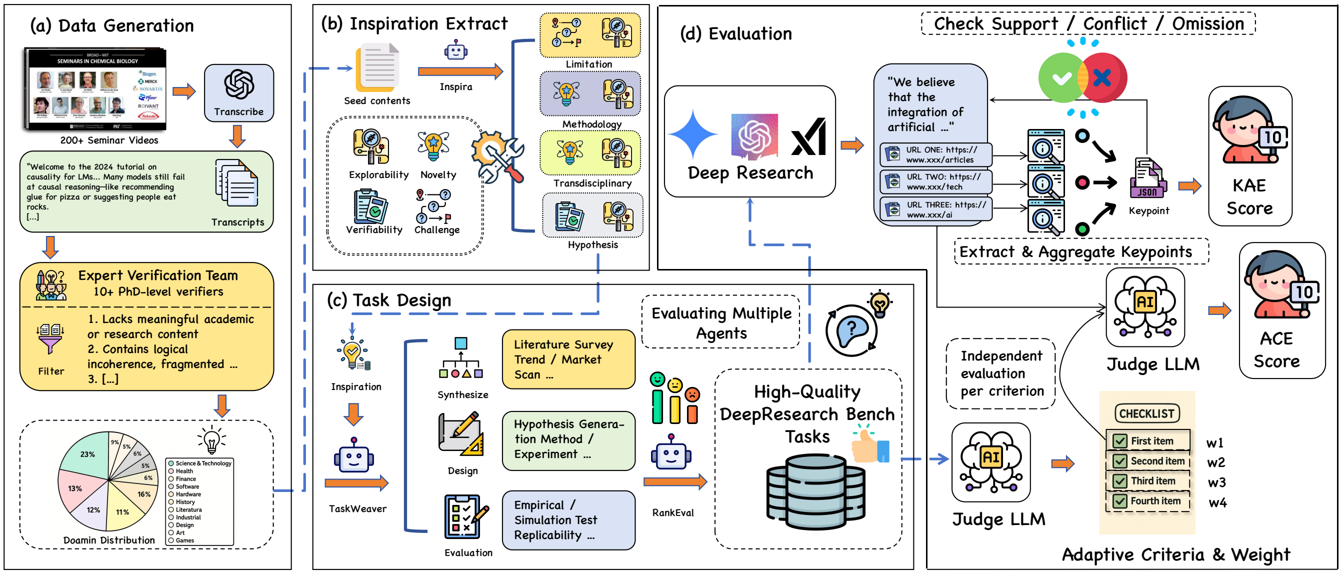

(a) Data Generation
BROAD—MIT SEMINARS IN CHEMICAL BIOLOGY
200+ Seminar Videos
Transcribe
"Welcome to the 2024 tutorial on causality for LMs... Many models still fail at causal reasoning—like recommending glue for pizza or suggesting people eat rocks.
[...]
Transcripts
Expert Verification Team
10+ PhD-level verifiers
Filter
1. Lacks meaningful academic or research content
2. Contains logical incoherence, fragmented ...
3. [...]
23%
9%
5%
6%
5%
6%
16%
11%
12%
13%
Science & Technology
Health
Finance
Software
Hardware
History
Literatura
Industrial
Design
Art
Games
Doamin Distribution
(b) Inspiration Extract
Seed contents
Inspira
Explorability
Novelty
Verifiability
Challenge
Limitation
Methodology
Transdisciplinary
Hypothesis
(c) Task Design
Inspiration
TaskWeaver
Synthesize
Design
Evaluation
Literature Survey Trend / Market Scan ...
Hypothesis Generation Method / Experiment ...
Empirical / Simulation Test Replicability ...
Evaluating Multiple Agents
RankEval
High-Quality DeepResearch Bench Tasks
(d) Evaluation
Deep Research
Check Support / Conflict / Omission
"We believe that the integration of artificial ..."
URL ONE: https:// www.xxx/articles
URL TWO: https:// www.xxx/tech
URL THREE: https:// www.xxx/ai
JSON
Keypoint
KAE Score
Extract & Aggregate Keypoints
Judge LLM
ACE Score
Independent evaluation per criterion
Judge LLM
CHECKLIST
First item
Second item
Third item
Fourth item
w1
w2
w3
w4
Adaptive Criteria & Weight

Figure 2: Overview of our benchmark construction pipeline, including four stages: (a) Data generation from transcribed seminar videos, (b) extraction of research inspirations, (c) multi-phase task design, and (d) evaluation using both KAE and ACE metrics.

Task Generation and Filtering. Building on the structured inspirations extracted from seminar transcripts, we deploy **TaskWeaver Agent** that aggregates and reorganizes content across multiple inspirations to synthesize a focused set of concrete research tasks distributed across three key phases—*Synthesize*, *Design*, and *Evaluate*, as illustrated in Figure 2. These tasks are constructed by identifying the core problem focus or methodological cues within the inspirations and are paired with clearly defined, executable goals. This structured synthesis process enables the scalable construction of diverse, high-quality DeepResearch tasks aligned with the demands of real-world scientific inquiry (Yu et al. 2022).

To rank the quality of research tasks, we adopt **RankEval Agent** based on the Elo rating system (Glickman 1995). Each task is initialized with a rating score of 1200. In each round, we randomly sample disjoint pairs of tasks and compare them based on evaluation criteria such as originality, clarity, and scientific relevance. Given a pair of tasks t_a and t_b with current Elo scores r_a and r_b, we first compute the expected winning probabilities using:

$$e_a = \frac{1}{1 + 10^{(r_b - r_a)/400}}, \qquad (1)$$

where $e_b = 1 - e_a$. An evaluator determines which task is preferred, along with a confidence score $C \in [0.5, 1.0]$. Based on this, we assign soft outcomes:

$$s_a = C, \quad s_b = 1 - C \qquad (2)$$

We then update the Elo scores using the following update rule:

$$r'_a = r_a + K \cdot (s_a - e_a), \quad r'_b = r_b + K \cdot (s_b - e_b) \quad (3)$$

where K is a tunable constant controlling the update magnitude, set to $K = 32$ in our implementation. This procedure is repeated over R rounds of comparisons (e.g., $R = 2$), allowing the scores to stabilize. After all rounds, we select the top K tasks with the highest Elo scores as the final outputs.

Evaluation Methodology

To comprehensively assess the capabilities of deep research agents in research-oriented tasks, we propose a hybrid evaluation framework that integrates both objective and subjective dimensions of performance. Traditional benchmarks often focus narrowly on surface-level accuracy or retrieval metrics, failing to capture the nuanced reasoning, creativity, and methodological rigor required for real-world research. In contrast, our framework disentangles these facets by combining (1) Keypoint-Aligned Evaluation (KAE) to measure factual correctness and grounding against reference materials, and (2) Adaptively-generated Checklist Evaluation (ACE) to score open-ended outputs via fine-grained, model-adaptive rubrics. This dual approach enables multi-perspective assessment across all stages of the research workflow, from literature synthesis to hypothesis generation and empirical validation, offering a more faithful estimate of models' deep research competence.

Keypoint-Aligned Evaluation. To evaluate the factual adequacy of model-generated research reports in a reference-grounded and scalable manner, we propose a structured KAE pipeline.

Let R denote a model-generated report, and let $URL(R)$ represent the set of all cited URLs in R. For each URL $u \in URL(R)$, we retrieve the underlying webpage content and extract its factual keypoints using a keypoint extraction function $Extract(u)$:

$$K_u = Extract(u) \qquad (4)$$

We then aggregate the keypoints from all cited sources into

Term	Illustration	Example
I. Core Unit: Inspiration		
Inspiration	A research-worthy idea distilled from academic discourse, exhibiting at least two of: *novelty*, *explorability*, *challenge*, *verifiability*. Serves as the seed for task generation.	"A greedy maximal independent-set algorithm ... achieves a 2-approximation in O (sum of hyperedge sizes) time ... shows classical graph methods can solve edge-colored hypergraph clustering without auxiliary graphs."
II. Types of Inspiration		
Limitation	An open problem, deficiency, or bottleneck in existing methods.	"Few models handle transdisciplinary seminar reasoning."
Methodology	A new or adapted approach, pipeline, or tool.	"Introduce retrieval-augmented reranking framework."
Transdisciplinarity	Ideas involving the fusion of theories or tools across disciplines.	"Apply ecological network theory to social dynamics"
Hypothesis	A testable proposition that guides design or evaluation.	"Grounded citations improve factual accuracy."
III. Task Phase Labels		
Synthesize	Collecting, integrating, and analyzing prior work to form direction.	"Identify gaps in seminar-based QA literature."
Design	Designing solutions, models, or experiments to address a problem.	"Propose a multimodal tree-search method."
Evaluate	Assessing results using structured criteria or benchmarks.	"Compare keypoint coverage across baselines."

