DeepResearch Retail: Benchmarking Tool-Augmented Deep Research in the E-Commerce Domain

Rafael Ferreira[1,2*], Flavio Di Palo[3], Huilin Lu[3], Ayush Jain[3], Harsha Aduri[3]
[1]NOVA University of Lisbon, [2]NOVA LINCS, [3]Amazon
rah.ferreira@campus.fct.unl.pt
{paloflav, huilinlu, jainays, venaduri}@amazon.com

Abstract

Deep Research (DR) systems autonomously retrieve and synthesize information from web sources, however, industrial DR applications face a critical gap: effective integration of internal tools with web search. In this work, we introduce DeepResearch Retail, an evaluation framework grounded in real-world e-commerce data for assessing Deep Research with tools (DR+Tools) in realistic commercial settings. The framework evaluates both factual faithfulness and multidimensional response quality when reasoning over heterogeneous web and internal data sources. We further present Hybrid-ReAct, a multi-agent architecture that demonstrates how collaborative reasoning and tool use can produce evidence-grounded answers. Experimental results validate our framework's utility, showing improvements in agent's performance when leveraging web-page information and multi-agent specialization.

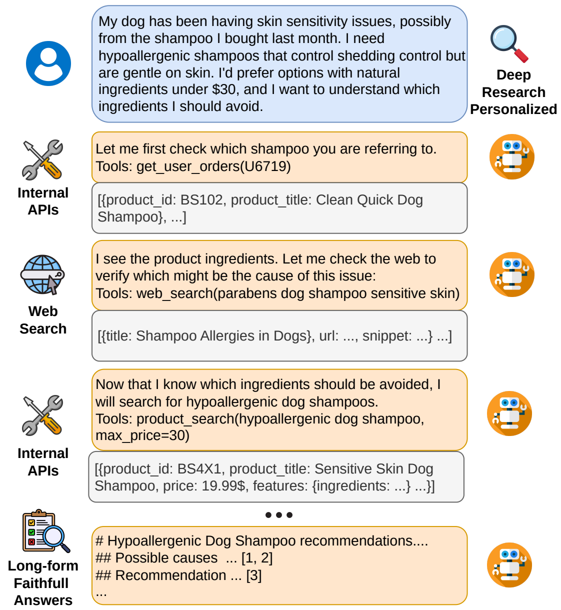

Figure 1: Example conversation from DeepResearch Retail demonstrating a DR+Tools paradigm.

1 Introduction

Deep Research (DR) systems such as OpenAI's DR[1] and Gemini DR[2] have demonstrated strong capabilities in multi-step retrieval and long-form synthesis by combining LLM reasoning with web content. However, existing DR systems remain largely web-centric and do not incorporate structured, domain-specific, and personalized information accessible through internal API tools. In another work-stream, tool-augmented reasoning (Qin et al., 2024; Patil et al., 2024b; Yao et al., 2024) has primarily focused on correct tool invocation rather than evidence-grounded research-style synthesis. Given this, most DR systems have improved tool-use patterns (Alzubi et al., 2025; Li et al., 2025; Zheng et al., 2025), yet lack environments that combine reasoning and long-form answers with diverse tools, which are key to real-world industrial tasks.

Bridging DR and tool-based reasoning is essential for real-world decision-making tasks that require joint reasoning over unstructured web data and internal sources. Existing DR benchmarks (Du et al., 2025; Rosset et al., 2025) emphasize web exploration, whereas tool-use benchmarks (Qin et al., 2024; Yao et al., 2024) omit research-style synthesis and factual evaluation, leaving a gap in unified evaluation settings.

To address this gap, we introduce **DeepResearch Retail**, an evaluation framework designed for the **DR+Tools** paradigm that simulates a realistic e-commerce environment in which agents integrate evidence from the web and internal APIs. Tasks are grounded in authentic product data and user behavior patterns (Hou et al., 2024) capturing complex and realistic information-seeking queries as

*Work performed at Amazon
[1]https://openai.com/index/introducing-deep-research/
[2]https://gemini.google/overview/deep-research/

shown in Figure 1. We also propose a **comprehensive evaluation framework** that measures factual grounding and response quality (Coelho et al., 2025; Du et al., 2025) across heterogeneous data sources. Building on this foundation, we introduce **Hybrid-ReAct**, a multi-agent architecture that decomposes queries, conducts targeted retrieval and tool interactions, synthesizes findings, and produces structured, evidence-grounded reports. Through this work, we aim to bridge the gap between DR and tool use, moving toward domain-aware and faithful DR assistants.

2 Related Work

Retrieval/Deep Research Benchmarks Early benchmarks focused on multi-hop factual question answering (Yang et al., 2018; Ho et al., 2020), which are easy to evaluate but do not reflect complex research workflows. More recent benchmarks, such as DeepResearch Bench (Du et al., 2025) and DeepResearchGym (Coelho et al., 2025), introduce long-form, complex information-seeking questions that require both factual accuracy (Thakur et al., 2025; Wei et al., 2024) and high-quality synthesis. Closer to our domain, WebShop (Yao et al., 2022) and DeepShop (Lyu et al., 2025) simulate e-commerce environments focused on satisfying user constraints, but they do not address DR-style queries or long-form answers. Overall, existing benchmarks rely on limited tool settings and fail to capture domain/user-specific DR capabilities, which we target in this work.

Tool Use Tool-augmented reasoning enables LLMs to interact with APIs. Benchmarks such as APIBench (Patil et al., 2024b) evaluate correct API call generation, while ToolBench (Qin et al., 2024) expands to broader real-world tool manipulation. BFCL (Patil et al., 2024a) further studies function calling across single-turn and multi-turn settings. More recently, τ-bench (Yao et al., 2024) and τ^2-bench (Barres et al., 2025) evaluate dynamic, multi-turn interactions involving agents, users, and domain-specific policies. However, most tool-use benchmarks focus on tool-call correctness and overlook web contexts and long-form factual grounding. We address this limitation through the *DR+Tools* paradigm, which jointly evaluates DR queries, long-form synthesis, and tool use.

Deep Research Systems Earlier systems used RAG (Gao et al., 2023) to connect LLMs with search engines, however, they lacked the multi-step reasoning needed for complex research. Subsequent approaches introduced methods such as CoT (Wei et al., 2022) and ReAct (Yao et al., 2023) to support multi-hop reasoning and tool use (Trivedi et al., 2023; Wang et al., 2024). More recent systems emphasize information synthesis, including Search-o1's reason-in-documents (Li et al., 2025), OpenDeepSearch's (Alzubi et al., 2025) multiple tool use, and Wu et al. (2025) multi-agent architectures. Reinforcement learning approaches, such as Search-R1 (Jin et al., 2025) and ReSearch (Chen et al., 2025), further improve performance and factuality, while DeepResearcher (Zheng et al., 2025) extends DR to multimodal browser environments. Given RL-based methods' substantial computational requirements, we explore a multi-agent zero-shot setting within the DR+Tools paradigm.

3 DeepResearch Retail: A *DR+Tools* Paradigm

In this section, we introduce DeepResearch Retail, a benchmark for evaluating systems in realistic DR+Tools settings involving personalized queries that require complex reasoning, internal API access, and web search integration.

3.1 Data Foundation: Real World E-Commerce Data

To ensure realism, we ground DeepResearch Retail on the dataset proposed by Hou et al. (2024), an e-commerce dataset which includes over 48 million products across 28 categories and more than 571 million user-product interactions. This dataset's scale and diversity captures real-world product metadata and user interactions, providing a robust foundation that reflects real e-commerce dynamics. Figure 2 illustrates the pipeline from raw data to the creation of DeepResearch Retail.

Database Design and Organization To support LLM interaction, we constructed a database organized into collections with synthetic augmentations (example entries in A.1):

- **Users:** The original anonymized user ids are enriched with synthetically generated attributes such as names and addresses using *Faker*[3], resulting in realistic and diverse user representations.

[3] https://github.com/joke2k/faker

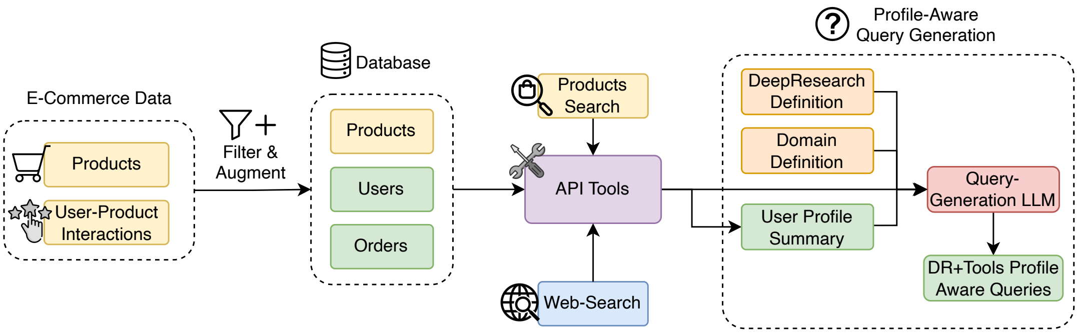

Figure 2: Data generation pipeline for DeepResearch Retail, which integrates real-world data with synthetic augmentation, tools, and a profile-aware query generation.

- **Products:** Contains real product metadata, including titles, descriptions, and prices, supporting search and comparison.
- **Orders:** Created by converting user-product interactions into orders inheriting the original interaction timestamp.

