Benchmarking Models for Conversational E-Commerce: A Reproducible Evaluation Framework

Yuri Vorontsov[1], Diogo S. Carvalho[1], Anastasia Vorontsov[1], Anna Platonova[1], Vladimir Gorovoy[1], Ilya Briskin[1], Felix Tseitlin[1], Senka Krivic[1,2] and Salman Ahmad[1]

[1]*Rezolve Ai Labs, United Kingdom*
[2]*University of Sarajevo, Bosnia and Herzegovina*

Abstract

We present a reproducible framework for benchmarking large language models as multi-turn conversational e-commerce agents across the full shopping loop of search, clarification, recommendation, and add to cart. We demonstrate the framework in a first-in-class 12-model study, complemented by a smaller cross-category evaluation. The framework comprises product catalog ingestion, scenario generation grounded in real consumer search queries, a model-breaking scenario selection filter, LLM-driven customer simulation, and a five-rubric LLM judge covering intent understanding, clarification behaviour, recommendation quality, add-to-cart correctness, and grounding fidelity. We apply the framework to 12 models on a primary jeans case study (1,200 scored conversations across five runs and 20 scenarios), using it as a worked example of the shopping loop; these scores are not intended to generalise across e-commerce. We also conduct a cross-category study on three additional product domains under a different scenario protocol. Rank order on that smaller study is directional only ($\rho = 0.40$–0.70, $n = 5$). On the primary study, grounding fidelity is the main separator between score bands, Clarification is the least saturated rubric among ranks 1–7, and the highest-scoring models form overlapping bootstrap confidence bands once run-to-run variance is accounted for. These findings rest on a judge-reliability programme: a seven-judge cross-family panel with super-judge arbitration that reduces judge swing from 24.5% to 9.2%, validated against a preliminary human gold calibration (103 rubric cells from a single rater; 11 pass / 92 fail). GPT-4.1 alone is too lenient ($\kappa = -0.13$); the reliable stack reaches 88.3% accuracy against human labels, below an always-fail dummy at 89.3%, so Cohen's κ (0.273) is the more informative agreement figure. Benchmark code and data are at https://github.com/rezolved/conversational-commerce-benchmark-framework.

Keywords

conversational commerce, benchmark, large language models, evaluation, tool calling, recommender systems

1. Introduction

Retailers deploying LLM-based shopping assistants need to know which model to use, and whether it will perform well once customers are actually talking to it. This is harder to answer than it looks. A conversational commerce agent must sustain multi-turn dialogue, decide when to search a live catalog via tool calls, ask clarifying questions, recommend relevant items, and execute transactional actions such as adding a product to a cart, without inventing products, prices, or transactional outcomes that never occurred, a failure mode we previously characterised as *journey hallucination* [1]. Single-turn recommendation and product search are comparatively well studied, with established benchmarks; the conversational, tool-using, transactional layer is not, so teams end up comparing models informally or borrowing benchmarks built for a different problem. From a recommender-systems perspective, this matters because conversational commerce agents are the next generation of interactive recommendation interfaces: they turn the traditional search-rank-present pipeline into a multi-turn negotiation between user intent and system capability, and there is little available to evaluate that negotiation directly.

Several related lines of work come close but stop short. Conversational recommendation benchmarks

GenAIECommerce'26: The Third Workshop on Agentic and Generative AI for E-Commerce, co-located with RecSys, September 28, 2026, Minneapolis, MN, USA
yurivorontsov@rezolve.com (Y. Vorontsov); diogocarvalho@rezolve.com (D. S. Carvalho); anastasiavorontsov@rezolve.com (A. Vorontsov); annaplatonova@rezolve.com (A. Platonova); vladimirgorovoy@rezolve.com (V. Gorovoy); ilyabriskin@rezolve.com (I. Briskin); felixtseitlin@rezolve.com (F. Tseitlin); senkakrivic@rezolve.com (S. Krivic); salmanahmad@rezolve.com (S. Ahmad)


such as ReDial [2] and INSPIRED [3] model multi-turn dialogue, but predate the tool-calling paradigm and do not evaluate transactional actions. General-purpose agent benchmarks such as ToolLLM [4] and AgentBench [5] evaluate tool use broadly, but without the domain-specific structure—product catalogs, shopping intent, cart correctness—that determines whether a shopping assistant is actually useful. Shopping MMLU [6] captures e-commerce knowledge at scale, but in a single-turn, multiple-choice format that cannot exercise multi-turn dialogue. The closest prior work, RecBench+ [7], evaluates LLMs as personalised recommendation assistants on complex queries, but over static item sets rather than the live, multi-turn search, clarify, recommend, and transact loop a shopping assistant has to execute. We evaluate that loop directly.