Table 2: Core terminology used in our benchmark, grouped into inspiration, its subtypes, and research task phases. This table standardizes interpretation of key concepts throughout the paper.

a unified, de-duplicated list of keypoints, which we term the Unified Evidence Keypoints (UEK):

$$\text{UEK} = \mathit{Dedup}\left(\bigcup_{u \in URL(R)} K_u\right) \tag{5}$$

Given this set of reference keypoints, we evaluate the report R along three dimensions:

(1) Keypoint Supported Rate (KSR): the proportion of keypoints from UEK that are explicitly covered or supported in the report:

$$\text{KSR}(R) = \frac{|\mathit{Supported}(R, \text{UEK})|}{|\text{UEK}|} \tag{6}$$

(2) Keypoint Conflict Rate (KCR): the proportion of keypoints from UEK that are contradicted by content in the report:

$$\text{KCR}(R) = \frac{|\mathit{conflict}(R, \text{UEK})|}{|\text{UEK}|} \tag{7}$$

(3) Keypoint Omission Rate (KOR): the proportion of keypoints from UEK that are omitted by content in the report:

$$\text{KCR}(R) = \frac{|\mathit{Omitted}(R, \text{UEK})|}{|\text{UEK}|} \tag{8}$$

Ideally, a high-quality research report should achieve a high KSR (indicating comprehensive factual coverage) and a low KCR and KOR (indicating consistency with evidence). These metrics enable interpretable, reference-grounded evaluation of factual alignment.

Adaptively-generated Checklist Evaluation. To address the challenges of evaluating open-ended research tasks that lack fixed reference answers, we introduce Adaptively-generated Checklist Evaluation (ACE), a two-stage evaluation protocol that leverages the analytical capabilities of large language models (LLMs) while mitigating common sources of bias and inconsistency.

In the first stage, we use a high-capacity LLM (e.g., GPT-4o) to perform meta-analysis over the task prompt, generating a customized checklist of evaluation criteria tailored to the query. Each checklist item corresponds to a critical evaluation dimension, such as factual correctness, methodological soundness, formatting, or reasoning clarity, and is assigned a normalized weight to reflect its relative importance. This step serves to concretize abstract judgment into discrete, model-understandable subgoals.

In the second stage, a separate LLM is tasked with scoring the model-generated response against the checklist. For each item, the evaluator model independently assesses whether the response satisfies the criterion and assigns a local score. These individual scores are then aggregated via a weighted average to produce a final task-level rating. By decoupling checklist generation from scoring, ACE reduces evaluation bias, especially those arising from the evaluator's limited comprehension or heuristic shortcuts.

ACE addresses key limitations of existing evaluation paradigms. Human evaluation, while often considered the gold standard, suffers from subjectivity, inter-annotator inconsistency, and high cost. LLM-as-a-judge methods, especially when using smaller models, struggle with complex query understanding, detailed analytical reasoning, and accurate interpretation. Furthermore, rubric-based methods ei-

Model	KAE						ACE		Avg. Token (k)		Avg. references	
	KSR		KCR		KOR							
gpt-4o-search-preview	50.0	**85.0**	8.9	5.0	41.1	10.0	2.41	2.00	1.21	2.85	4.24	3.49
gpt-4o-mini-search-preview	78.7	55.6	8.5	16.7	12.8	27.8	2.23	2.05	**1.07**	2.23	3.83	2.07
gpt-4.1-mini w/search	62.5	76.5	10.9	**5.9**	26.6	17.6	2.21	1.87	1.10	**2.02**	4.75	2.39
gpt-4.1 w/search	77.8	60.6	**2.8**	6.1	19.4	33.3	2.43	2.22	1.20	2.43	3.51	2.44
o4-mini-deepresearch	77.2	75.8	4.3	18.2	18.5	6.1	**4.03**	3.88	5.59	12.5	**29.66**	**37.27**
gemini-2.5-pro w/search	65.1	76	14.3	12	20.6	12	2.97	**4.03**	4.29	9.14	23.86	21.39
gemini-2.5-flash w/search	78.7	80	3.4	16	18	**4**	3.81	3.58	64.09	19.78	29.54	28.07
grok-4 w/search	**83.3**	50	7.5	13.8	**9.2**	36.2	2.97	2.97	3.16	6.60	20.59	19.95

Table 3: Evaluation metrics across models. The model release dates are omitted for brevity. Each column reports two values, with the left representing the evaluation results on the English task and the right on the Chinese task.

ther rely on static reference answers, which are unsuitable for open-ended tasks, or require hand-crafted criteria that are difficult to scale and generalize. In contrast, ACE provides a flexible, scalable, and more reliable alternative for nuanced research task evaluation.

Experiments

Implementation Details. Our MAHTG system comprises several specialized agents, each responsible for a distinct stage in transforming raw academic seminars into structured research tasks and evaluations.

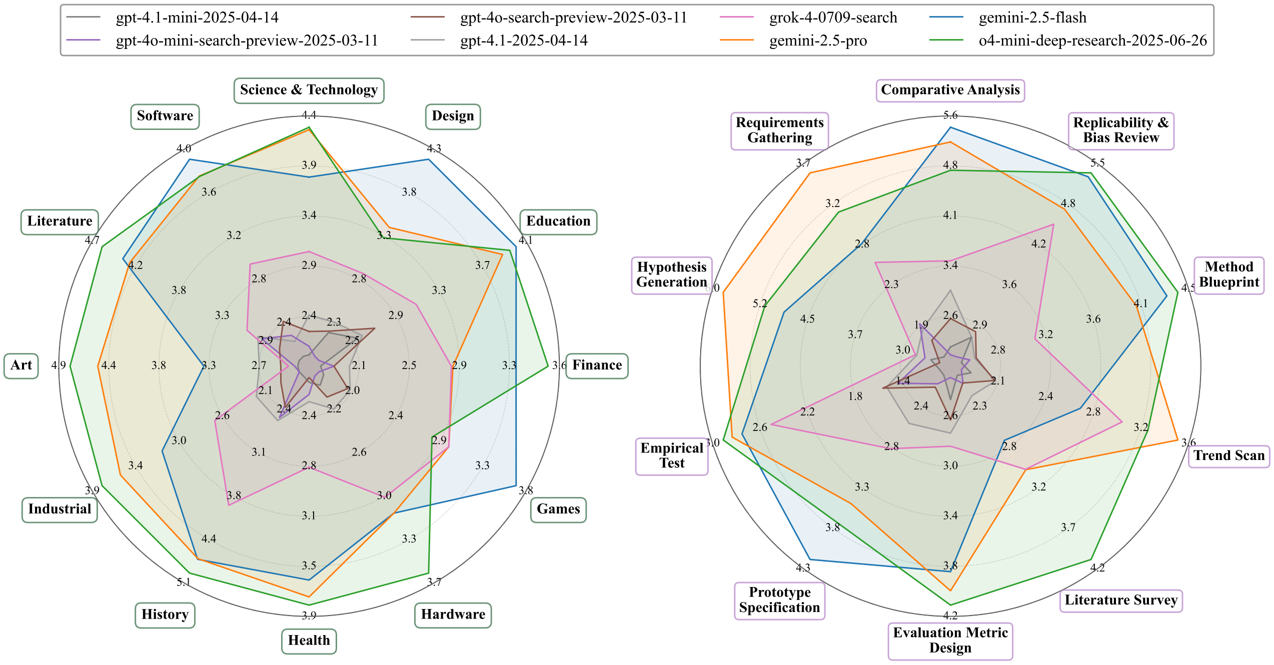
Model Selection Rationale. We adopt a *heterogeneous model configuration* across the MAHTG system, guided by three principles: (1) *capability-task alignment*, assigning models suited to each agent's functional role; (2) *cost-effectiveness and scalability*, ensuring efficiency over large-scale data; and (3) *robustness through model diversity*, mitigating systemic bias. Large models like *claude-sonnet-4-20250514* are used for structured reasoning and code-like outputs, while lightweight ones like *gpt-4o-mini* support tasks requiring relative preference. The Inspira Agent adopts *claude-sonnet-4-20250514* for its strong long-context handling and structured generation. The same model powers the TaskWeaver Agent to ensure schema consistency in transforming inspirations into structured tasks. For efficient pairwise evaluation, the RankEval Agent uses *gpt-4o-mini*, balancing accuracy and cost under the ELO-based framework. To reduce costs, we selected the top 100 highest-scoring samples from the full dataset for evaluation. The choices for LLM align with human-like action and are empirically validated (see appendix).