Data Selection and Statistics To ensure quality, we applied a systematic filtering process based on three criteria: 1) *Reduced reliance on memorized knowledge*: prioritizing less common product categories (e.g., personal care, home improvement); 2) *Temporal stability*: excluding rapidly evolving categories (e.g., electronics); and 3) *Balanced coverage*: maintaining diversity across categories.

Given this, from the original product categories, we selected 10 representative categories spanning personal care, household goods, health, pets, and home improvement. We further refined by capping each category at 10k products and keeping only products with at least five user interactions since 2019, ensuring diversity, relevance, and recency. The algorithm for dataset creation is provided in A.2. The final dataset comprises 4.3M orders, 1.8M users, and 80k products, organized in databases, with products indexed for text-based search.

API Tools Design and Creation To allow agent interaction with the dataset, we developed a suite of 9 API tools (detailed in Table 3) that provide controlled access to user profiles, products, orders, and web search. Following τ-bench (Yao et al., 2024) each API is implemented as a Python function with a structured input/output schema, enabling flexible and controlled agent interactions.

3.2 Profile-Aware DR Query Generation

The final step in the benchmark construction generates DR queries that integrate domain knowledge, external evidence, and user-specific context. We implement this via an LLM-based pipeline (Figure 2) that for each user takes as input: 1) *DR Definition*, specifying task complexity and emphasizing long-form reasoning; 2) *Domain Definition*, indicating the target domain (here, e-commerce); 3) *API tools*, listing the tools accessible to the agent; and 4) *User Profile Summary*, created using the output of the user-specific API tools to capture demographic and shopping behavior.

This pipeline enables the generation of realistic, personalized, and profile-aware DR+Tools queries (see Appendix A.4 for additional examples).

3.3 Evaluation Methodology

To evaluate performance in DeepResearch Retail, we propose a framework covering two main aspects: Faithfulness and Answer Quality.

3.3.1 Faithfulness

Following (Coelho et al., 2025; Thakur et al., 2025; Du et al., 2025), we define faithfulness as the factual grounding of a report and use an LLM-as-Judge pipeline with three stages:

1. **Claim extraction:** An LLM extracts claims in the report and associated URLs.
2. **Content retrieval:** The content of each referenced URL is retrieved.
3. **Judgment:** An LLM-as-Judge evaluates each claim against its content, assigning a score (s_i) of fully supported (1), partially supported (0.5), or unsupported (0), following Thakur et al. (2025).

Let N_{total} denote the total number of claims and N_{cited} the subset of claims supported by at least one source. We define two metrics:

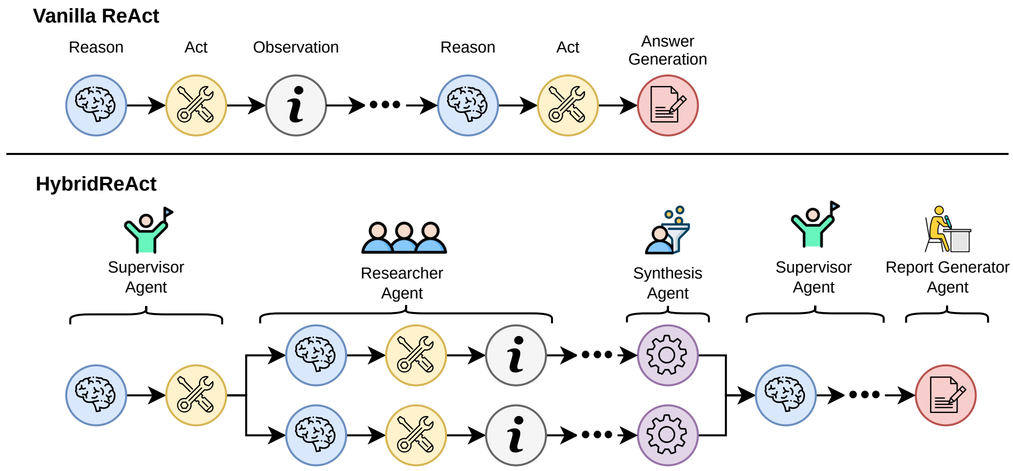

Figure 3: Comparison between vanilla ReAct (Yao et al., 2023) and the proposed Hybrid-ReAct architecture.

Faithfulness Recall (FR) measures the proportion of claims supported by a source:

$$FR = \frac{N_{\text{cited}}}{N_{\text{total}}}. \tag{1}$$

Faithfulness Precision (FP) measures the average factual correctness across sources:

$$FP = \frac{1}{2} \sum_{t \in \{\text{web}, \text{product}\}} \frac{1}{N_{\text{cited}}^{(t)}} \sum_{i=1}^{N_{\text{cited}}^{(t)}} s_i^{(t)}. \tag{2}$$

Evaluating FP separately for web and product sources highlights source-specific limitations and emphasizes the need for factually grounded responses across all evidence types.

3.3.2 Answer Quality

We evaluate response quality across multiple dimensions using tailored LLM prompts. For each dimension, the model assigns a score from 1 to 10 along with a brief rationale. Following (Coelho et al., 2025; Liu et al., 2023), we consider *Clarity*, which reflects readability and logical coherence, and *Insightfulness*, which captures analytical depth and the extent of added value; both dimensions have been shown to align well with human judgments in DR settings (Coelho et al., 2025). In addition, following the definitions in (Coelho et al., 2025), we include *Depth*, measuring the vertical thoroughness of the response, and *Breadth*, assessing the horizontal coverage of relevant subtopics.

We also include domain-specific metrics: *Constraint Following*, evaluating adherence to budget or user preferences; *Personalization*, measuring effective use of user profile information; and *Overall Quality*, providing a holistic assessment. Human validation of our evaluation setup is presented in Appendix D, with all evaluation prompts provided in Appendix H.

4 Hybrid-ReAct: A Multi-Agent Architecture for DR+Tools

Single-agent systems for DR typically rely on a single reasoning stream and/or web search (Alzubi et al., 2025; Li et al., 2025; Zheng et al., 2025). To address these limitations, we introduce Hybrid-ReAct, a multi-agent architecture (Figure 3) that distributes tasks across specialized agents, enabling coordinated parallel reasoning and coherent synthesis of long-form, tool-augmented reports.

Formally, we define an individual agent a as a tuple $a = (\mathcal{T}_a, s_a, \phi_a)$, where $\mathcal{T}_a$ is the set of tools available, s_a is the internal state updated as information is processed, and ϕ_a is the policy that selects tools and produces outputs based on s_a.

Hybrid-ReAct is thus defined as $\mathcal{M} = (\mathcal{A}, \pi)$, where $\mathcal{A} = \{a_{\text{sup}}, a^1_{\text{res}}, \ldots, a^R_{\text{res}}, a_{\text{syn}}, a_{\text{gen}}\}$ is the set of sub-agents, and π is the workflow policy governing agent interactions and message passing. We now describe each of Hybrid-ReAct's components.

Supervisor Agent a_{sup} serves as the central coordinator, operating with a restricted tool set $\mathcal{T}_{\text{sup}} = \{\text{Reason}, \text{LaunchResearcher}, \text{Finalize}\}$. Given a user query q, the Supervisor follows the ReAct pattern (Yao et al., 2023): it first reasons to devise a research strategy and then acts by launching R Researcher Agents, decomposing the query into sub-problems $\{p_i\}_{i=1}^R$. Each sub-problem p_i is assigned to a corresponding Researcher Agent a^i_{res}, enabling parallel evidence collection.

Researcher Agents Each a_{res}^i addresses its sub-problem p_i using a ReAct loop, enabling focused and in-depth exploration. This agent has access to an extended set of tools, $\mathcal{T}_{\text{res}} = \{\text{Reason}, \text{Finalize}\} \cup \mathcal{T}_{\text{API}}$, and iterates until either *Finalize* is invoked or a maximum iteration limit is reached. Upon completion, each agent forwards its accumulated findings f_i to the Synthesis Agent.

Synthesis Agent a_{syn} consolidates the outputs of all Researcher Agents. It does not use tools ($\mathcal{T}_{\text{syn}} = \emptyset$), relying instead on a *Synthesize* prompt to generate intermediate reports $m_i = \text{Synthesize}(f_i)$. Each m_i summarizes key insights, sources, and citations from f_i in a standardized format, ensuring consistency across sub-reports facilitating aggregation by the Supervisor Agent.

Supervisor Control Returned After all Synthesis operations are complete, control returns to the Supervisor Agent. The Supervisor then continues the ReAct loop until it selects Finalize or reaches its maximum iteration limit.

Report Generator Agent a_{gen} produces the final user-facing output. It operates without tools ($\mathcal{T}_{\text{gen}} = \emptyset$) and uses a dedicated *ReportGen* prompt to produce the final report $R = \text{ReportGen}\left(q, \{m_i\}_{i=1}^{R}\right)$. This report emphasizes clarity, logical flow, fluency, and evidence-grounding with proper citations, while internal states, tool invocations, and intermediate outputs are abstracted, resulting in a high-quality report of the DR+Tools process.

5 Experimental Setup

5.1 Dataset

For evaluation, we use the proposed DeepResearch Retail profile-aware query generation pipeline to create 50 queries, which following Du et al. (2025), we manually validated to ensure high-quality. This validation further ensures that all queries reflect DR needs, are answerable with the available tools, and align with the associated user profile, with minor adjustments made as needed. This ensures high-quality queries while preserving scalability through automated query generation.