Our objective is to benchmark LLMs on the full shopping loop of search, clarification, recommendation, and add to cart, and to make that evaluation reproducible. To our knowledge, this is the first multi-model study at this scale for conversational e-commerce agents. Its empirical benchmark and reusable evaluation framework are complementary contributions: the study demonstrates the framework, and the framework lets practitioners reproduce the evaluation on new product feeds. The framework comprises four components: (1) a structured product catalog derived from a public dataset [8], (2) a scenario generation pipeline grounded in real shopping intents mined from consumer search platforms, (3) an LLM-driven customer simulation agent that produces deterministic, multi-turn shopping conversations, and (4) a five-rubric evaluation—a contribution in its own right—scored by an LLM judge covering intent understanding, clarification behaviour, recommendation quality, add-to-cart correctness, and grounding—complemented by automatic behavioural metrics from conversation traces. We apply it to 12 models (five runs $\times$ 20 scenarios; 1,200 conversations) on a jeans case study, using it as a worked example rather than a general e-commerce leaderboard, and conduct a systematic judge-reliability programme to bound judge-induced uncertainty.

Empirically, we find that model scale does not reliably predict conversational e-commerce quality: the highest-scoring models span more than an order of magnitude in parameter count, and they form overlapping bootstrap confidence bands once run-to-run variance is taken into account. Grounding—whether an assistant's claims are actually backed by tool output—is the rubric that most separates higher- and lower-scoring assistants, and native tool-calling access is close to a prerequisite for acceptable performance at all. These findings, and where they do and do not generalise beyond a single product category, are detailed in Section 6.

In summary, our contributions are:

- A reproducible benchmark framework for multi-turn conversational e-commerce, comprising a product catalog, scenario generation pipeline, customer simulation agent, and tool-using assistant agent—the first public benchmark to cover this full shopping loop (code and data at https://github.com/rezolved/conversational-commerce-benchmark-framework).
- A five-rubric evaluation methodology, iteratively refined over multiple design cycles, that decomposes conversation quality into independently scored dimensions including a novel grounding rubric, together with complementary automatic behavioural metrics.
- A judge-reliability study with seven base judges and a three-model super-judge arbitration panel, decomposing evaluation noise and providing a preliminary human gold calibration—showing that a multi-judge reliable stack reaches 88.3% accuracy against an imbalanced single-rater gold set, below an always-fail dummy at 89.3% ($\kappa = 0.273$).
- The first large-scale empirical comparison of LLMs as conversational e-commerce agents: 12 models across five independent runs with bootstrap confidence intervals on a jeans worked example, with a smaller cross-category study under a different scenario protocol on three additional product domains.

The remainder of this paper is organised as follows. Section 2 reviews related work. Section 3 describes the benchmark design. Section 4 details the evaluation framework. Section 5 presents the experimental setup. Section 6 reports results. Section 7 discusses limitations, and Section 8 concludes.

2. Related Work

Our work sits at the intersection of four research threads: conversational recommendation benchmarks, LLM-as-judge evaluation, agent benchmarks with tool use, and hallucination detection in conversational commerce.

Conversational recommendation benchmarks. ReDial [2] introduced a crowd-sourced dataset of movie-recommendation dialogues and established the conversational recommendation task. INSPIRED [3] extended this with sociable recommendation strategies grounded in social science, and TG-ReDial [9] added topic-guided transitions to model natural conversational flow toward a recommendation. Together these established the datasets and tasks conversational recommendation builds on, from before the tool-calling paradigm existed and without the transactional layer—adding an item to a cart, confirming an order—that conversational commerce adds on top. RecBench+ [7], the closest prior work, evaluates LLMs on complex single-turn recommendation queries with explicit conditions, implicit reasoning, and user profiles, as a single-turn decision over a static item set rather than the multi-turn loop of search, clarify, recommend, and transact that defines conversational commerce. Shopping MMLU [6] covers 57 online shopping tasks across 20,000 questions derived from Amazon data, a complementary way to probe e-commerce knowledge at scale in a single-turn, multiple-choice format rather than the dialogue dynamics and tool interactions we target.