We use *gemini-2.5-flash* as a unified evaluator for both factual and subjective scoring, leveraging its strong instruction-following and long-context reasoning. In KAE, it extracts key factual statements from sources retrieved via the Jina AI API and determines whether each is supported, contradicted, or omitted. In ACE, it generates detailed, task-specific checklists and conducts criterion-based evaluation. This setup ensures consistency across evaluation stages while maintaining precision, scalability, and interpretability.

Evaluated Models. We evaluate a diverse suite of large language models covering both frontier-level deep research agents and models augmented with real-time retrieval capabilities. Specifically, we include *gpt-4o-search-preview-2025-03-11*, *gpt-4o-mini-search-preview-2025-03-11*, *gpt-4.1-2025-04-14 w/search*, *gpt-4.1-mini-2025-04-14 w/search*, *o4-mini-deepresearch-2025-06-26*, *gemini-2.5-pro w/search*, *gemini-2.5-flash w/search*, and *grok-4-0709 w/search*. When referring to these models in the future, abbreviations will be used, ignoring with search and time versions.

Overall Performance. The table 3reveals clear differences in both ACE and KCE across models. The best ACE performance is achieved by *gpt-o4-mini-deep-research*, which combines the highest ACE score of 4.03 with strong KAE metrics, demonstrating accurate, well-structured, and comprehensive outputs. *GPT-4.1* excels in factual precision but falls short in subjective quality, with the lowest KCR. It minimizes factual errors, yet its lower ACE scores suggest limited coherence and depth. *Gemini-2.5-flash* also performs strongly, with relatively high factual coverage and low contradiction and omission, though it uses significantly more tokens than any other model, indicating a trade-off between thoroughness and efficiency. In contrast, *gpt-4o-search-preview*and *gpt-4o-mini-search-preview* use far fewer tokens but do not perform so well in both evaluation dimensions, suggesting limited ability to handle complex research tasks. *grok-4* demonstrates the strongest factual grounding on English tasks (KSR 83.3), but its performance drops sharply in Chinese, with significantly lower coverage and higher omission. This highlights its limited multilingual generalization despite strong English capabilities. Overall, the results reflect varying model strengths, with some excelling in precision and others in depth or efficiency.

Performance on Different Tasks. As shown in Figure 3, the ACE-based subjective evaluation reveals substantial differences in how models perform across various research task types. Models like *gpt-o4-mini-deepresearch* and *gemini-2.5-flash* demonstrate consistently strong performance across nearly all tasks, especially excelling in complex and high-level tasks such as hypothesis genera-



(a) Performance by Disciplines

(b) Performance by Tasks

Figure 3: Comparison of current mainstream models on the DeepResearch Arena benchmark. (a) Performance across 12 research disciplines (*e.g.*, Science & Technology, Art, Finance). (b) Performance across 10 research task types (*e.g.*, Hypothesis Generation, Method Blueprint, Evaluation Metric Design), highlighting task-specific capabilities.

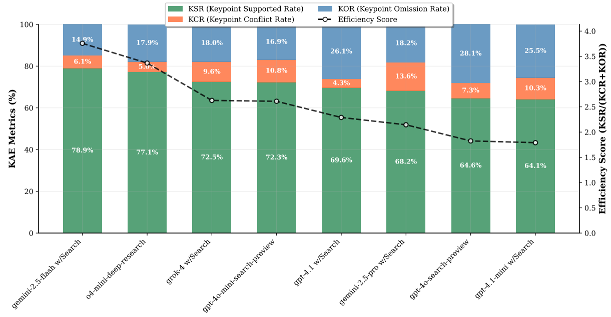


Figure 4: Comparison of DeepResearch agents in terms of Keypoint-Aligned Evaluation (KAE) metrics and efficiency.

tion, evaluation metric design, and methodological planning. *Gemini-2.5-pro* also shows well-rounded capabilities, performing reliably in tasks that require comparative analysis and methodological reasoning. The *gpt-4o* family, particularly the mini version, performs poorly across most task types, struggling especially with tasks that require multi-step logic and structured outputs. These differences highlight each model's unique strengths and limitations, underscoring the importance of task-specific evaluation in assessing deep research competence.

Models also show clear differences in task performance under the KAE as shown in Figure 4. *Gemini-2.5-flash* and *gpt-o4-mini-deepresearch* achieve the strongest overall results, with high keypoint coverage and low conflict and omission rates, leading to the highest efficiency scores. In contrast, *gemini-2.5-pro*, *gpt-4o-search-preview*, and *gpt-4.1-mini* struggle with higher conflict and omission rates, resulting in the lowest efficiency and limited reliability for fact-intensive generation. Overall, the results highlight substantial differences in how models handle task complexity and factual alignment, underscoring the value of KAE for fine-grained evaluation of research capabilities.

The experiment for effectiveness of our benchmark to prevent data leakage is detailed in the appendix.

Conclusion

We present the ***DeepResearch Arena***, a novel benchmark for evaluating the deep research capabilities of large language models in realistic, open-ended settings. Grounded in cognitive theories and authentic seminar discourse, DeepResearch Arena captures the contextual complexity and methodological ambiguity of real-world research. It systematically assesses LLM-based agents across three essential stages, through a curated corpus of multidisciplinary seminars, a hierarchical task generation pipeline, and a hybrid evaluation protocol measuring both factual grounding and higher-order reasoning. By bridging the gap between retrieval-centric agent design and cognitively demanding research tasks, it offers a rigorous, theory-aligned foundation for advancing next-generation research assistants.

References

Baek, J.; Jauhar, S. K.; Cucerzan, S.; and Hwang, S. J. 2024. Researchagent: Iterative research idea generation over scientific literature with large language models. *arXiv preprint arXiv:2404.07738*.

Carlini, N.; Tramer, F.; Wallace, E.; Jagielski, M.; Herbert-Voss, A.; Lee, K.; Roberts, A.; Brown, T.; Song, D.; Erlingsson, U.; et al. 2021. Extracting training data from large language models. In *30th USENIX security symposium (USENIX Security 21)*, 2633–2650.

Castaño, J.; Gambarte, M. L.; Park, H. J.; del Pilar Avila Williams, M.; Pérez, D.; Campos, F.; Luna, D.; Benítez, S.; Berinsky, H.; and Zanetti, S. 2016. A Machine Learning Approach to Clinical Terms Normalization. In Cohen, K. B.; Demner-Fushman, D.; Ananiadou, S.; and Tsujii, J.-i., eds., *Proceedings of the 15th Workshop on Biomedical Natural Language Processing*, 1–11. Berlin, Germany: Association for Computational Linguistics.

Clark, P.; Cowhey, I.; Etzioni, O.; Khot, T.; Sabharwal, A.; Schoenick, C.; and Tafjord, O. 2018. Think You Have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *arXiv preprint arXiv:1803.05457*.

De Boom, C.; Van Canneyt, S.; Bohez, S.; Demeester, T.; and Dhoedt, B. 2015. Learning Semantic Similarity for Very Short Texts. In *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*, 1229–1234. IEEE.

Du, M.; Xu, B.; Zhu, C.; Wang, X.; and Mao, Z. 2025. Deep-Research Bench: A Comprehensive Benchmark for Deep Research Agents. *arXiv preprint*.

Einstein, A.; and Infeld, L. 1938. *The Evolution of Physics*. Simon and Schuster.

Geva, M.; Khashabi, D.; Segal, E.; Khot, T.; Roth, D.; and Berant, J. 2021. Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies. *Transactions of the Association for Computational Linguistics*, 9: 346–361.