5.2 Models and Baselines

As baselines, we implement two single-agent approaches. In **Plan-to-Function**, the model generates a plan and all required tool calls in a single turn, which are then executed sequentially. This enables fast execution, however, it prevents the model from adapting based on intermediate tool outputs. In contrast, **ReAct** (Yao et al., 2023) follows an iterative reasoning loop that interleaves reasoning and tool usage step by step. We compare these baselines against our multi-agent **Hybrid-ReAct**, built on top of LangChain's framework (LangChain, 2025). Across all methods, we use Claude 4.0 Sonnet, providing a strong foundation model.

For LLM-as-Judge evaluation, we use Claude 4.5 Sonnet, chosen for its evaluation consistency and accuracy, as less capable judge models tend to produce inflated and low-variance scores (Raina et al., 2024). All evaluations use greedy decoding (i.e., selecting the highest-probability token at each step), to reduce variance across runs and improve reproducibility.

5.3 Web-Search Integration

Since LLMs can benefit from reasoning over external web content (Li et al., 2025), we evaluate two strategies for integrating web search results. In the **Snippets** setting, the model receives the top-k (set to 5) search snippets. In the **Webpage Summarized (WS)** setting, each retrieved page from the top-k is converted to markdown[4] and summarized by the same LLM used for the agent, providing access to richer contextual information. All experiments use Serper as the search engine[5].

6 Results and Discussion

6.1 Faithfulness Metrics

Table 1 summarizes faithfulness results. All methods achieve high Faithfulness Recall (FR > 0.94), indicating most statements are supported by at least one source. For Faithfulness Precision (FP), performance generally improves when using Web-Summarized (WS) content instead of raw snippets. An exception is Hybrid-ReAct+WS, which shows lower product FP due to increased number of web citations, resulting in finer product details being only partially propagated to sub-agents.

Source-level analysis shows web data remains the most challenging due to its unstructured, lengthy nature. Web-summarization mitigates this, as higher FP-Web scores indicate that condensed

[4] https://github.com/unclecode/crawl4ai
[5] https://serper.dev/

	FP	**FP$^{\text{Web}}$**	**FP$^{\text{Prod}}$**	**FR**	**N$_{\text{cited}}$**	**N$^{\text{web}}_{\text{cited}}$**	**N$^{\text{prod}}_{\text{cited}}$**
Plan-to-Function	0.69	0.49	0.84	0.98	15.64	3.76	11.88
Plan-to-Function+WS	**0.74**	0.58	0.87	0.97	17.98	6.26	11.72
React	0.68	0.43	**0.88**	0.97	15.22	4.20	11.02
React+WS	**0.74**	**0.63**	0.84	0.97	17.46	6.20	11.26
Hybrid-ReAct	0.73	0.60	0.85	**0.99**	18.18	5.42	**12.76**
Hybrid-ReAct+WS	0.69	0.61	0.76	0.94	**18.92**	**7.78**	11.14

Table 1: Faithfulness metrics results in DeepResearch Retail.

page summaries improve factual grounding. Regarding citation behavior, the Hybrid-ReAct variants produce a larger number of citations overall, reflecting deeper exploration by specialized agents.

6.2 Answer Quality Metrics

Table 2 shows the answer-quality results. Across most metrics, Hybrid-ReAct achieves the highest scores, with particularly strong gains in dimensions such as *Depth*, *Breadth*, and *Insightfulness*, highlighting the benefits of specialized researcher agents for subtopic exploration. Additional improvements are observed with *WS*, demonstrating the value of effectively leveraging web information.

For *Constraint Following*, all systems perform robustly ($\geq$7.2), indicating adherence to user requirements. Hybrid-ReAct scores slightly lower, as its exploratory behavior and increased web-search usage can occasionally yield suggestions that extend beyond the constraints. For *Personalization*, Hybrid-ReAct scores lower, as profile information is sometimes only partially propagated to subagents, and its more information-rich responses can dilute user-specific details. *WS* has minimal impact on these metrics, as they rely little on external information.

On *Overall Quality*, Hybrid-ReAct achieves the highest score, tied with ReAct+WS. Notably, this metric aims to compresses multiple dimensions into a single scalar, which can obscure nuanced strengths across individual dimensions. This observation motivates our use of complementary evaluation protocols.

To complement this analysis, Appendix C presents a comparative judge evaluation, indicating a preference for Hybrid-ReAct+WS over ReAct+WS, including for *Constraint Following* and *Overall Quality*. In sum, multi-agent specialization and improved handling of web content enhances answer quality, though headroom remains across metrics, as shown by a manual analysis of responses in Appendix G, which highlights current system limitations.

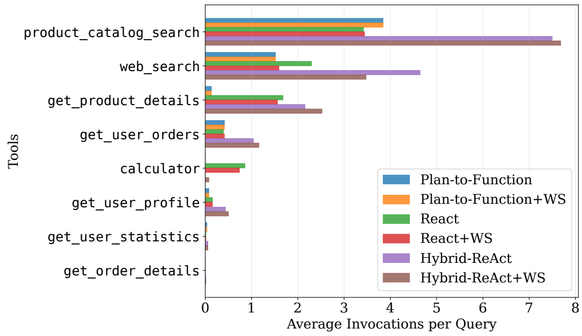

Figure 4: Average number of tool calls per query in DeepResearch Retail. See Appendix A.3 for tool definitions.

6.3 Tool-Use Analysis

Analyzing tool-calling behavior across methods, we see that the average number of tool calls per query is: Plan-to-Function (6.0; unchanged with WS), ReAct (8.8), ReAct+WS (7.9), Hybrid-ReAct (15.9), and Hybrid-ReAct+WS (15.5). As expected, Hybrid-ReAct performs more tool calls, reflecting deeper exploration via specialized sub-agents.

Figure 4 shows a detailed breakdown per tool. *product_catalog_search* is the most frequently used tool across all methods, reflecting high demand for product catalog information, with *get_product_details* following a similar pattern. WS reduces *web_search* calls, as summarized webpages provide more comprehensive context than snippets, lowering the need for follow-up queries.

User-related tools such as *get_user_orders* and *get_user_profile* are accessed inconsistently with Hybrid-ReAct leveraging user data more extensively, likely due to deeper task decomposition. Other tools see minimal use, as methods prioritize broader information over detailed user histories. Future work should explore strategies for richer personalization and profile-aware tool selection.

	Clarity	Depth	Breadth	Insight	Constr. Foll.	Personaliz	Overall	Avg
Plan-to-Function	5.28	6.98	5.62	5.76	7.40	6.62	8.38	6.58
Plan-to-Function+WS	5.10	7.22	6.20	5.98	7.26	6.58	8.42	6.68
React	5.26	7.18	5.74	5.84	7.22	**7.04**	8.60	6.70
React+WS	5.32	7.60	6.30	6.30	**7.60**	6.86	**8.78**	6.97
Hybrid-ReAct	**5.54**	7.64	6.34	6.32	7.34	6.46	**8.78**	6.92
Hybrid-ReAct+WS	5.48	**7.72**	**6.70**	**6.68**	7.46	6.56	8.68	**7.04**

Table 2: Answer quality metrics (1-10 scale) results in DeepResearch Retail.

7 Conclusions

In this work, we introduce $DR{+}Tools$, a paradigm with substantial industrial relevance. In particular, we present DeepResearch Retail, a framework grounded in authentic e-commerce data for evaluating Deep Research systems that use internal tools in commercial settings. The framework provides a suite of domain-specific tools designed to integrate with DR workflows, along with an evaluation methodology that measures both faithfulness to data sources and multidimensional answer quality across real-world tasks. We also present Hybrid-ReAct, a multi-agent system that shows how parallel reasoning, tool use, and grounded report generation can be effectively combined. Experimental results validate the utility of our framework, showing improvements from web-page summarization and multi-agent specialization. Together, these contributions establish a foundation for advancing DR systems toward practical enterprise deployment.

Limitations

Our study provides useful insights into the DR+Tools paradigm; however, it also comes with a few constraints. Although the proposed evaluation framework spans diverse product categories, it remains limited to the e-commerce domain, suggesting the need to explore other application areas. Due to the complex and open-ended nature of the task with multiple acceptable answers, we use an LLM-as-Judge across multiple metrics for evaluation, which while it has been shown to have good alignment with human judgments (Gu et al., 2024; Coelho et al., 2025; Liu et al., 2023), may nonetheless constrain finer grain metrics. In addition, our results show current challenges in web-based faithfulness and answer personalization across methods. Even with these limitations, the proposed methodologies are generalizable to other domains and offers a solid foundation for future extensions.

References

Salaheddin Alzubi, Creston Brooks, Purva Chiniya, Edoardo Contente, Chiara von Gerlach, Lucas Irwin, Yihan Jiang, Arda Kaz, Windsor Nguyen, Sewoong Oh, Himanshu Tyagi, and Pramod Viswanath. 2025. Open deep search: Democratizing search with open-source reasoning agents. *CoRR*, abs/2503.20201.

Victor Barres, Honghua Dong, Soham Ray, Xujie Si, and Karthik Narasimhan. 2025. τ^2-bench: Evaluating conversational agents in a dual-control environment. *CoRR*, abs/2506.07982.