LLM-as-judge evaluation. G-Eval [10] demonstrated that GPT-4 with chain-of-thought prompting achieves higher correlation with human judgments than traditional NLG metrics, establishing the viability of LLM-based evaluation. MT-Bench [11] scaled this approach to multi-turn conversations and introduced the Chatbot Arena for pairwise comparison, finding that strong LLM judges agree with human preferences at rates comparable to inter-annotator agreement. Prometheus [12] trained open-source evaluator models on custom rubrics, showing that rubric-based evaluation can be made transparent and reproducible. Our work extends this line by applying rubric-based LLM judging to a domain-specific, tool-augmented conversation setting and by systematically characterising judge reliability and human alignment.

Agent and tool-use benchmarks. ToolLLM [4] evaluates LLMs on 16,000+ real-world APIs, establishing general-purpose tool-use evaluation. AgentBench [5] provides eight interactive environments for multi-turn agent evaluation. WebShop [13] simulates text-based online shopping with a search-and-purchase loop, using a fixed environment rather than LLM tool calling over a live catalog. MINT [14] evaluates multi-turn tool use across domains generally, rather than e-commerce specifically. GAIA [15] tests general AI assistants on tasks requiring reasoning, web browsing, and tool use. Concurrently, the BitGN Agent Challenge [16] introduces an operational e-commerce benchmark (fraud prevention, dispute automation, payment routing) with deterministic scoring; our work is complementary, focusing on the *conversational* product-discovery and recommendation loop.

Hallucination and state verification in conversational commerce. In prior work we studied a related but narrower problem [1]: an AI commerce agent claiming a transactional event occurred (an item added to a basket, an order completed) when the underlying execution trace shows otherwise, which we call a journey hallucination, and verify such claims against trace invariants rather than the model's own account. That method assumes access to ground-truth execution logs at deployment time; the grounding rubric in this paper instead scores this behaviour from an LLM judge on simulated conversations, without such logs. The two are complementary: this benchmark measures how often models make ungrounded claims across models and scenarios; the earlier method offers a mechanism to catch such claims once a system is live.

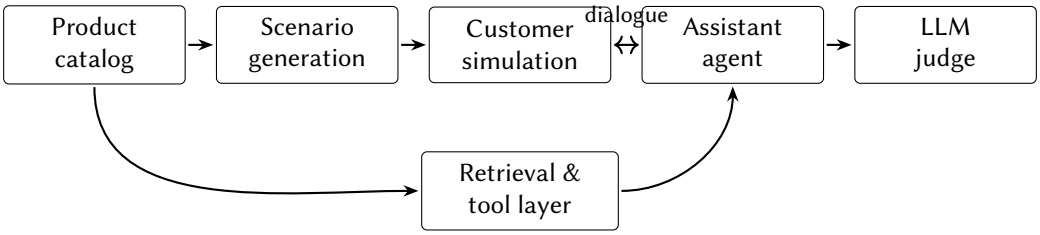

Figure 1: Benchmark architecture. The retrieval & tool layer indexes the catalog once and serves live search and cart results to the assistant agent at inference time, separately from the catalog's role in grounding scenario generation.

3. Benchmark Design

The benchmark comprises five components arranged in a pipeline, shown in Figure 1: a structured product catalog, a scenario generation module, an LLM-driven customer simulation agent, a tool-using assistant agent, and an LLM judge. Sections 3.1–3.4 describe the design of the first four; the judge's design—rubrics, panel composition, human calibration—is substantial enough to warrant its own treatment and is detailed separately in Section 4. The catalog feeds the pipeline twice: once into scenario generation, and once, at inference time, into a retrieval-and-tool layer that indexes the catalog, retrieves and re-ranks candidates for a live query, and serves the result to the assistant agent through the `catalog_search` and `add_to_cart` tools (Section 3.4); Figure 1 shows both paths.

3.1. Product Catalog

We use the Amazon Reviews 2023 dataset [8], a publicly available collection of product metadata and reviews spanning 33 product categories. For the initial evaluation we select the *jeans* subcategory from the Clothing, Shoes and Jewelry domain, yielding 3,229 products after filtering for records with non-null title, price, and at least one product attribute.

Each product record is mapped to a structured schema containing: title, brand, price (USD), gender target, available sizes, colors, materials, product features, a textual description, and a product URL. Each product is embedded once, offline, via a dedicated retrieval-text representation combining its title, attributes, and description, using the same embedding model (Qwen3-Embedding [17] by default) that later encodes queries at inference time; the resulting vectors form a static per-category index (Section 3.4) that does not change over the course of a benchmark run.