Glazer, E.; Erdil, E.; Besiroglu, T.; Chicharro, D.; Chen, E.; Gunning, A.; Olsson, C. F.; Denain, J.-S.; Ho, A.; de Oliveira Santos, E.; Järviniemi, O.; Barnett, M.; Sandler, R.; Vrzala, M.; Sevilla, J.; Ren, Q.; Pratt, E.; Levine, L.; Barkley, G.; Stewart, N.; Grechuk, B.; Grechuk, T.; Enugandla, S. V.; and Wildon, M. 2024. FrontierMath: A Benchmark for Evaluating Advanced Mathematical Reasoning in AI. arXiv:2411.04872.

Glickman, M. E. 1995. A comprehensive guide to chess ratings. *American Chess Journal*, 3.

Google. 2025. Deep Research is now available on Gemini 2.5 Pro Experimental. Gemini Blog (online). Gemini Advanced subscribers can use Deep Research powered by Gemini 2.5 Pro Experimental.

Huang, Y.; Chen, Y.; Zhang, H.; Li, K.; Fang, M.; Yang, L.; Li, X.; Shang, L.; Xu, S.; Hao, J.; Shao, K.; and Wang, J. 2025. Deep Research Agents: A Systematic Examination And Roadmap. arXiv:2506.18096.

Jindal, S.; and Leema, M. 2024. A Survey of Text Similarity Approaches. *Journal of Artificial Intelligence and Capsule Networks*, 4(1): 33–45.

Jonassen, D. H. 1997. Instructional design models for well-structured and ill-structured problem-solving learning outcomes. *Educational Technology Research and Development*, 45(1): 65–94.

Kendall, M. G. 1938. A New Measure of Rank Correlation. *Biometrika*, 30(1-2): 81–93.

Laurent, J. M.; Janizek, J. D.; Ruzo, M.; Hinks, M. M.; Hammerling, M. J.; Narayanan, S.; Ponnapati, M.; White, A. D.; and Rodriques, S. G. 2024. LAB-Bench: Measuring Capabilities of Language Models for Biology Research. arXiv:2407.10362.

Lehman, E.; Jain, S.; Pichotta, K.; Goldberg, Y.; and Wallace, B. C. 2021. Does BERT pretrained on clinical notes reveal sensitive data? *arXiv preprint arXiv:2104.07762*.

Li, L.; Xu, W.; Guo, J.; Zhao, R.; Li, X.; Yuan, Y.; Zhang, B.; Jiang, Y.; Xin, Y.; Dang, R.; et al. 2024. Chain of ideas: Revolutionizing research via novel idea development with llm agents. *arXiv preprint arXiv:2410.13185*.

Li, W.; Li, D.; Dong, K.; Zhang, C.; Zhang, H.; Liu, W.; Wang, Y.; Tang, R.; and Liu, Y. 2025. Adaptive Tool Use in Large Language Models with Meta-Cognition Trigger. arXiv:2502.12961.

Liu, Y.; Yang, Z.; Xie, T.; Ni, J.; Gao, B.; Li, Y.; Tang, S.; Ouyang, W.; Cambria, E.; and Zhou, D. 2025. Researchbench: Benchmarking llms in scientific discovery via inspiration-based task decomposition. *arXiv preprint arXiv:2503.21248*.

Lu, P.; Mishra, S.; Xia, T.; Qiu, L.; Chang, K.-W.; Zhu, S.-C.; Tafjord, O.; Clark, P.; and Kalyan, A. 2022. Learn to Explain: Multimodal Reasoning via Thought Chains for Science Question Answering. arXiv:2209.09513.

OpenAI. 2025. Introducing Deep Research. https://cdn.openai.com/API/docs/deep_research_blog.pdf?utm_source=chatgpt.com. Accessed July 30, 2025.

Ott, S.; Hebenstreit, K.; Liévin, V.; Hother, C. E.; Moradi, M.; Mayrhauser, M.; Praas, R.; Winther, O.; and Samwald, M. 2023. ThoughtSource: A central hub for large language model reasoning data. *Scientific Data*, 10(1).

Öztürk, H.; Ozkirimli, E.; and Özgür, A. 2016. A comparative study of SMILES-based compound similarity functions for drug-target interaction prediction. *BMC bioinformatics*, 17(1): 128.

Pearson, K. 1895. Note on Regression and Inheritance in the Case of Two Parents. *Proceedings of the Royal Society of London*, 58: 240–242.

Phan, L.; Gatti, A.; Han, Z.; Li, N.; Hu, J.; Zhang, H.; and et al. 2025. Humanity's Last Exam. arXiv:2501.14249.

Qu, C.; Dai, S.; Wei, X.; Cai, H.; Wang, S.; Yin, D.; Xu, J.; and Wen, J.-r. 2025. Tool learning with large language models: a survey. *Frontiers of Computer Science*, 19(8).

Rein, D.; Hou, B. L.; Stickland, A. C.; Petty, J.; Pang, R. Y.; Dirani, J.; Michael, J.; and Bowman, S. R. 2024. GPQA: A Graduate-Level Google-Proof Q&A Benchmark. In *First Conference on Language Modeling*.

Ruan, J.; Nair, I.; Cao, S.; Liu, A.; Munir, S.; Pollens-Dempsey, M.; Chiang, T.; Kates, L.; David, N.; Chen, S.;

et al. 2025. ExpertLongBench: Benchmarking Language Models on Expert-Level Long-Form Generation Tasks with Structured Checklists. *arXiv preprint arXiv:2506.01241*.

Schmidgall, S.; Su, Y.; Wang, Z.; Sun, X.; Wu, J.; Yu, X.; Liu, J.; Moor, M.; Liu, Z.; and Barsoum, E. 2025. Agent Laboratory: Using LLM Agents as Research Assistants. arXiv:2501.04227.

Spearman, C. 1904. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1): 72–101.

Tang, Q.; Deng, Z.; Lin, H.; Han, X.; Liang, Q.; Cao, B.; and Sun, L. 2023. ToolAlpaca: Generalized Tool Learning for Language Models with 3000 Simulated Cases. arXiv:2306.05301.

Trivedi, H.; Balasubramanian, N.; Khot, T.; and Sabharwal, A. 2022. MuSiQue: Multihop Questions via Single-hop Question Composition. arXiv:2108.00573.

Wang, J.; and Dong, Y. 2020. Measurement of Text Similarity: A Survey. *Information*, 11(9).

Wei, J.; Sun, Z.; Papay, S.; McKinney, S.; Han, J.; Fulford, I.; Chung, H. W.; Passos, A. T.; Fedus, W.; and Glaese, A. 2025. Browsecomp: A simple yet challenging benchmark for browsing agents. *arXiv preprint arXiv:2504.12516*.

Wu, J.; Zhu, J.; Liu, Y.; Xu, M.; and Jin, Y. 2025. Agentic Reasoning: A Streamlined Framework for Enhancing LLM Reasoning with Agentic Tools. arXiv:2502.04644.

xAI. 2025. Grok 3. https://x.ai/news/grok-3. Accessed: 2025-07-30.

Xu, R.; and Peng, J. 2025. A Comprehensive Survey of Deep Research: Systems, Methodologies, and Applications. arXiv:2506.12594.

Xu, R.; Wang, Z.; Fan, R.-Z.; and Liu, P. 2024. Benchmarking Benchmark Leakage in Large Language Models. arXiv:2404.18824.

Yang, Z.; Qi, P.; Zhang, S.; Bengio, Y.; Cohen, W. W.; Salakhutdinov, R.; and Manning, C. D. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Yu, J.; He, R.; and Ying, R. 2023. Thought propagation: An analogical approach to complex reasoning with large language models. *arXiv preprint arXiv:2310.03965*.

Yu, J.; Xu, T.; Rong, Y.; Huang, J.; and He, R. 2022. Structure-aware conditional variational auto-encoder for constrained molecule optimization. *Pattern Recognition*, 126: 108581.

Zhou, J.; Li, W.; Liao, Y.; Zhang, N.; Qi, T. M. Z.; Wu, Y.; and Yang, T. 2025a. AcademicBrowse: Benchmarking Academic Browse Ability of LLMs. *arXiv preprint arXiv:2506.13784*.