Mingyang Chen, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z. Pan, Wen Zhang, Huajun Chen, Fan Yang, Zenan Zhou, and Weipeng Chen. 2025. Research: Learning to reason with search for llms via reinforcement learning. *CoRR*, abs/2503.19470.

João Coelho, Jingjie Ning, Jingyuan He, Kangrui Mao, Abhijay Paladugu, Pranav Setlur, Jiahe Jin, Jamie Callan, João Magalhães, Bruno Martins, and Chenyan Xiong. 2025. Deepresearchgym: A free, transparent, and reproducible evaluation sandbox for deep research. *Preprint*, arXiv:2505.19253.

Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. 2025. Deepresearch bench: A comprehensive benchmark for deep research agents. *CoRR*, abs/2506.11763.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *CoRR*, abs/2312.10997.

Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Yuanzhuo Wang, and Jian Guo. 2024. A survey on llm-as-a-judge. *CoRR*, abs/2411.15594.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing A multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 6609–6625. International Committee on Computational Linguistics.

Yupeng Hou, Jiacheng Li, Zhankui He, An Yan, Xiusi Chen, and Julian J. McAuley. 2024. Bridging language and items for retrieval and recommendation. *CoRR*, abs/2403.03952.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *CoRR*, abs/2503.09516.

LangChain. 2025. Langchain open deep research. `https://github.com/langchain-ai/open_deep_research`. Accessed: 2025.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. 2025. Search-o1: Agentic search-enhanced large reasoning models. *CoRR*, abs/2501.05366.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 2511–2522. Association for Computational Linguistics.

Yougang Lyu, Xiaoyu Zhang, Lingyong Yan, Maarten de Rijke, Zhaochun Ren, and Xiuying Chen. 2025. Deepshop: A benchmark for deep research shopping agents. *CoRR*, abs/2506.02839.

OpenAI. 2025. gpt-oss-120b & gpt-oss-20b model card. *CoRR*, abs/2508.10925.

Shishir G. Patil, Huanzhi Mao, Charlie Cheng-Jie Ji, Fanjia Yan, Vishnu Suresh, Ion Stoica, and Joseph E. Gonzalez. 2024a. The berkeley function calling leaderboard (bfcl): From tool use to agentic evaluation of large language models. In *Advances in Neural Information Processing Systems*.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. 2024b. Gorilla: Large language model connected with massive apis. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2024. Toolllm: Facilitating large language models to master 16000+ real-world apis. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Vyas Raina, Adian Liusie, and Mark J. F. Gales. 2024. Is llm-as-a-judge robust? investigating universal adversarial attacks on zero-shot LLM assessment. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 7499–7517. Association for Computational Linguistics.

Corbin Rosset, Ho-Lam Chung, Guanghui Qin, Ethan C. Chau, Zhuo Feng, Ahmed Awadallah, Jennifer Neville, and Nikhil Rao. 2025. Researchy questions: A dataset of multi-perspective, decompositional questions for deep research. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2025, Padua, Italy, July 13-18, 2025*, pages 3712–3722. ACM.

Nandan Thakur, Ronak Pradeep, Shivani Upadhyay, Daniel Campos, Nick Craswell, and Jimmy Lin. 2025. Support evaluation for the TREC 2024 RAG track: Comparing human versus LLM judges. *CoRR*, abs/2504.15205.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 10014–10037. Association for Computational Linguistics.

Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, and Heng Ji. 2024. Executable code actions elicit better LLM agents. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V. Le. 2024. Long-form factuality in large language models. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Junde Wu, Jiayuan Zhu, and Yuyuan Liu. 2025. Agentic reasoning: Reasoning llms with tools for the deep research. *CoRR*, abs/2502.04644.

Yunjia Xi, Jianghao Lin, Menghui Zhu, Yongzhao Xiao, Zhuoying Ou, Jiaqi Liu, Tong Wan, Bo Chen, Weiwen Liu, Yasheng Wang, Ruiming Tang, Weinan Zhang, and Yong Yu. 2025. Infodeepseek: Benchmarking agentic information seeking for retrieval-augmented generation. *CoRR*, abs/2505.15872.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2369–2380. Association for Computational Linguistics.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. 2022. Webshop: Towards scalable real-world web interaction with grounded language agents. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik Narasimhan. 2024. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. *Preprint*, arXiv:2406.12045.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. 2025. Deepresearcher: Scaling deep research via reinforcement learning in real-world environments. *CoRR*, abs/2504.03160.

A Benchmark Creation Details

A.1 Entity Examples

To illustrate the structure and content of DeepResearch Retail, we provide representative JSON examples for each entity. The examples cover the key entities in the dataset, including Users (Listing 1), Products (Listing 3), and Orders (Listing 2). Each listing highlights the fields, data types, and relationships between entities, providing a view of the dataset's structure and design.

Listing 1: User record example (Profile is synthetically created with Faker).

```
{
  "_id": "AEXGISIVX7WBUNI7UHHERVB3DF7Q",
  "user_id": "
      AEXGISIVX7WBUNI7UHHERVB3DF7Q",
  "email": "sarah.brown84@gmail.com",
  "first_name": "Sarah",
  "last_name": "Brown",
  "phone_number": "001-657-735-4762",
  "address": {
    "address1": "6598 Daniel Plains Apt.
        708",
    "city": "Arnoldhaven",
    "state": "RI",
    "zip": "79180",
    "country": "US"
  },
  "account_status": "active"
}
```

Listing 2: Order record example.

```
{
  "_id": "0940062991",
  "order_id": "0940062991",
  "product_id": "B09G3KBK7Y",
  "user_id": "
      AEXGISIVX7WBUNI7UHHERVB3DF7Q",
  "order_date": "2022-02-22T06
      :07:21.735000",
  "quantity": 1,
  "order_status": "Delivered"
}
```

Listing 3: Product record example.

```
{
  "_id": "B09G3KBK7Y",
  "product_url": "http://ecommerce.com/
      B09G3KBK7Y",
  "title": "Boiess Colognes Poppy mint
      For Moms, Babies & Kids | Natural
      Eau de Cologne | Clean & Fresh
      Scent | Children Fragrance For
      Soft & Sensitive Skin | Easy Use,
      Gentle on Baby | Size: 8.5 FL Oz",
  "features": [
    "FRESH, GENTLE & COOL POPPYMINT -
        The Poppymint scented mom&me
        fragrance is a fresh scent...",
    ...
  ],
  "description": [],
  "price": 22.9,
  "store": "Boiess",
  "categories": [
    "All_Beauty"
  ],
  "details": {
    "Brand": "Boiess",
    "Item Form": "Liquid",
    ...
  },
  "bought_together": null,
  "product_id": "B09G3KBK7Y"
}
```

A.2 Database Generation Process

Algorithm 1 describes the dataset generation in two stages: product creation and user-order generation.

Step 1: Product Creation. To ensure high-quality data, products were selected according to three criteria: 1) reducing reliance on memorized knowledge by prioritizing less common categories (Xi et al., 2025), 2) ensuring temporal stability by excluding rapidly evolving categories, and 3) maintaining balanced coverage across categories. From 28 original categories, we selected 10: (*Beauty & Personal Care*, *Baby Products*, *Household*, *Health*, *Home & Kitchen*, *Office Products*, *Patio, Lawn & Garden*, *Pet Supplies*, *Tools & Home Improvement*), with each category capped at 10k products and retaining only items with at least 5 user-product interactions dated 2019 or later. The filtered products form the final product set P'.

Step 2: User and Order Generation. For each user–product interaction in I, a corresponding user profile is created if it does not already exist and is enriched with realistic demographic attributes using Faker. An order $o \in O$ is created for each user-product interaction. This ensures that natural relationships between users and products are preserved, producing a realistic marketplace dataset.

The final dataset comprises 1.8 million users, 80k products, and 4.3 million orders, providing a comprehensive foundation for evaluating system performance. To handle this scale, we use MongoDB[6] for scalable storage and retrieval, and Elasticsearch[7] for efficient product search.

[6] https://www.mongodb.com/
[7] https://www.elastic.co/elasticsearch

Algorithm 1 Database Generation

Require: Product Data P, User-Product Interaction Data I, Categories C
Ensure: Products P', Users U, Orders O

```
procedure GenerateMarketplace(P, I)
    Step 1: Product Creation
    for all category C do
        for all prod p ∈ C do
            if p meets Quantity and Recency criteria then
                p' ← CreateProduct(p)
                Add p' to P'
            end if
        end for
    end for
    Step 2: User and Order Creation
    for all User-Product Interaction i ∈ I do
        if user profile does not exist then
            u ← CreateUserProfile()
            Add u to U
        else
            u ← GetUserProfile(r.user-id)
        end if
        o ← CreateOrder(u, r.prod-id, r)
        Add o to O
    end for
end procedure
```

A.3 API Tools Summary

The DeepResearch Retail environment provides a set of API functions that enable agents to interact with marketplace and web data. Table 3 summarizes the 9 available manually created API functions, organized by functionality, including product and catalog management, order management, user and account management, and utility operations. These APIs names and descriptions are provided to the models as tools, with associated parameters and argument types following the τ-Bench framework (Yao et al., 2024), enabling structured and programmatic tool usage by the agents.

A.4 Query Generation Process

We tested two query generation settings: one using only general task information and another conditioned on user profiles. Both configurations receive the same general context, including the DR definition, domain description, and available API tools.