3.2. Scenario Generation

Scenarios define the shopping intent that drives each benchmark conversation. We ground scenario generation in real consumer search behaviour using two complementary sources: Semrush keyword data for product-specific intents and Pinterest queries for occasion-based intents. After deduplication via cosine similarity on Qwen3-Embedding vectors, this yields a pool of 714 unique queries for the jeans category.

Each query is processed through a three-stage LLM pipeline (using Kimi K2.5): (1) persona generation, (2) initial question generation, and (3) scenario synthesis combining persona, question, and expected shopping behaviour.

From the 714-query pool, we select a subset of 20 scenarios via a *model-breaking* filter: a scenario is retained if at least one model from an initial screening run scores under 4 out of 5 on the five-rubric total of Section 4.1 (i.e. fails at least one rubric). The screening run uses a small set of tool-calling models on the full pool; we do not name those models here and do not claim that this particular screening set is optimal. The resulting subset is designed to maximise discriminative power by targeting cases where model capability genuinely differs; it is not a representative sample of average shopping interactions (see Section 7), and it is a different protocol from the unfiltered GPT-4.1-generated scenarios used in the

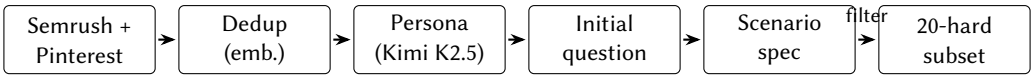

Figure 2: Scenario generation pipeline.

cross-category study (Section 6.6). Applying the filter to a new product domain requires only a screening run of 2–3 models on an initial scenario pool; the failure threshold (score $< 4/5$) is domain-agnostic. Figure 2 summarises the pipeline.

3.3. Customer Simulation Agent

Each benchmark conversation is driven by an LLM-based customer simulation agent that follows the intent and flow of a *golden dialogue*—an LLM-written shopping script produced by the same three-stage Kimi K2.5 pipeline as Section 3.2 (persona, opening question, and expected behaviour), not a real customer transcript and not human-reviewed—while adapting naturally to the actual assistant responses. Key design decisions for determinism and comparability:

- **Golden first-message injection:** The first customer message is taken verbatim from that golden dialogue, ensuring all models receive an identical opening.
- **Temperature 0:** The customer agent operates at temperature 0 to minimise stochastic variation across runs.
- **Loop-detection guard:** Repetition of the opening message triggers a re-prompt nudge; persistent repetition ends the conversation gracefully.
- **Termination:** A `#END_OF_DIALOGUE` marker triggers termination. A default limit of 10 user turns is enforced.

The customer agent uses a fixed base model (GPT-4.1) to ensure consistent behaviour across all evaluated assistant models. Recorded conversations average 5–6 turn pairs. We omit the historical maximum because some stored `num_turns` values predate the current 10-user-turn serialization and are not directly comparable to the enforced user-turn limit.

3.4. Assistant Agent and Tool Interface

The assistant agent under evaluation is a pure-LLM agent with access to two tools via the OpenAI function-calling interface:

- `catalog_search`**(query, top_k):** encodes the query using Qwen3-Embedding [17], retrieves top-k candidates from the txtai [18] vector index described in Section 3.1, then passes them through an LLM-based re-ranker that reorders the results by intent fit—product type, gender, colour/style, and budget cues—keeping only SKUs present in the retrieved set; a lexical token-overlap heuristic serves as a fallback when the re-ranker is unavailable. Default $k = 10$.
- `add_to_cart`**(sku, size, quantity):** simulates a cart action. The assistant must confirm SKU, size, and quantity with the customer before calling it.

A fixed system prompt, identical across all evaluated models, operationalises the rubrics in Section 4.1 as explicit behavioural rules: the assistant must call `catalog_search` before recommending anything and must never state a product detail not present in the tool's response (grounding); it must call `add_to_cart` and receive a successful response before confirming an addition, and is instructed never to write phrases such as "added to cart" otherwise (add-to-cart correctness); it may ask at most one clarifying question per turn (clarification); and it must track customer preferences (size, colour, fit, budget) already stated earlier in the conversation rather than re-asking. On a no-match search, the assistant is instructed to silently reformulate the query and search once more before replying, and to state the limitation at most once rather than repeating it across turns.

Table 1
Five evaluation rubrics with binary pass/fail verdicts.