Zhou, X.; Weyssow, M.; Widyasari, R.; Zhang, T.; He, J.; Lyu, Y.; Chang, J.; Zhang, B.; Huang, D.; and Lo, D. 2025b. LessLeak-Bench: A First Investigation of Data Leakage in LLMs Across 83 Software Engineering Benchmarks. arXiv:2502.06215.

A Data Leakage Detection

To verify that our benchmark minimize the risk of data leakage from the pretraining corpora of LLMs, we conduct a comprehensive *leakage simulation experiment* (Xu et al. 2024; Zhou et al. 2025b) across all 8 evaluated models. This procedure estimates whether any model can reproduce the withheld portion of a task when prompted with only the first half of the task description.

A.1 Experimental Procedure

Given a task instance T, we split it into two parts at a punctuation boundary i^* closest to the midpoint:

$$i^* = \arg\min_{i \in \mathcal{P}} \left| i - \frac{|T|}{2} \right|, \quad \mathcal{P} = \{ j \mid T[j] \in \text{punctuation} \} \tag{9}$$

Let $\mathcal{M}_1, \ldots, \mathcal{M}_8$ denote the 8 models evaluated in the main paper. Each model $\mathcal{M}_k$ is queried with the prompt T_{prefix}, yielding a generated continuation:

$$\hat{T}_{\text{suffix}}^{(k)} = \mathcal{M}_k(T_{\text{prefix}}) \tag{10}$$

where $T_{\text{prefix}} = T[: i^*]$, $T_{\text{suffix}} = T[i^* :]$. This formulation allows us to compare the model-generated continuation $\hat{T}_{\text{suffix}}^{(k)}$ with the ground-truth suffix T_{suffix}. If the similarity between these two sequences is unexpectedly high, even though the model only received the input prefix, it may suggest that the model has memorized or encountered the full task during pretraining, thereby posing a risk of data leakage.

A.2 Similarity Metrics

To assess whether $\hat{T}_{\text{suffix}}^{(k)}$ potentially replicates the ground-truth suffix T_{suffix}, we compute three types of similarity:

1. String Similarity. We compute string-level similarity between the model-generated suffix and the ground-truth suffix using the normalized *Longest Common Subsequence* (LCS) metric (Öztürk, Ozkirimli, and Özgür 2016; Wang and Dong 2020). The similarity score for model $\mathcal{M}_k$ is defined as:

$$\text{Sim}_{\text{string}}^{(k)} = \frac{2 \cdot |\text{LCS}(\hat{T}_{\text{suffix}}^{(k)}, T_{\text{suffix}})|}{|\hat{T}_{\text{suffix}}^{(k)}| + |T_{\text{suffix}}|} \tag{11}$$

Here:

- $\mathcal{M}_k$ denotes the k-th evaluated model.
- T_{suffix} is the reference suffix (i.e., the ground-truth continuation of a given task).
- $\hat{T}_{\text{suffix}}^{(k)} = \mathcal{M}_k(T_{\text{prefix}})$ is the suffix generated by model $\mathcal{M}_k$ when prompted with the task prefix T_{prefix}.
- $\text{LCS}(A, B)$ denotes the *Longest Common Subsequence* between sequences A and B, i.e., the longest sequence of characters that appear left-to-right (but not necessarily contiguously) in both A and B.
- $| \cdot |$ denotes the number of characters in a sequence.

This normalized LCS score ranges from 0 to 1, where 1 indicates that the two sequences are identical (character order preserved), and 0 indicates no character-level overlap. The formula symmetrically normalizes the LCS length by the average length of the two sequences, ensuring robustness to differing output lengths.

2. TF-IDF Cosine Similarity. We compute lexical similarity between the generated suffix and the reference suffix using cosine similarity over their TF-IDF representations (Castaño et al. 2016; Wang and Dong 2020). The score for model $\mathcal{M}_k$ is given by:

$$\text{Sim}_{\text{tfidf}}^{(k)} = \frac{\mathbf{v}^{(k)} \cdot \mathbf{v}_T}{\|\mathbf{v}^{(k)}\| \cdot \|\mathbf{v}_T\|} \tag{12}$$

Here:

- $\mathcal{M}_k$ denotes the k-th evaluated model.
- $\hat{T}_{\text{suffix}}^{(k)}$ and T_{suffix} are the model-generated and reference suffixes, respectively.
- $\mathbf{v}^{(k)} \in \mathbb{R}^d$ is the TF-IDF vector of $\hat{T}_{\text{suffix}}^{(k)}$.
- $\mathbf{v}_T \in \mathbb{R}^d$ is the TF-IDF vector of T_{suffix}.
- $\mathbf{v}^{(k)} \cdot \mathbf{v}_T$ denotes the dot product between the two vectors.
- $\|\mathbf{v}\|$ denotes the Euclidean norm (i.e., $\|\mathbf{v}\| = \sqrt{\sum_i v_i^2}$) of vector $\mathbf{v}$.

TF-IDF vectors are computed over a fixed vocabulary, transforming each suffix into a weighted bag-of-words representation. Cosine similarity then measures the angular similarity between these two vectors, ranging from 0 (completely dissimilar) to 1 (identical in direction).

3. Word Overlap Ratio. We compute word-level lexical overlap between the generated and reference suffixes using the normalized word set intersection (Jindal and Leema 2024). The score for model $\mathcal{M}_k$ is defined as:

$$\text{Sim}_{\text{overlap}}^{(k)} = \frac{|W^{(k)} \cap W_T|}{|W_T|} \tag{13}$$

Here:

- $\mathcal{M}_k$ denotes the k-th evaluated model.
- $\hat{T}_{\text{suffix}}^{(k)}$ and T_{suffix} are the model-generated and reference suffixes, respectively.
- $W^{(k)}$ is the set of unique words in $\hat{T}_{\text{suffix}}^{(k)}$, after tokenization and lowercasing.
- W_T is the set of unique words in T_{suffix}, processed identically.
- $|A|$ denotes the cardinality (i.e., number of elements) of set A.

This metric captures the proportion of reference words that are correctly recovered in the model output, regardless of order or repetition. A higher score indicates greater lexical fidelity to the reference.

4. Composite Similarity. To obtain a unified similarity score that balances multiple aspects of textual similarity, we compute a weighted combination of the three individual metrics (De Boom et al. 2015):

$$\text{Sim}^{(k)}_{\text{composite}} = 0.4 \cdot \text{Sim}^{(k)}_{\text{string}} + 0.4 \cdot \text{Sim}^{(k)}_{\text{tfidf}} + 0.2 \cdot \text{Sim}^{(k)}_{\text{overlap}} \tag{14}$$

Here:

- $\text{Sim}^{(k)}_{\text{string}}$ measures normalized character-level alignment via longest common subsequence.
- $\text{Sim}^{(k)}_{\text{tfidf}}$ measures cosine similarity over TF-IDF vector representations.
- $\text{Sim}^{(k)}_{\text{overlap}}$ measures word-level lexical overlap based on unique token sets.

The weights $(0.4, 0.4, 0.2)$ were heuristically chosen to prioritize structural and semantic similarity (string and TF-IDF), while still accounting for lexical coverage (overlap). The resulting composite score lies in the range $[0, 1]$, with higher values indicating greater alignment with the reference text.

A.3 Leakage Criterion

We consider a task as *potentially leaked* by model $\mathcal{M}_k$ if the composite score exceeds a threshold:

$$\texttt{is\_leaked}^{(k)} = \mathbb{I}\left[\text{Sim}^{(k)}_{\text{composite}} > \tau\right], \quad \text{with } \tau = 0.7 \tag{15}$$

We set the threshold $\tau = 0.7$ as a conservative criterion to identify potential cases of memorization or contamination. This choice is supported by prior work on training data extraction (Carlini et al. 2021) and sensitive domain leakage (Lehman et al. 2021), which report that similarity scores at or above 0.7 often correspond to memorized or verbatim training content. Such a threshold ensures high precision in detecting potential data leakage while minimizing false positives.

A.4 Experimental Setup

We evaluate all 8 models on the same set of 100 benchmark tasks. Each model is accessed via external API. We use a decoding temperature of 0.1, a maximum output length of 500 tokens.