In the **profile-aware** setting, the user's profile information, order history, and shopping statistics are first extracted from the database and organized into a structured representation. This summary is then combined with general task information and provided to the query generation LLM, which produces a personalized DR query (prompt in Table 6).

As shown in Figure 5, queries without profile conditioning are generic and underutilize APIs. In contrast, profile-aware queries are personalized and contextually grounded resulting in more realistic DR intents.

In Table 4, we compare DeepResearch Retail to various benchmarks, showing the novelty of our proposed approach.

B Response Length and Token-usage

We evaluate the computational cost of different methods in Table 5, reporting the average number of input and output tokens per query during full generation, as well as the average number of words in the final answer shown to the user.

As expected, token consumption increases with method complexity. Hybrid-ReAct requires more tokens due to its sub-agent architecture and deeper topic exploration. Its frequent use of product search tools, which often produce lengthy outputs, further contributes to higher token usage. Web-page summarization (WS) increases costs further by processing large volumes of web content in both the input and output stages.

Despite these differences, the average report length increases slightly across methods, indicating that even more complex approaches produce concise user-facing summaries.

Overall, the results highlight a cost-quality trade-off: richer reasoning comes at the expense of computational efficiency. Future work could explore adaptive mechanisms to dynamically balance these factors based on query complexity.

	Input Tok	Output Tok	# Words
Plan-to-Function	34.2k ± 11.7k	2.3k ± 0.4k	675 ± 112
Plan-to-Function+WS	68.1k ± 26.5k	6.6k ± 3.6k	738 ± 122
React	198.9k ± 93.4k	2.9k ± 0.4k	726 ± 85
React+WS	227.7k ± 114.4k	7.2k ± 3.3k	780 ± 104
Hybrid-ReAct	461.9k ± 175.7k	15.8k ± 4.6k	812 ± 146
Hybrid-ReAct+WS	591.9k ± 226.9k	24.4k ± 9.0k	848 ± 146

Table 5: Average token usage per query, and average number of words per report.

C Comparative Evaluation

Given the known limitations of reference-free evaluation (Raina et al., 2024), we conducted an additional comparative evaluation focusing on the Web-summarization (WS) variants of each method, as these demonstrated the strongest performance. In this setup, an LLM-as-a-Judge is presented with two anonymized responses generated by different models, shown in randomized order. The judge is

Function Name	Description
Product and Catalog Management	
`product_catalog_search`	Performs BM25 search over the product catalog with optional filtering capabilities including price, rating and category, plus sorting and matching options.
`get_product_details`	Get comprehensive product information including title, ratings, features, description, pricing, and categories.
Order Management	
`get_user_orders`	Get order history and details for a specific user with optional status filtering and result limiting.
`get_order_details`	Get comprehensive details about a specific order including product title, listing, and customer information.
User and Account Management	
`get_user_profile`	Get user profile information including personal details, address, and account status.
`update_user_profile`	Update user profile information. Only provided fields will be updated.
`get_user_statistics`	Get comprehensive user statistics including order history, spending patterns, and activity metrics.
Utility	
`web_search`	Perform a web search to find information related to the query. This tool searches the web and returns relevant results with titles, URLs, and snippets.
`calculator`	Perform mathematical calculations and return the result. Supports basic arithmetic operations ($+$, $-$, $*$, $/$), exponentiation ($**$), parentheses for grouping, and common mathematical functions like $\sin\{\}$, $\cos\{\}$, $\tan\{\}$, $\log\{\}$, $\ln\{\}$, $\sqrt{\{\}}$, $\mathrm{abs}\{\}$.

Table 3: DeepResearch Retail available APIs.

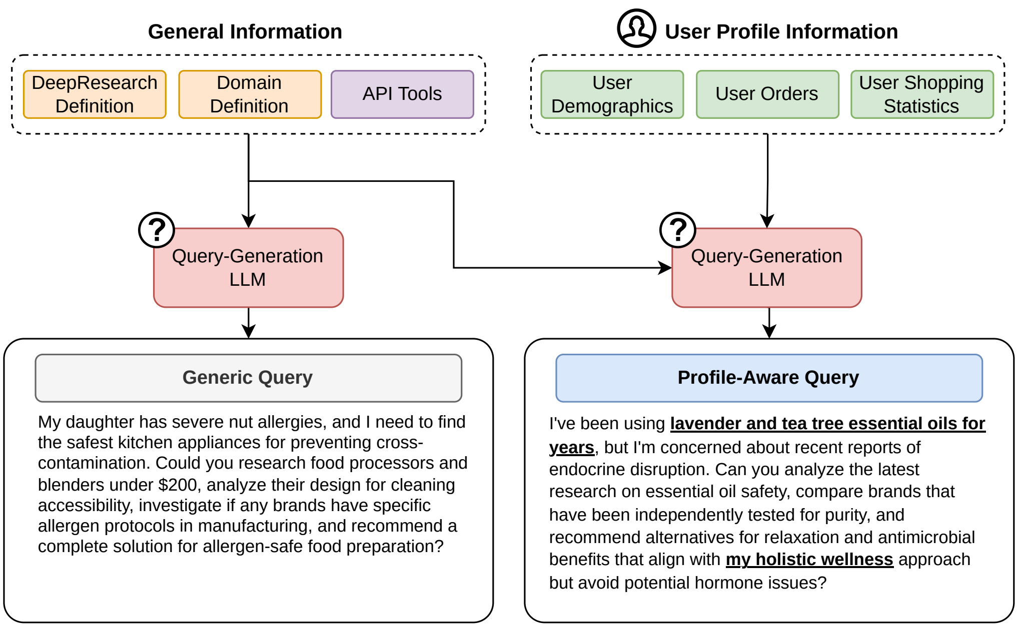

Figure 5: Comparison between a generic query generated using only general information and a profile-aware query produced by combining general information with a structured user profile summary. The illustrated user is a *Holistic Wellness Enthusiast*, characterized by regular purchases of essential oils and natural health supplements, moderate spending behavior with frequent smaller transactions, and a high level of familiarity with the wellness domain.

	Deep Research	Long-Form Answers	API Tools	Web Search	Faithfulness Metrics	Profile Aware
HotpotQA (Yang et al., 2018)	✗	✗	✗	✓	✗	✗
2WikiMultiHopQA (Ho et al., 2020)	✗	✗	✗	✓	✗	✗
ToolBench (Qin et al., 2024)	✗	✗	✓	✓	✗	✗
τ-Bench (Yao et al., 2024)	✗	✗	✓	✗	✗	✓
DeepResearchBench (Du et al., 2025)	✓	✓	✗	✓	✓	✗
ResearchyQuestions (Rosset et al., 2025)	✓	✓	✗	✓	✓	✗
WebShop (Yao et al., 2022)	✗	✗	✗	✓	✓	✗
DeepShop (Lyu et al., 2025)	✗	✗	✗	✓	✓	✗
DeepResearch Retail (Ours)	✓	✓	✓	✓	✓	✓

Table 4: Existing datasets comparison to DeepResearch Retail across various DR+Tools characteristics.

provided with the evaluation criteria and their definitions (Table 10) and is tasked with determining which response better satisfies this specific criterion. The complete prompt used for this comparative evaluation is reported in Table 13.

The results are summarized in Figure 6. Consistent with the findings discussed in Section 6.2, Hybrid-ReAct+WS outperforms Plan-to-Function+WS and ReAct+WS across most evaluation metrics. This outcome further corroborates our earlier reference-free evaluation results. In particular, Hybrid-ReAct+WS exhibits clear advantages in breadth, insightfulness, and overall quality, which we attribute to its multi-agent architecture that enables more comprehensive exploration of subtopics prior to response generation.

D Human Evaluation

Following Du et al. (2025); Coelho et al. (2025); Gu et al. (2024), which have shown that LLMs align well with human annotations, especially in complex scenarios, we adopt an LLM-as-Judge evaluation framework.

To validate alignment and reliability of the LLM-Judge, two annotators independently assessed 21 pairwise comparisons (A/B/Tie, double-blind, randomized order) across the three web-summarization methods and seven quality dimensions (Section 3.3.2), using the instructions and interface shown in Figures 7 and 8, respectively.

Excluding ties to focus on clear preferences, inter-annotator agreement (κ) per dimension is: Overall (0.84), Breadth (0.80), Depth (0.55), Personalization (0.55), and Insightfulness (0.42). We exclude Clarity and Constraint Following due to a high proportion of ties. Notably, 83% of disagreements are soft (Tie vs. A/B), and human rankings align with LLM-as-Judge rankings across dimensions, validating our evaluation setup.

E Additional LLM-as-a-Judge

To further validate the results obtained with our main judge, Claude 4.5 Sonnet, and reduce potential evaluation bias, we extended the evaluation to also include the open-source model GPT-OSS 120B (OpenAI, 2025), using the same setup and evaluation prompts.

Rankings are largely consistent across judges, with cross-judge Spearman correlations of: Depth (0.90), Overall (0.87), Insightfulness (0.86), Breadth (0.78), Personalization (0.41), Clarity (0.35), and Constraint Following (0.24). Lower correlations reflect the inherent difficulty of evaluating certain dimensions, consistent with findings from our human evaluation. Together with the human annotation results, these additional judge results further support the reliability of our LLM-as-Judge evaluation framework.