Rubric	Description	0 (Fail)	1 (Pass)
Intent Understanding & User Guidance	Measures whether the assistant correctly interprets the customer's shopping intent and guides the conversation toward fulfilling it.	Misunderstands intent; off-topic responses.	Accurately tracks evolving intent across turns.
Asking Clarification When Needed	Assesses whether the assistant asks relevant clarifying questions when the customer's request is ambiguous.	Never clarifies; assumes incorrectly.	Asks targeted questions that narrow the search effectively.
Recommendation Usefulness	Evaluates whether the products recommended match the customer's stated and inferred preferences.	Recommendations are irrelevant or absent.	Recommendations closely match preferences.
Add-to-Cart Correctness	Checks whether the assistant correctly executes cart actions.	Fabricates cart confirmations without tool calls.	Confirms details, calls tool, reports outcome faithfully.
Grounding, Coherence & Non-Hallucination	Measures whether the assistant's claims are faithful to tool-returned data.	Invents product details not in tool results.	All claims traceable to tool-returned data.

Each assistant turn receives the full conversation history—including prior tool calls and tool results—followed by the new customer message; the assistant model runs at temperature 0, the same determinism choice as the customer simulator (Section 3.3). Within a turn, the assistant may call tools for up to ten rounds before a final reply is required. LLM completions are retried up to three times with capped exponential backoff on transient provider errors (rate limits, timeouts), consistent with the provider-reliability behaviour discussed in Section 5.

Each evaluated model is instantiated as the assistant's backbone LLM while the system prompt, tool schemas, and retrieval index remain identical across all models. The evaluated model also performs the LLM-based re-ranking step, so re-ranking behaviour is part of the model under evaluation rather than a fixed external component.

4. Evaluation Framework

4.1. Rubric Design and Rationale

The evaluation rubrics were developed iteratively over multiple design cycles, not chosen upfront. The rubric set originated from an earlier internal evaluation exercise and was progressively refined—weak or redundant rubrics were dropped, overlapping ones were merged—arriving at four binary criteria covering intent, clarification, recommendation, and cart correctness. A subsequent review of assistant failures found that models could invent products, prices, or claims without ever having called the search tool, a failure none of the four existing rubrics caught; this motivated adding grounding as a fifth, separately scored rubric. Table 1 summarises the final set of five rubrics.

Binary pass/fail scoring was chosen over Likert scales for higher inter-run judge consistency and a directly interpretable total (0–5).

4.2. Judge Pipeline

Conversations are scored by an LLM judge that receives the *full* conversation history including tool-call and tool-result messages—enabling it to verify grounding claims against actual tool outputs. Each conversation is scored in a single judge call that returns all five rubrics together, rather than one call per

rubric. Before scoring, the transcript is reformatted so that tool calls and their results are inlined into the assistant's turn as explicit annotations—which tool was called, with what arguments, and what it returned—and any hidden reasoning traces the model may have emitted are stripped, so the judge sees exactly the tool interactions and final response a customer would experience. Judge calls are retried with exponential backoff on transient failures (timeouts, rate limits, server errors), consistent with the provider-reliability behaviour discussed in Section 5. The primary headline judge is GPT-4.1; we complement this with a systematic judge-reliability programme described below. Benchmark code and judge prompt templates are at https://github.com/rezolved/conversational-commerce-benchmark-framework. The reported scores were produced through the internal evaluation endpoint. The public runner provides an OpenAI-compatible reference implementation of the same five-rubric interface; it enables new evaluations but is not a bit-for-bit reproduction of that internal judge service.

4.3. Judge Reliability and Human Calibration

A single LLM judge produces leaderboard scores that are contingent on that judge's leniency and interpretation. We address this with a three-part reliability programme. This subsection is methodological groundwork rather than a result in itself; readers primarily interested in model rankings can skim it on a first pass and return to it when interpreting the judge-dependent findings in Section 6.