For each model-task pair, we log:

- Input prefix T_{prefix}
- Model continuation $\hat{T}^{(k)}_{\text{suffix}}$
- Reference suffix T_{suffix}
- Three similarity scores and a composite score
- Binary leakage flag

The following example illustrates a typical case where the model-generated continuation remains semantically and lexically distinct from the ground truth, thereby avoiding the leakage threshold:

Data Leakage Detection Example (Non-Leaked Sample)

Input Prefix:
Test the effectiveness of transformer architecture visualization tools in helping software architects understand generative AI systems compared to traditional documentation

Reference Suffix:
methods. Conduct controlled experiments measuring comprehension and decision-making quality, delivering statistical analysis results and recommendations.

Model-Generated Suffix:
To assess the effectiveness of transformer architecture visualization tools in aiding software architects' understanding of generative AI systems compared to traditional documentation, a comprehensive...

Similarity Scores:

- String similarity: 15.3%
- TF-IDF similarity: 9.4%
- Word overlap: 14.3%
- Composite similarity: 12.7%

Leakage Flag: `False`

A.5 Results and Analysis

Table 4 presents the detection summary for all 8 models.

Across all evaluated models, none of the 100 sampled tasks triggered the leakage criterion, indicating that no model exceeded the composite similarity threshold of 0.7. The average similarity scores remain consistently low across string-level, semantic, and lexical dimensions. This suggests that the generated continuations are largely dissimilar from the ground-truth suffixes and unlikely to be the result of memorization. These results provide strong evidence that our benchmark is free from pretraining contamination or data leakage.

B Alignment Between Automated Evaluation and Human Judgment

B.1 Motivation

To ensure the reliability of our benchmark evaluations, it is essential to verify that our automated scoring metrics (KAE and ACE) align well with human judgments. This section provides a systematic analysis of their agreement with expert annotations.

B.2 Experimental Setup

We randomly sample a representative subset of benchmark tasks and collect human evaluations for model-generated responses. Human annotators are instructed to assess each response according to the same criteria used in our automated evaluation. Each response is rated independently by three annotators, and their scores are averaged.

Model	**Leak**	**Avg. Comp.**	**Avg. StrSim**	**Avg. TFIDF**	**Avg. Overlap**	**Count**
gpt-4o-search-preview	0.0%	9.8%	10.4%	7.1%	14.1%	0
gpt-4o-mini-search-preview	0.0%	9.4%	7.7%	7.2%	17.2%	0
gpt-4.1	0.0%	14.8%	22.4%	8.3%	12.4%	0
gpt-4.1-mini	0.0%	13.6%	17.9%	8.9%	14.4%	0
o4-mini	0.0%	10.5%	5.7%	9.4%	21.9%	0
gemini-2.5-pro	0.0%	13.8%	20.8%	7.5%	12.4%	0
gemini-2.5-flash	0.0%	13.7%	22.2%	6.9%	10.1%	0
grok-4	0.0%	14.8%	22.2%	8.4%	12.7%	0
Average	0.0%	12.6%	16.1%	8.0%	14.4%	0.0

Table 4: Average similarity scores across 100 benchmark tasks for each evaluated model. Each row reports how similar the model-generated suffix is to the ground-truth suffix, given the same task prefix. **Avg. Comp.** denotes the composite similarity score, computed as a weighted average of **Avg. StrSim** (string similarity), **Avg. TFIDF** (TF-IDF cosine similarity), and **Avg. Overlap** (token overlap rate). **Leak** shows the proportion of tasks whose composite similarity exceeds 0.7, indicating potential data leakage. **Count** reflects the number of suspected leakage cases (all zero). Averages are computed over 100 tasks per model.

Metric Definitions We compute the following correlation coefficients between the automated scores and the averaged human scores:

- **Spearman's Rank Correlation** (ρ): Measures the monotonic relationship between two ranked variables (Spearman 1904). It is computed as:

$$\rho = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)} \tag{16}$$

  where d_i is the difference between the ranks of the i-th observation and n is the total number of samples.

- **Pearson Correlation** (r): Measures the linear correlation between two variables X and Y (Pearson 1895):

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \tag{17}$$

  where $\bar{X}$ and $\bar{Y}$ denote the means of X and Y, respectively.

- **Kendall's Tau** (τ): Measures ordinal association by comparing concordant and discordant pairs (Kendall 1938):

$$\tau = \frac{C - D}{\frac{1}{2}n(n-1)} \tag{18}$$

  where C and D are the number of concordant and discordant pairs among all $n(n-1)/2$ possible pairs.

B.3 Results and Analysis

Metric	**KAE vs. Human**	**ACE vs. Human**
Spearman's ρ	0.84	0.81
Pearson's r	0.79	0.76
Kendall's τ	0.68	0.65

Table 5: Agreement between automated evaluation methods and human judgments on a representative set of benchmark tasks.

As shown in Table 5, both KAE and ACE exhibit strong correlations with human scores, particularly in terms of ranking consistency. This demonstrates that our evaluation protocol provides a faithful and efficient approximation of human preference and judgment.

B.4 Qualitative Case Study

To complement the quantitative analysis, we present a representative example illustrating how ACE aligns with human intuition in a realistic task setting. This case is shown in Table 6.

B.5 Summary

The observed correlations and case study confirm that KAE and ACE offer scalable, interpretable, and human-aligned metrics for evaluating model performance on research tasks. They allow for reliable comparisons without incurring the cost and variability of manual annotation.

C Sample Checklist Generated by ACE

We present a full example of a task-specific checklist automatically generated by a strong LLM (*gemini-2.5-flash*) as part of the ACE framework. The checklist is conditioned on a challenging task in the *Design* phase, requiring methodological synthesis and theoretical precision. Each criterion includes a semantic title, a detailed description, and a relative weight, summing to 1.0 across all dimensions.

Task: *Design an experimental protocol to compare the 2-approximation guarantee of greedy maximal independent set algorithms against other linear-time edge-colored clustering methods on hypergraphs. Include benchmark datasets, performance metrics for $O(\sum$ hyperedge sizes$)$ time complexity validation, and statistical significance testing procedures in a detailed methodology document.*

Model Output:
# Methodology
Datasets: We will evaluate on a suite of real-world hypergraphs with categorical (edge-colored) interactions that have been used as benchmarks in prior ECC research...
Human Score: 4.5/10 **ACE Score**: 4.35
Comment: Both human and ACE assign similar moderate scores, suggesting the response partially meets the task objective with a coherent but limited methodology description.

Table 6: Example of strong agreement between ACE and human judgment.

Generated Checklist

Task Query (Design Phase):
Design an experimental protocol to compare the 2-approximation guarantee of greedy maximal independent set algorithms against other linear-time edge-colored clustering methods on hypergraphs. Include benchmark datasets, performance metrics for $O(\sum$ hyperedge sizes$)$ time complexity validation, and statistical significance testing procedures in a detailed methodology document.

Checklist Criteria:

1. **Methodological Rigor and Experimental Design Quality** **(Weight: 0.45)**
   Evaluates the scientific soundness of the experimental protocol including: proper control variables, valid comparison methodology between greedy maximal independent set algorithms and edge-colored clustering methods, appropriate experimental conditions, clear hypothesis formulation, and rigorous approach to validating $O(\sum$ hyperedge sizes$)$ time complexity. Must demonstrate understanding of algorithmic analysis principles and fair comparison frameworks.

2. **Technical Accuracy and Theoretical Grounding** **(Weight: 0.30)**
   Assesses correctness of technical concepts including: accurate understanding of 2-approximation guarantees, proper characterization of greedy maximal independent set algorithms on hypergraphs, correct complexity analysis methodology, valid performance metrics for the specified time complexity, and appropriate statistical significance testing procedures. Must demonstrate deep understanding of graph theory, approximation algorithms, and computational complexity.

3. **Completeness and Implementation Feasibility** **(Weight: 0.15)**
   Evaluates whether the response addresses all required components: benchmark dataset specifications with hypergraph characteristics, comprehensive performance metrics beyond time complexity, detailed statistical testing procedures, practical implementation considerations, and completeness of the methodology document structure. Must provide actionable protocols that can be realistically executed.

4. **Clarity and Professional Documentation Standards** **(Weight: 0.10)**
   Assesses the quality of presentation including: clear structure suitable for a methodology document, precise technical language, logical flow of experimental steps, appropriate level of detail for reproducibility, and professional formatting. Must be comprehensible to researchers in the field while maintaining technical precision.