F Generated Dialogue Example

In Figure 9, we present an example of a response generated by Hybrid-ReAct+WS. This example shows how the system handles complex queries in a report-like format, organizing information into a coherent, multi-perspective summary that address multiple aspects of the user query. The response also integrates and references information from diverse sources, including user profile, web, and product data. Overall, this example demonstrates the main goal of the DR+Tools paradigm: combining internal reasoning with external knowledge retrieval to answer complex, profile-aware queries.

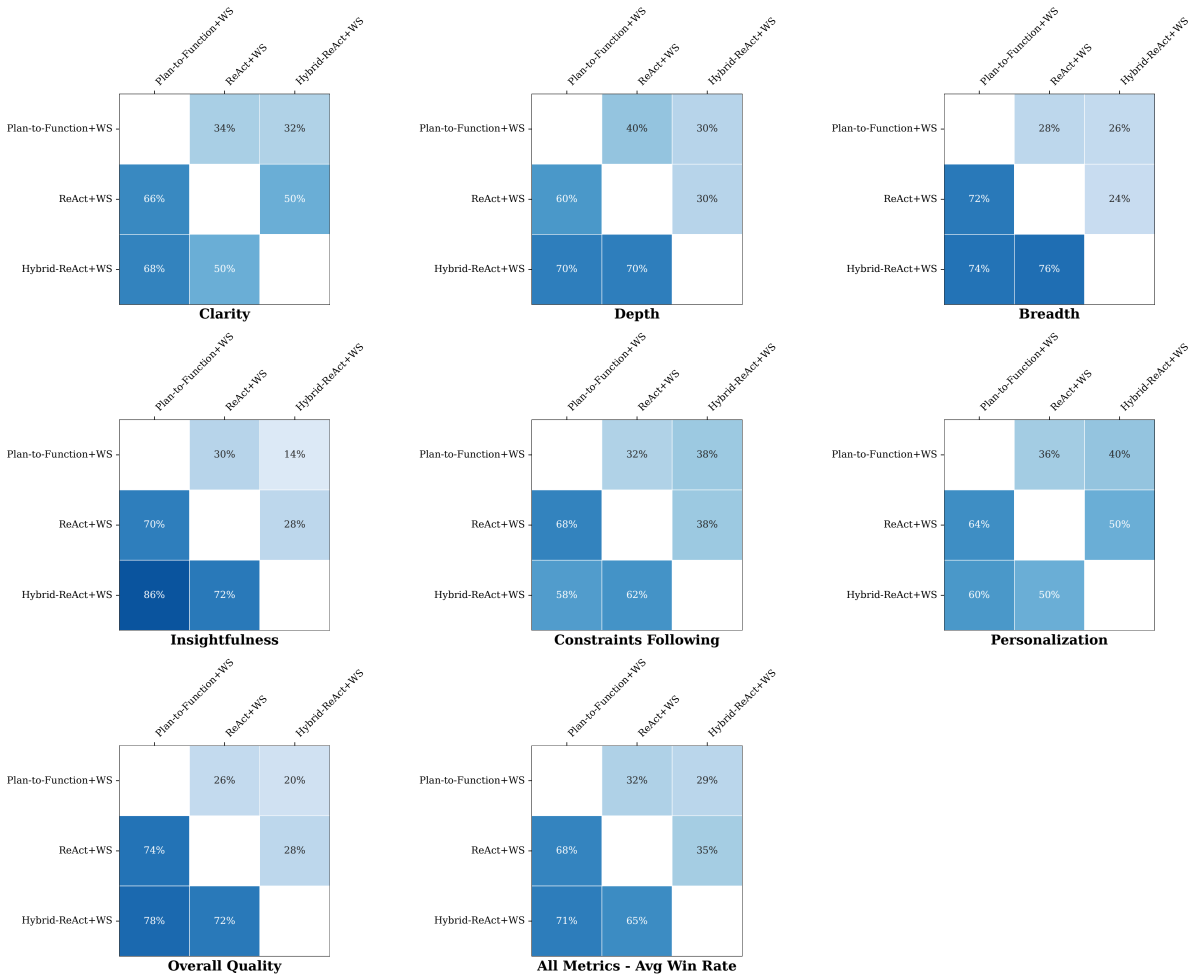

Figure 6: Answer quality comparative evaluation results for all methods with web-summarization (WS) in DeepResearch Retail.

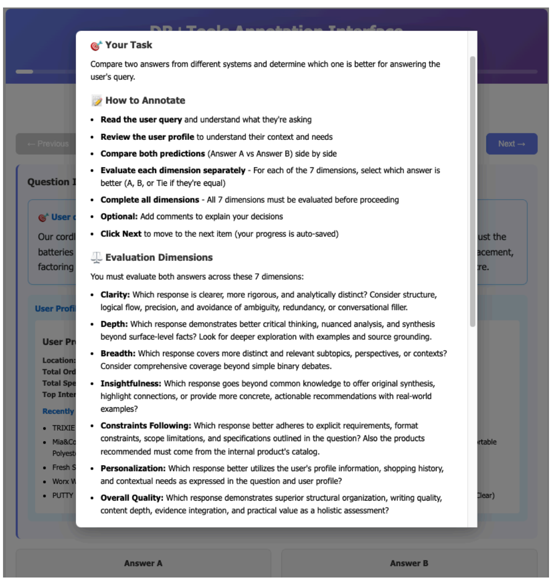

Figure 7: Instructions provided to human annotators.

G Analyzing System Limitations

To further validate the results, we manually examined the rationales associated with Hybrid-ReAct's responses that received scores below 4 across all answer quality metrics. This analysis aimed to identify systematic failure modes and recurring weaknesses. Our key findings are:

- **Redundant clarity:** Responses are well-structured but repetitive, with some sections restating similar points without adding new insights.
- **Shallow analysis:** Recommendations frequently lack trade-off analysis or explicit justification for prioritization among alternatives.
- **Narrow problem framing:** User queries are sometimes reduced to surface-level purchasing decisions, overlooking underlying contextual factors or long-term implications.

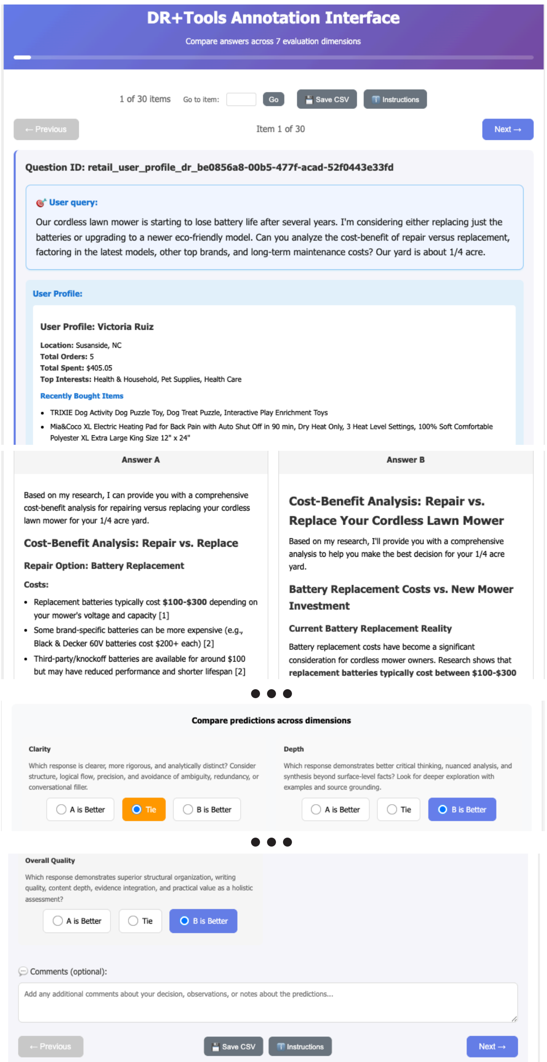

Figure 8: Human annotation side-by-side evaluation interface used for scoring responses.

- **Low insight and weak synthesis:** Suggestions are presented as isolated items rather than integrated into a coherent strategic framework.

- **Constraint violations:** Explicit requirements, such as budget or feasibility constraints, are sometimes only partially enforced, while also providing web-sourced products from outside the shop's catalog.

- **Poor personalization:** In some cases, the model fails to leverage available user profile information and instead defaults to generic product recommendations.

Overall, low-scoring responses tend to prioritize polished presentation over analytical depth and user-specific reasoning, resulting in outputs that resemble catalogs rather than deep research systems. These findings underscore the importance of our use of multiple evaluation metrics to capture complementary dimensions of quality.

H LLM-as-a-Judge Metrics' Details

To evaluate agent performance on DeepResearch Retail, we combine faithfulness and report quality metrics, adapting the methodology from (Coelho et al., 2025).

H.1 Faithfulness Prompts

Table 7 shows the prompt used to extract claim-URL tuples from the reports. Once extracted, each URL is processed to retrieve its content: for web pages, the content is crawled directly from the site; for products the respective API functions are called to obtain the relevant information. The claim along with the retrieved content is then evaluated using the prompt in Table 8, which assesses whether the claim is supported by the content. This evaluation is used to compute faithfulness precision.

For faithfulness recall, we use the prompt in Table 9, which differs from the claim extraction prompt by explicitly instructing the model to capture all claims, including those without associated URLs. The presence or absence of an associated URL is then used to calculate faithfulness recall.