Seven-judge cross-family panel. We score 14 models (the 12 public models plus the two secondary ablation models of §6.5) with seven independent base judges chosen to spread coverage across families (OpenAI GPT-4.1 and GPT-5.2, open GPT-oss, and Anthropic Claude), not because this particular set is claimed to be optimal: GPT-4.1, GPT-5.2-high, GPT-5.2-low, GPT-oss-120b-medium, GPT-oss-120b-high, Claude-Opus-4-8, and Claude-Sonnet-4-6. Judges differ sharply in strictness (overall pass rates: 88.3% for GPT-4.1 vs. 53.2% for GPT-5.2-low) and change who ranks first: GPT-4.1 gives the top rank to Qwen3.5-122B-A10B, GPT-5.2 to Gemma4-31B, and Claude to Kimi-K2.6. A cross-family panel (OpenAI + Claude) improves inter-judge alignment vs. an OpenAI-only panel: pooled disagreement (the share of rubric cells without a unanimous base-judge verdict) falls from 26.6% to 24.5% while Cohen's κ (chance-corrected agreement) rises from 0.391 to 0.402, with the Claude pair ($\kappa = 0.693$) being the tightest cross-model pairing. Disagreement concentrates on two rubrics: Grounding (13 of the 20 largest model-by-rubric swings) and Recommendation (7 of 20); Intent and Add-to-Cart remain stable across judges. *What survives every judge:* coarse score bands (highest-scoring models near 4.6–4.8, a middle band around 4.1–4.6, and a lower band below 4) are judge-robust; fine-grained ranking within the top cluster is not defensible from a single judge.

Super-judge arbitration. We select a three-model super-judge panel (Gemini, DeepSeek, GLM) from eight candidates. *Swing* is the rate at which 0/1 verdicts differ across judges or repeated scorings of the same cell; *centrality* is how often a candidate's verdicts coincide with the majority of the other candidates. The three chosen models had low swing and high centrality relative to that eight-candidate pool. The panel arbitrates only the 53.4% of rubric cells where the seven base judges disagree, keeping the base-judge majority on unanimous cells. This *reliable stack*—the seven base judges plus the three-model super-judge on disputed cells—reduces pooled swing from 24.5% to **9.2%** (a 62% relative reduction), with 74.1% *unanimous resolution* (all three super-judges agreeing) on disputed cells.

Noise decomposition. We decompose the benchmark's noise floor into two components: *system-level swing* (each run regenerates the dialogue, so dialogue variance is included) and *judge-only swing* (the same frozen dialogue re-judged 10× at temperature 0). System-level swing is 22.0%; judge-only swing is 8.7%, meaning approximately 60% of apparent instability is dialogue-resampling variance, not judge wobble. GPT-4.1 is roughly 3× more self-reproducible than GPT-5.2 on frozen dialogues (8.0% vs. 22.4% swing), making it the more stable headline judge even though it is the most lenient.

Table 2
Alignment of judge configurations with a 103-cell human gold set (single reviewer; 11 pass / 92 fail; directional only; multi-reviewer adjudication ongoing). An always-fail dummy would score $92/103 = 89.3\%$.

Judge / stack	Accuracy	κ	Pass rate
Reliable stack	**88.3%**	0.273	6.8%
GPT-5.2-low	83.5%	**0.324**	17.5%
GPT-5.2-high	78.6%	0.244	22.3%
GPT-4.1 (headline)	17.5%	−0.126	79.6%

Human gold calibration (preliminary). We hand-labelled 103 rubric cells, deliberately sampled from hard cases where base judges disagree or scores are borderline. A single reviewer scored them blind to AI judgments. Human pass rate on this hard slice is 10.7% (11 pass / 92 fail). Table 2 shows alignment per judge and stack against this gold set.

GPT-4.1 is far too lenient on this hard slice ($\kappa = -0.13$, 79.6% pass rate vs. 10.7% human rate). The reliable stack achieves the highest accuracy (88.3%) and the pass rate closest to human gold (6.8%). Because the gold set is heavily imbalanced (11 pass / 92 fail), an always-fail dummy would score $92/103 = 89.3\%$, slightly above the stack; Cohen's κ (0.273) measures agreement beyond chance and is therefore the more informative figure. GPT-5.2-low is the best single-base judge by κ (0.324). We report GPT-4.1 as the headline leaderboard judge for reproducibility and self-consistency, and include the reliable-stack ranking as a robustness check. *Caveat:* this is a single-reviewer, hard-case slice; multi-reviewer adjudication is in progress. The κ ordering should be treated as directional, not definitive.

5. Experimental Setup

Models. We evaluate 12 publicly available models spanning four model families and five inference providers (Table 3). The twelve models span more than an order of magnitude in scale, from 20B to roughly 1T total parameters, and include both dense and mixture-of-experts (MoE) architectures; three families (Qwen3.5, Gemma-4, gpt-oss) contribute more than one size point, allowing within-family scaling comparisons alongside cross-family ones. One model, Gemma-4-26B-A4B, is evaluated twice under identical weights but different providers (SiliconFlow and self-hosted) specifically to isolate serving infrastructure as a confound (Section 7). All twelve models support native function calling, the minimum requirement for participating in this benchmark at all (Section 3.4). The roster reflects models already available to our evaluation infrastructure at the time of this study, filtered to those meeting the tool-calling requirement, rather than an exhaustive or formally representative sample of all publicly available models; we discuss this as a limitation in Section 7. We additionally run a secondary tool-calling ablation on two further models (Appendix A), reported separately from this primary leaderboard since it targets a narrower question—the causal effect of tool access—rather than a head-to-head model comparison.