Checklist Metadata:

- **Generated by:** *gemini-2.5-flash*
- **Task Type:** Method Blueprint
- **Task Category:** Science & Technology
- **Task Difficulty:** Advanced
- **Video Source:** *DSI Seminar Series — Algorithms and Applications of Edge-Colored Hypergraph Clustering*

D Prompt Templates

In this section, we include all the prompt templates employed during the data construction and model evaluation stages. These prompts are carefully crafted to align with the task objectives and ensure standardized interactions across models and tasks, thereby supporting transparency and reproducibility of our benchmark.

Specifically:

- **INSPIRATION_EXTRACTION_PROMPT**
  Extracts categorized research inspirations (Limitation, Methodology, Transdisciplinarity, Hypothesis) from seminar transcripts, capturing authentic research motivations to seed task generation.
- **TASK_GENERATOR_PROMPT**
  Transforms extracted inspirations into structured DeepResearch tasks that span the full research workflow (*Synthesize*, *Design*, *Evaluate*), grounding the benchmark in real research challenges.
- **RESEARCH_TASK_SCORING_PROMPT**
  Enables head-to-head comparison of task quality, where a judge assesses competing task formulations based on clarity, specificity, feasibility, and academic value.
- **KEY_POINT_EXTRACTION_PROMPT**
  Extracts key points from the content retrieved via URL associated with a research query. These points serve as targeted evidence crucial for evaluating response faithfulness.
- **KEY_POINT_RELEVANCE_PROMPT**
  Evaluates whether a model-generated response appropriately reflects a specific key point, helping assess alignment with source-grounded facts or requirements.
- **CHECKLIST_TEMPLATE_PROMPT**
  Supports the construction of comprehensive, task-specific evaluation rubrics used to guide human or model-based scoring of open-ended responses.
- **SINGLE_CRITERION_SCORING_PROMPT**
  Enables fine-grained assessment of LLM responses along a single evaluation criterion from the checklist, promoting transparency and score traceability.

INSPIRATION_EXTRACTION_PROMPT

System Role:
You are Inspiration-Extractor, an expert research assistant.

Goal:
Read the transcript below and output a list of *inspirations* — concise research leads with academic value. Each inspiration must satisfy at least **two** of the following four qualities:

- **Novelty** — introduces or implies a new idea, method, or perspective.
- **Explorability** — offers a clear starting point for further modeling, experiments, or policy analysis.
- **Challenge** — exposes a limitation, bottleneck, or unresolved issue.
- **Verifiability** — can ultimately be confirmed or refuted via data, experimentation, or simulation.

Categorization Schema: Each inspiration must be assigned exactly one of the following types:

- **Limitation** — Typical Focus: unresolved issue or missing evidence; Required Traits: **Challenge + Explorability**
- **Methodology** — Typical Focus: new technique or framework; Required Traits: **Novelty + Explorability**
- **Transdisciplinary** — Typical Focus: cross-domain application; Required Traits: **Novelty + Explorability**
- **Hypothesis** — Typical Focus: causal or quantitative statement; Required Traits: **Verifiability + Explorability**

Output Format: Each line must be a compact JSON object:

```
{
  "text": "< 4-5 sentences, <= 300 words, faithful to transcript >",
  "type": "Limitation | Methodology | Transdisciplinary | Hypothesis"
}
```

Extraction Algorithm:

1. **Scan:** Detect cue phrases:
   Limitation → “unsolved”, “bottleneck”, “lack of. . . ”
   Methodology → “we propose. . . ”, “new framework. . . ”
   Transdisciplinary → “apply A to B”, “bridge. . . ”
   Hypothesis → causal verbs (e.g., “leads to”), quantitative predictions
2. **Cluster:** Combine adjacent lines on the same idea ($\leq$100 words).
3. **Qualify:** Ensure each candidate satisfies $\geq$2 of the four qualities.
4. **Limit:** Output maximum 10 inspirations.
5. **Faithfulness:** No hallucination; paraphrase lightly.
6. **Reasoning:** You may reason internally, but **output only JSONL**.

Transcript Format:
`<|begin_of_transcript|>` {transcript} `<|end_of_transcript|>`

TASK_GENERATOR_PROMPT

System Role:
You are DeepResearch-Task-Generator.

Goal:
Transform a set of research *inspirations* into concrete DeepResearch tasks that span the full research workflow.

1. Input:
You will receive a JSON array named `<<<INSPIRATIONS>>>`, where each element has the schema:

```
{
  "text": "< 4-5 sentences, <= 300 words, faithful to transcript>",
  "type": "Limitation | Methodology | Transdisciplinary | Hypothesis"
}
```

2. Output:
Return **5–8** objects in a JSON array. **Nothing else.** Each object must include exactly these fields:

Each object must include exactly the following fields:

- **phase** (string): One of Synthesize, Design, or Evaluate.
- **task type** (string): Choose from the task families listed in Section 3.
- **difficulty** (string): Basic or Advanced.
- **task** (string): A self-contained description of at most 100 words, including a concrete deliverable.

3. Exhaustive Task-Family Menu:
*(You may **NOT** invent new families.)*

Phase: Synthesize

- **Literature Survey** — e.g., map arguments in scholarly debates about Universal Basic Income (2020–2024)
- **Trend / Market Scan** — e.g., analyze company reports to identify top 3 priorities in the auto industry
- **Requirements Gathering / Needs Analysis** — e.g., survey researchers to uncover unmet needs in DNA software

Phase: Design

- **Hypothesis Generation** — e.g., propose a testable hypothesis on remote work and retention
- **Method / Experiment Blueprint** — e.g., design a double-blind protocol for supplement efficacy
- **Prototype / System Specification** — e.g., write a functional spec for a library checkout system
- **Evaluation Metric Design** — e.g., define a “Fairness-Accuracy Score” for AI algorithm evaluation

Phase: Evaluate

- **Empirical / Simulation Test** — e.g., simulate tax cut impact using economic models
- **Replicability & Bias Review** — e.g., audit published experiments for sampling bias
- **Comparative Analysis** — e.g., compare feature sets of major cloud storage providers

4. Construction Rules:

1. Cover at least one task from each **phase**; no family repeated more than twice.
2. Ground every task in one or more inspirations. Explicitly **weave** key wording from the inspiration(s) into the task.
3. Let the `type` steer emphasis: **Limitation** → find gaps; **Methodology** → design; **Transdisciplinary** → bridge domains; **Hypothesis** → test assertions.
4. **Difficulty**: `Basic` = feasible with public data in $\leq$3h; `Advanced` = needs novel data, tools, or reasoning.
5. Each task must be self-contained and include a deliverable (e.g., “deliver a taxonomy table”).
6. Do **not** reference the full transcript or original inspirations; the task must stand alone.

5. Final Output:
Respond **only** with the JSON array. **No extra commentary.**

RESEARCH_TASK_SCORING_PROMPT

System Role:
You are DeepResearch-Task-Judge, a strict reviewer who must decide which of two research tasks is higher quality.

Rubric (equal weight for each dimension):

- **Clarity** – Wording unambiguous; reader needs no transcript lookup.
- **Actionability** – Deliverable concrete; scope doable via LLM reasoning or code-writing.
- **Novelty** – Offers non-obvious angle; avoids duplication of similar tasks.
- **Depth-Fit** – Difficulty tag (Basic — Advanced) matches workload and construction rules.
- **Consistency** – Fully follows template ($\leq$100 words, no meta phrases like “the seminar noted...”, etc.).

Scoring Procedure:

1. Compare task_A and task_B holistically under the rubric.
2. Assign each dimension an integer score from 1 to 5.
3. Compute: overall = round((clarity + actionability + novelty + depth_fit + consistency) / 5, 2).
4. Select the task with the higher overall score as the winner.
5. If the scores tie, choose the task that is slightly better and set confidence to 0.55.
6. Return only valid JSON. No other explanation or preamble.

Output Format (One JSON Object):

```
{
  "winner_id": "A or B",
  "loser_id": "A or B",
  "scores": {
    "winner_overall": x.xx,
    "loser_overall": y.yy
  },
  "winner_reason": "<= 40-word justification>",
  "confidence": 0-1 float
}
```

Assume:
The assistant receives **one** user message containing:

```
{
  "task_A": { ... full task object ... },
  "task_B": { ... full task object ... }
}
```

Begin Judgement.