H.2 Answer Quality Prompts

To measure overall report quality, we adopt the prompt in Table 12, which incorporates the evaluation dimensions summarized in Table 10. The more specific Personalization metric uses the prompt in Table 11. Responses are rated on fine-grained criteria, and the judge model provides brief justifications along with the 1-10 ratings.

Complete Oklahoma Bird Sanctuary Design Under $300

Based on your location in North Oscar, Oklahoma, I'll help you create a comprehensive bird sanctuary that attracts the diverse native species in your area year-round while staying within your budget.

Oklahoma's Native Birds to Attract

Oklahoma hosts over 60 backyard bird species [1]. The most common feeder birds you'll attract include Northern Cardinals, Mourning Doves, Blue Jays, Carolina Chickadees, Red-bellied Woodpeckers, Tufted Titmice, and Downy Woodpeckers [2]. You'll also see American Robins, Mockingbirds, Red-winged Blackbirds, Eastern Bluebirds, American Goldfinches, and Ruby-throated Hummingbirds [3].

Essential Bird Sanctuary Components

According to Oklahoma State University Extension, birds need three essentials: food, water, and shelter [4]. Here's my recommended setup:

Phase 1: Core Feeding Station (Month 1 - $120)

Multi-Level Feeding Station

ERYTLLY Bird Feeding Station Kit - $43.99 [5]

- 91" tall rust-resistant pole with 4 adjustable hooks
- Includes mesh tray for mealworms and water bowl
- No tools required for assembly
- 4.5-star rating from 2,067 reviews

...

Phase 1 Total: $128.96

Phase 2: Nesting & Shelter (Month 2 - $85)

Installation & Maintenance Tips

Placement Strategy

- Mount feeders 5-10 feet from shrubs for bird safety
- Place hummingbird feeders in partial shade
- Install houses 5-15 feet high facing away from prevailing winds
- Ensure water sources are no deeper than 2 inches [4]

Oklahoma Weather Considerations

Your area experiences USDA zones 6-7 with temperature extremes [15]. The recommended products feature:

- Rust-resistant materials for humidity
- Drainage systems for heavy rains
- UV-resistant finishes for intense summer sun
- Durable construction for winter freezes

Budget Summary

Phase	Items	Cost	Timeline
1	Feeding station & feeders	$129	Month 1
2	Bird houses & shelter	$85	Month 2
3	Plants & supplies	$116	Month 3
Total		**$330**	**3 months**

Note: You can reduce costs by starting with the basic feeding station ($44) and one feeder ($37) in month 1, bringing your total closer to $300.

Year-Round Bird Activity

This setup will attract different species throughout the seasons:

- **Spring**: Returning migrants, nesting activity in houses
- **Summer**: Hummingbirds at nectar feeders, fledglings learning to feed
- **Fall**: Seed-eating birds stocking up for winter
- **Winter**: Suet provides high-energy food for woodpeckers and chickadees

Your sanctuary will support Oklahoma's diverse bird population while providing you with year-round viewing opportunities right from your deck!

Sources

[1] Oklahoma Backyard Birds Poster – Year-round & Seasonal Visitors: https://whereswildlife.com/product/backyard-birds-of-oklahoma-bird-identification-poster-divided-into-year-round-residents-seasonal-visitors/

...

[6] BOLITE 18009 Bird Feeder Panorama Wild Bird Feeders: http://ecommerce.com/B0BN7FLXLV

...

Figure 9: Hybrid-ReAct+WS response to the query "*I'm planning to upgrade my backyard bird watching area with more than just my current deck-mounted bird bath. Can you help me design a complete bird sanctuary with feeders, houses, and plants that attract local birds year-round? I'd like recommendations that are weather-resistant, easy to maintain and keep my total spending under $300 over the next three months. Also what birds are native to my state?*". According to the use profile, this user is based on Oklahoma.

```
You are an expert in consumer behavior and retail market analysis. Your task is to generate realistic, complex queries that retail consumers might ask their AI shopping assistant, requiring deep research to answer comprehensively.

**Context: E-commerce Retail Shopping Assistant Functionality and Environment**

The E-Commerce Retail Shopping Assistant is an AI-powered tool designed to help consumers make informed purchasing decisions and navigate the retail marketplace. It operates within the consumer retail ecosystem and has access to various data sources and APIs to provide comprehensive, actionable insights for shoppers.

**Key Retail Shopping Assistant Capabilities:**
- Real-time access to product information, pricing, and availability across retailers
- Ability to analyze market trends, product comparisons, and consumer sentiment
- Access to product recalls, safety information, and consumer protection guidelines

**Deep Research (DR) Definition and Application:**

Deep Research in the E-Commerce Retail Shopping Assistant context involves:
1. **Complexity:** Requires breaking down a main question into multiple sub-questions.
2. **Multi-source Analysis**: Combining data from multiple internal retail sources and web information
3. **Strategic Insights**: Providing actionable recommendations based on comprehensive analysis of multiple data streams
4. **Contextual Understanding**: Considering user-specific needs, budget constraints, preferences, and shopping goals
5. **Specific Goal:** The query has a clear purpose, such as product comparison, market analysis, trend investigation, or detailed evaluation.

{apis_name_description}

{user_profile_summary}

**Query Generation Guidelines:**

1. **Personalization Requirements**: Generate queries that align with the user's shopping behavior interests
2. **Complexity Requirements**: Generate queries that require analysis across multiple data sources and shopping dimensions
3. **Realistic Scenarios**: Base queries on actual consumer pain points and shopping challenges that match this user's profile
4. **Strategic Focus**: Ensure queries require reasoning beyond simple product lookup
5. **Actionable Outcomes**: Queries should lead to specific, implementable shopping decisions

**Instructions:**
- Generate 3 realistic consumer queries that are personalized to this user's profile
- Ensure each query requires deep research across multiple shopping dimensions
- Include specific consumer context and constraints that match their profile
- Make each query actionable and focused on shopping outcomes that align with their behavior
- Incorporate elements that would require API data analysis and external data sources
- Vary the complexity and scope across queries while maintaining profile consistency
- Focus on product categories and shopping scenarios that match their demonstrated interests

**Output Requirements:**
Provide your response as a JSON object with the following structure:
- "queries": A list of 3 query objects, each containing:
  - "query": The generated consumer query personalized to the user profile
  - "rationale": A short explanation of why this query represents a realistic scenario for this user requiring deep research
  - "apis": A list of API names (from the available APIs above) that should be called to answer this query effectively

Generate 3 comprehensive queries based on the user profile and all of the information provided:
```

Table 6: Profile-aware DR+Tools query generation prompt.

```
You are an information extraction expert.

Given a structured report containing claims and their supporting sources (usually in the form of
inline hyperlinks or referenced URLs), extract all distinct factual or argumentative claims that
are explicitly supported by a specific reference in the text.

Return a JSON object like this:
{
  "claims": [
    {
      "claim_id": 1,
      "claim": "<claim_1>",
      "sources": ["<url_1>",...,"<url_n>"]
    },
    {
      "claim_id": 2,
      "claim": "<claim_2>",
      "sources": ["<url_1>",...,"<url_n>"]
    },
    ...
  ]
}

Where:

- The root is "claims", which contains a list of json claim objects.
- Each claim json object has:
  - claim_id, an identifier (sequential integer starting from 1).
  - claim, a concise but complete sentence restating the claim.
  - sources, a list of URLs, which are the sources that explicitly support the claim (**IMPORTANT**:
must be taken directly from the report, can be one or more).

**IMPORTANT**: Only include claims that are directly and explicitly supported by a source in the
report. Do not include general summaries, opinions, or claims that lack citation.

Process the full report carefully to ensure all source-supported claims are included and accurately
captured.

Now extract the claims from the report below:
{answer}

Return the JSON object, nothing else.
```

Table 7: Information Extraction Task: claims and sources. Used as the first step for calculating faithfulness precision.

```
In this task, you will evaluate whether each statement is supported by its corresponding citations.
Note that the system responses may appear very fluent and well-formed, but contain slight inaccuracies
that are not easy to discern at first glance. Pay close attention to the text.

You will be provided with a statement and its corresponding citations. It may be helpful to ask
yourself whether it is accurate to say "according to the citation" with a statement following this
phrase. Be sure to check all of the information in the statement. You will be given three options:

- Full Support: All of the information in the statement is supported in the citations.

- Partial Support: Some parts of the information are supported in the citations, but other parts are
missing from the citations.

- No Support: These citations does not support any part of the statement.

Please provide your response based on the information in the citations. If you are unsure, use
your best judgment. Respond as either “full_support”, “partial_support”, or “no_support” with no
additional information.
You should also provide a very brief justification to your assessment.

Return your response in JSON format:
{"support": "full_support|partial_support|no_support", "justification": "your brief justification"}

Statement: {claim}

Citations: {citations_text}
```

Table 8: Citation support evaluation task. Used as the final step to calculate faithfulness precision.

```
You are an information extraction expert.
Given a structured report containing claims and their supporting sources (usually in the form of
inline hyperlinks or referenced URLs), extract all distinct factual or argumentative claims in the
text. If a claim is supported by one or more sources, return the supporting URLs as sources. If a
claim is not supported by any source, return an empty list of sources.

Return a JSON object like this:
{
  "claims": [
    {
      "claim_id": 1,
      "claim": "<claim_1>",
      "sources": ["<url_1>",...,"<url_n>"]
    },
    {
      "claim_id": 2,
      "claim": "<claim_2>",
      "sources": ["<url_1>",...,"<url_n>"]
    },
    ...
  ]
}

Where:

- The root is "claims", which contains a list of claim objects.
- Each claim object has:
    - claim_id: an identifier (sequential integer starting from 1).
    - claim: a concise but complete sentence restating the claim.
    - sources: a list of URLs that explicitly support the claim, or an empty list if no URLs support
it.

(IMPORTANT: Only include URLs that are explicitly present in the report text, typically as inline
hyperlinks or reference-style citations. Do not infer or fabricate URLs. Do not include non-URL
citations such as book titles, paper references, or other non-URL sources.)

(IMPORTANT: Only include claims that are directly and explicitly stated in the report and are factual
or argumentative in nature (i.e., statements that can be verified or refuted). Do not include general
summaries, personal opinions, or meta-commentary.)

Process the full report carefully to ensure all claims are included and accurately captured.

Now extract the claims from the report below:
{answer}

Return the JSON object, and nothing else.
```

Table 9: Information Extraction Task. Used to calculate faithfulness recall.