Scenarios and runs. Each model is evaluated on the same 20 hard jeans scenarios, with 5 independent runs per model (1,200 conversations total, 6,000 rubric cells).

Provider reliability. Self-hosted and SiliconFlow providers completed all assigned scenarios with 0% initial timeouts. Nebius reported 5.5% timeouts, all recovered. DeepInfra recorded 155/300 initial timeouts (51.7%) for the Qwen3.5 trio before those models migrated to DashScope. On DashScope, the same trio initially had 117/300 timeouts (39%), all cleared after extended recovery passes. The DeepInfra-hosted gpt-oss-20B also required recovery passes, but the available records do not support assigning the 51.7% rate to that model. Results include only successfully completed conversations.

Table 3
Evaluated models. All support native tool calling. †: same base model, different provider.

Model	Family	Params	Provider
Qwen3.5-122B-A10B	Alibaba	122B (MoE)	DashScope
Qwen3.5-35B-A3B	Alibaba	35B (MoE)	DashScope
Qwen3.5-27B	Alibaba	27B	DashScope
Kimi-K2.5	Moonshot AI	1T (MoE)	Nebius
Gemma4-31B	Google	31B	Self-hosted
Kimi-K2.6	Moonshot AI	1T (MoE)	Self-hosted
Gemma-4-26B-A4B	Google	26B (MoE)	SiliconFlow
gpt-oss-120B	OpenAI (open)	120B	Nebius
Gemma-4-26B-A4B†	Google	26B (MoE)	Self-hosted
gpt-oss-20B	OpenAI (open)	20B	DeepInfra
Qwen3-32B	Alibaba	32B	Nebius
Qwen3-30B-A3B	Alibaba	30B (MoE)	Nebius

Table 4
Model leaderboard (GPT-4.1 judge): 12 public models. Mean total scores (0–5), run standard deviation, and bootstrap 95% CI gap and sign-flip p-value vs. the fixed reference model, Qwen3.5-122B-A10B.

Rk	Model	Mean	σ	Δ (95% CI)	p
1	Qwen3.5-122B-A10B	4.77	0.07	—	—
2	Qwen3.5-35B-A3B	4.77	0.11	$+0.00\ [-0.10, +0.11]$	1.00
3	Qwen3.5-27B	4.76	0.02	$-0.01\ [-0.11, +0.09]$	1.00
4	Kimi-K2.5	4.72	0.13	$-0.05\ [-0.16, +0.04]$	0.44
5	Gemma4-31B	4.71	0.06	$-0.06\ [-0.22, +0.08]$	0.51
6	Kimi-K2.6	4.70	0.04	$-0.07\ [-0.19, +0.04]$	0.30
7	Gemma-4-26B-A4B	4.65	0.09	$-0.12\ [-0.27, +0.03]$	0.16
8	gpt-oss-120B	4.64	0.12	$-0.13\ [-0.28, -0.02]$	0.07
9	Gemma-4-26B-A4B†	4.59	0.17	$-0.18\ [-0.34, -0.04]$	0.03
10	gpt-oss-20B	4.56	0.14	$-0.21\ [-0.36, -0.08]$	<0.01
11	Qwen3-32B	4.15	0.13	$-0.62\ [-0.90, -0.44]$	<0.01
12	Qwen3-30B-A3B	3.52	0.23	$-1.25\ [-1.52, -1.00]$	<0.01

Timeout rates are uncorrelated with model score on completed conversations (provider stability and conversation quality are orthogonal dimensions of this evaluation).

6. Results

6.1. Model Leaderboard

Table 4 reports GPT-4.1-judged mean scores with bootstrap confidence intervals (BCa, 2,000 iterations, well past convergence for a 95% CI; sign-flip p-value against the fixed reference model, Qwen3.5-122B-A10B; methodology from [19]).