KEY_POINT_EXTRACTION_PROMPT

System Role:
You are an expert assistant performing key point extraction for question answering.

Goal:
Given a query and a supporting text passage, identify **key points** that are crucial to answering the query. These are not generic important sentences, but the specific evidence that directly helps address the query.

Instructions:

- Each key point must **help respond to the query**.
- Each point should be associated with one or more **verbatim spans** copied directly from the text.
- **Do not modify or rephrase any span.**
- Keep key point descriptions concise and abstract if needed, but all `spans` must be exact copies from the source text.
- No extra commentary, no markdown, no free-text outside of the JSON object.

Output Format:

```
{
  "points": [
    {
      "point_number": point_number,
      "point_content": point_content,
      "spans": [span1, span2, ...]
    },
    ...
  ]
}
```

Reminders:

- Key point content can be abstracted or summarized.
- Every span must be copied exactly as-is from the passage.
- Multiple spans can be associated with a single key point.
- Respond strictly with a valid JSON object — **no explanations, no markdown, no extra text**.

Inputs:

- **[Query]**: {question}
- **[Text]**: {text}

KEY_POINT_RELEVANCE_PROMPT

System Role:
You are a professional text relationship analyst. Your job is to evaluate whether a model-generated response appropriately reflects a specific key point in relation to the original research task.

Original Task:
{original_task}

Response Content:
{response_content}

Key Point to Analyze:
{key_point}

Analysis Instructions:

- Carefully read the key point, the original task, and the response content.
- Determine whether the response:
  - **SUPPORTS** the key point — it affirms, explains, or reinforces the point.
  - **OMITS** the key point — it does not mention or address the point at all.
  - **CONTRADICTS** the key point — it says something that disagrees with or negates the point.

Output Format (Valid JSON Only):

```
{
  "relationship": "SUPPORTS | OMITS | CONTRADICTS",
  "confidence": 0.0--1.0,
  "reasoning": "Detailed explanation of your judgment.",
  "key_aspects": ["list", "key", "determining", "factors"]
}
```

Important Notes:

- **relationship** must be exactly one of: SUPPORTS, OMITS, CONTRADICTS.
- **confidence** is a float between 0.0 and 1.0 indicating confidence in the judgment.
- **reasoning** should clearly justify the decision.
- **key_aspects** should list the main textual or semantic factors that influenced the judgment.

Final Instruction:
Please analyze the response according to the above instructions and return **only** the JSON object, with no extra commentary or formatting.

CHECKLIST_TEMPLATE_PROMPT (Page 1/2)

System Role:
You are a helpful assistant who creates comprehensive evaluation rubrics for LLM responses to help humans evaluate LLMs efficiently and accurately.

Goal:
Given a user query, generate a task-specific evaluation checklist to guide accurate and efficient human assessment of LLM responses.

Instruction:

- You will be given a user query.
- Your task is to analyze the query and produce a comprehensive evaluation rubric covering all key aspects for scoring LLM responses.
- Each rubric item must be actionable, weighted, and specific to the query's type and requirements.

Query Format:

```
<|begin_of_query|>
{user_query}
<|end_of_query|>
```

Checklist Construction Requirements:

- Be specific to the query (e.g., technical, creative, instructional).
- Cover multiple aspects: content accuracy, completeness, clarity, formatting, instruction following, etc.
- Include weights (0.0–1.0) that reflect each criterion's relative importance.
- Use 3–6 items per rubric depending on query complexity.
- **Do not** use identical weights across tasks. Vary by phase and task type.

Phase-Specific Priorities:
Synthesize Phase

- Literature Survey: Emphasize comprehensiveness and source quality
- Trend / Market Scan: Emphasize data accuracy and trend insight
- Requirements Analysis: Emphasize stakeholder coverage and need validation

Design Phase

- Hypothesis Generation: Emphasize testability and theoretical grounding
- Method / Experiment Blueprint: Emphasize methodological rigor and feasibility
- Prototype / System Specification: Emphasize technical accuracy and completeness
- Evaluation Metric Design: Emphasize metric validity and applicability

Evaluate Phase

- Empirical / Simulation Test: Emphasize statistical rigor and result interpretation
- Replicability Review: Emphasize methodology clarity and bias detection
- Comparative Analysis: Emphasize fairness and analytical depth

CHECKLIST_TEMPLATE_PROMPT (Page 2/2)

Output Format (Valid JSON Only):

```
{
  "evaluation_criteria": [
    {
      "title": "Most Critical Aspect for This Query Type",
      "weight": 0.4,
      "description": "Detailed description of what to evaluate and criteria"
    },
    {
      "title": "Secondary Important Aspect",
      "weight": 0.3,
      "description": "Detailed description of what to evaluate and criteria"
    },
    {
      "title": "Supporting Aspect",
      "weight": 0.2,
      "description": "Detailed description of what to evaluate and criteria"
    },
    {
      "title": "Additional Quality Check",
      "weight": 0.1,
      "description": "Detailed description of what to evaluate and criteria"
    }
  ]
}
```

Final Guidelines:

- Highest-weighted criterion should match the task’s critical requirement.
- Do not use generic titles or descriptions; each item must match the query type.
- All weights must sum to approximately 1.0.
- Output must be valid JSON that is directly parseable.

SINGLE_CRITERION_SCORING_PROMPT (Page 1/2)

System Role:
You are a highly respected academic evaluator known for upholding the most rigorous standards in your field. Institutions seek your expertise when they require a meticulous and uncompromisingly thorough assessment grounded in scholarly precision.

Evaluation Criterion:
Single Criterion Evaluation: {checklist_item.title}
{checklist_item.description}

Task Context:

- Category: {category}
- Task Type: {task_type}
- Difficulty: {difficulty}

Critical Instruction:
You are evaluating this response **solely** based on this specific criterion. While the focus is narrow, your expectations for this dimension should remain rigorous and well-calibrated to the task type and category.

Research Task:
{task_query}

Submitted Response:
{response_content}

Evaluation Approach:
Assess how well the response performs on the criterion ”{checklist_item.title}” using the same exacting standards applied to work submitted to top-tier venues. Evaluating a single aspect does not lower the bar — it raises the bar for that one dimension.

Uncompromising Quality Benchmarks:
Exceptional Mastery (8–10):
Handled with extraordinary rigor and insight:

- Comprehensive, flawless treatment of every nuance in the criterion
- Demonstrates domain-advancing insight and precision
- Impressive rigor, originality, and completeness

Basic Competence (5–7):
Functional but significantly limited in rigor or completeness:

- Covers the basics but lacks depth
- Demonstrates gaps or missed opportunities
- Requires improvement to meet high standards

Inadequate (1–4):
Deep deficiencies that compromise this criterion:

- Incomplete, flawed, or misguided
- Demonstrates poor understanding of what the criterion requires
- Fails to meet professional standards

Complete Failure (0):
No meaningful engagement with this specific criterion.

SINGLE_CRITERION_SCORING_PROMPT (Page 2/2)

Rigorous Single-Criterion Analysis:

- **Precision of Coverage:** Does the response address every essential element of this criterion?
- **Quality of Treatment:** Is the handling sophisticated enough to satisfy domain experts?
- **Depth vs. Superficiality:** Does it reflect genuine mastery or just surface-level familiarity?
- **Criterion-Specific Rigor:** Are claims and evidence within this criterion held to top-tier standards?
- **Professional Adequacy:** Would a specialist approve this for publication?
- **Gap Detection:** What deficiencies or oversights exist for this criterion?

Strict Evaluation Principles:

- No mercy for single dimensions — maximal scrutiny applies
- High bar = domain expert satisfaction
- Zero tolerance for mediocrity
- Actively seek flaws, gaps, and weaknesses
- Assume inadequacy by default

Response Format (Valid JSON Only):

```
{
  "rating": integer (0-10),
  "justification": "Explain how this response meets the criterion."
}
```

Final Reminder:
Your evaluation should maintain high standards, even when focusing on a single dimension. High scores should be reserved for responses that demonstrate truly exceptional performance on this specific criterion.