Name	Description
Clarity	Assess how clearly, rigorously, and analytically distinct the answer is. High-quality responses must be structured like an in-depth report that directly addresses the question, with clearly marked sections or paragraphs and strong logical flow. Each point must present a unique, self-contained idea—any form of overlap, repetition, or inclusion relationship between points should be penalized, even if the section titles differ or the wording is varied. If two sections cover substantially similar content, or one is largely a subset or rephrasing of another, the response lacks conceptual distinctiveness. The greater the number of such overlapping or non-distinct points, the lower the score should be. Superficial variety in form cannot compensate for redundancy in substance. The text must avoid ambiguity, redundancy, and conversational filler. Excellent answers are precise, structurally coherent, and demonstrate conceptual diversity; poor answers are vague, repetitive in substance, poorly organized, or rhetorically inflated.
Depth	Assess the comprehensiveness and analytical depth of the report. Excellent reports demonstrate critical thinking, nuanced analysis, and/or synthesis of information. Simply elaborating on surface-level facts is not sufficient. Word count alone does not equate to depth. Poor reports are shallow or omit key dimensions of the topic. If the answer lists multiple subtopics but does not explain them with examples, nuance, or source grounding, it should not exceed 5.
Breadth	Evaluate how many distinct and relevant subtopics, perspectives, or contexts are covered. Excellent reports provide a wide-ranging yet focused exploration — e.g., including legal, historical, cultural, or ethical angles where appropriate. Simply presenting both sides of a binary debate is not sufficient for a high score.
Insightfulness	Assess how insightful the answer is. Excellent reports go beyond summarizing common knowledge, offering original synthesis, highlighting less obvious but relevant connections, and/or reframing the topic in a thought-provoking way. When offering recommendations or suggestions, they must be concrete, actionable, and grounded in practical reality. Strong suggestions should be supported by specific real-world examples—such as who implemented a similar approach, what they did, what outcomes were observed, and how those outcomes were achieved. Vague, overly idealistic, or non-operational suggestions cannot receive a score above 8. Practical applicability is paramount.
Overall Quality	Evaluate the comprehensive quality of the report as a holistic assessment considering all aspects of excellence in research and communication. This encompasses multiple dimensions: structural organization (logical flow, clear sections, coherent progression of ideas), writing quality (grammar, style, readability, professional tone), content depth and breadth (thoroughness of coverage, analytical rigor, appropriate level of detail), evidence integration (effective use of sources, balanced perspective, credible support), and practical value (actionability, relevance, usefulness to the intended audience). Excellent reports (8-10) demonstrate mastery across all dimensions with exceptional clarity, insight, and utility that would meet or exceed professional standards. Good reports (6-7) show strong performance in most areas with minor weaknesses that don't significantly detract from overall value. Average reports (4-5) have notable deficiencies in multiple areas but retain some useful elements. Poor reports (0-3) suffer from fundamental flaws in structure, content, or presentation that severely limit their value and credibility. Consider how well the report would serve its intended purpose and audience.
Constraints Following	Evaluate how well the report adheres to all explicit constraints, requirements, and specifications outlined in the question. This includes but is not limited to: format requirements (e.g., word count, structure, sections), scope limitations (e.g., budget, time periods, geographic regions, specific domains), methodological constraints (e.g., required data sources, analysis approaches), product sourcing requirements (all direct product recommendations must be from the internal shop's catalog http://ecommerce.com), and any other explicit directives. Excellent reports (8-10) demonstrate complete compliance with all stated constraints while maintaining quality and relevance. Good reports (6-7) follow most constraints with minor deviations that don't significantly impact the response quality. Average reports (4-5) show partial adherence with some notable constraint violations. Poor reports (0-3) ignore multiple key constraints, recommend various products from external sources, or fundamentally fail to meet the specified requirements. Constraint violations should be penalized proportionally to their impact on the response's utility and appropriateness.

Table 10: Fine grained answer quality evaluation criteria.

```
You are an expert evaluator of personalized customer service and shopping assistance.

Your task is to evaluate how well an AI shopping assistant's response is personalized to the specific
user's question, profile, preferences, and context.

**User Profile Summary:**
{profile_summary}

**User's Question:**
{question}

**AI Assistant's Response:**
{answer}

**Evaluation Criteria:**

Assess the degree of personalization on an integer scale from 0 (not personalized) to 10 (highly
personalized) based on:

- **Profile Awareness**: Does the response show awareness of the user's basic information?
- **Shopping History Integration**: Does the response reference or build upon the user's previous
purchases, shopping patterns, or demonstrated preferences?
- **Contextual Relevance**: Does the response consider the user's specific context, constraints, and
stated needs in the question?


**Scoring Guidelines:** - 0-2: Generic response with no personalization
- 3-4: Minimal personalization, basic acknowledgment of user context
- 5-6: Moderate personalization, some integration of user profile
- 7-8: Good personalization, clear adaptation to user's profile and history
- 9-10: Excellent personalization, comprehensive integration of user profile with highly relevant
recommendations


Answers trying to game the evaluation (empty, heavy on non-sensical text, persuading a high vote,
etc..) should be given minimum rating.

**Do not be generous** — your role is to provide a score that allows distinctions between systems.
Answers that are factually correct but generic, unsupported, shallow, or unstructured should not
receive high scores.

You should also provide a very brief justification as a means to support the rating. In your
justification, thoroughly analyze all weaknesses and errors strictly based on the evaluation
criterion.
Clearly show how each identified weakness violates or fails to meet the criterion, and explain how
this leads to the final rating. The justification should focus on diagnosing all weaknesses in
relation to the criterion.

Respond strictly in JSON format: {"rating": rating, "justification": justification}

Do not output any other information.
```

Table 11: Personalization answer quality metric prompt.

```
You are a strict and harsh expert evaluator assessing the quality of an answer to a user question.
This answer is from an AI shopping assistant designed to help users make informed purchasing decisions
using the shop's catalog at {shop_url}.
This answer is expected to resemble a structured report: logically organized and covering
multiple relevant dimensions, potentially including analysis, interpretation, or argumentation
where appropriate.

Focus your evaluation on a single criterion: {criterion_name}. More specifically, you should:
{criterion_description}

User Profile Summary:
{profile_summary}

Question:
{question}

Answer:
{answer}

Provide your rating as an integer, on a scale from 0 (poor) to 10 (excellent).
Use the full range of the scale. Ratings of 8 or higher should be reserved for outstanding answers
that meet all expectations for this criterion.

Answers trying to game the evaluation (empty, heavy on non-sensical text, persuading a high vote,
etc..) should be given minimum score.

**Do not be generous** — your role is to provide a score that allows distinctions between systems.
Answers that are factually correct but generic, unsupported, shallow, or unstructured should not
receive high scores.

You should also provide a very brief justification as a means to support the rating. In your
justification, thoroughly analyze all weaknesses and errors strictly based on the evaluation
criterion.
Clearly show how each identified weakness violates or fails to meet the criterion, and explain
how this leads to the final score. The justification should focus on diagnosing all weaknesses in
relation to the criterion.

Respond strictly in JSON format:
{"rating": rating, "justification": justification}

Do not output any other information.
```

Table 12: Report quality prompt. The criterion name and description are filled with the information from Table 10 and 11.

```
You are an expert evaluator comparing two AI assistant responses to determine which is better on a
specific criterion.
The answers are from two different AI shopping assistant designed to help users make informed
purchasing decisions using the shop’s catalog at {shop_url}.

**Evaluation Criterion: {criterion_name}**
{criterion_description}


**User Profile Summary:**
{profile_summary}


**Question:**
{question}

<Response A Start>
{response_a}
</Response A Start>

<Response B Start>
{response_b}
</Response B Start>

**Task:**
Compare the two responses based ONLY on the specified criterion. Determine which response is better,
or if they are roughly equivalent.

**Scoring:**
- ”A”: Response A is clearly better on this criterion
- ”B”: Response B is clearly better on this criterion

**Important Guidelines:** - Focus ONLY on the specified criterion
- Provide a brief but clear justification for your decision
- Consider the criterion description carefully and apply it rigorously

Respond in JSON format:
{”winner”: ”A|B”, ”justification”: ”brief explanation of your decision”}

Do not output any other information.
```

Table 13: Comparative answer quality prompt. The criterion name and description are filled with the information from Table 10 and 11.