The top seven rows all overlap the reference model at $\alpha = 0.05$. Rank 8 (gpt-oss-120B) is a borderline case: the BCa interval on the gap excludes zero ($[-0.28, -0.02]$) while the sign-flip p-value is 0.07; we do not treat this as evidence of a ranked difference, nor as proof of equality. From rank 9 all gaps are statistically significant ($p < 0.05$). We therefore talk in overlapping bands rather than a hard score cut: ranks 1–7 sit in a high overlapping band (means 4.65–4.77); ranks 8–11 form a lower overlapping band (4.15–4.64) whose upper end (gpt-oss-120B, 4.64) sits next to rank 7 (4.65). Rank 12 (Qwen3-30B-A3B, 3.52) stands notably apart. Bootstrap CIs confirm that overlapping bands, not point estimates, are the defensible unit of comparison. Coarse band membership is judge-robust for the highest-scoring models.

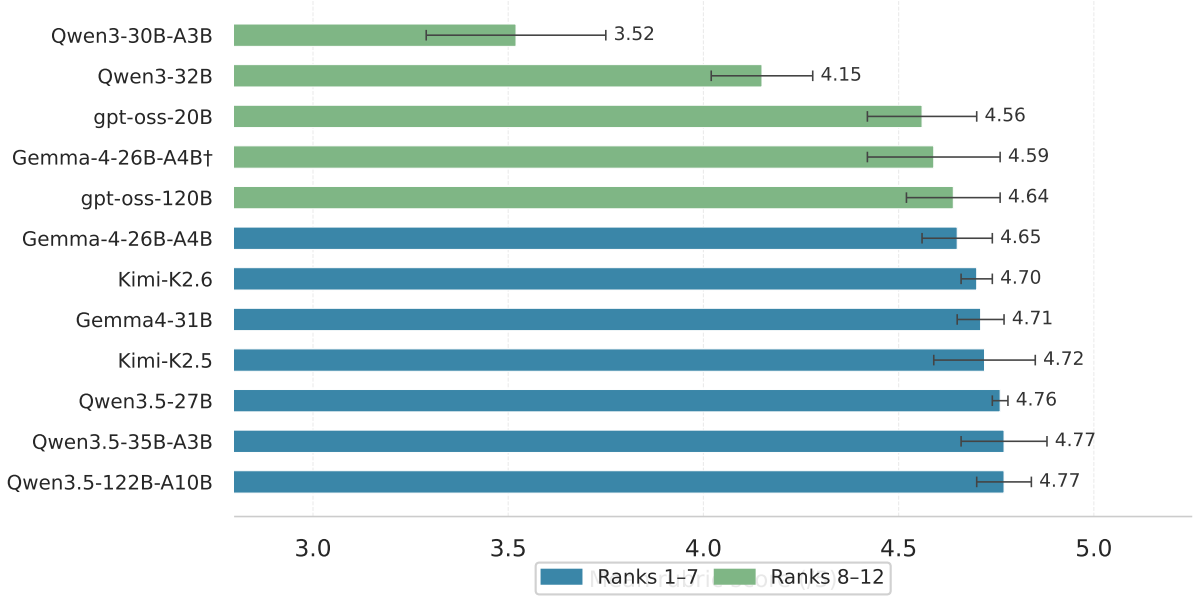

Figure 3: Mean total score with run-to-run standard deviation (GPT-4.1 judge; 12 public models; † = same base model different provider); the same means and standard deviations are reported in Table 4. Band colours mark ranks 1–7 vs. 8–12; error bars for the top seven models overlap substantially.

Reliable-stack ranking. Under the reliable stack (Section 4.3), the leader changes to Kimi-K2.6 (4.36/5), with Gemma4-31B second (4.06) and Kimi-K2.5 third (3.85). The Qwen3.5 trio drops from ranks 1–3 to ranks 4–7. Absolute scores decrease under the reliable stack, with model-dependent changes that alter the fine-grained ordering. This finding illustrates that leaderboard ordering is judge-dependent, while coarse band membership is more stable.

6.2. Variance Analysis

Run-to-run variance. The three GPT-4.1 leaders, Qwen3.5-122B-A10B (4.77 ± 0.07), Qwen3.5-35B-A3B (4.77 ± 0.11), and Qwen3.5-27B (4.76 ± 0.02), have inter-model gaps (0.00–0.01) smaller than the run-to-run standard deviation of the most variable model in this group (0.11). The data therefore do not support a reliable fine-grained ordering; a single-run evaluation would yield a spuriously definitive ranking. Figure 3 visualises this same overlapping-band structure across all twelve models—the same means and standard deviations as Table 4, plotted as horizontal bars with band shading—making the overlap among the highest-scoring models visually immediate; Table 4 remains the source for the formal bootstrap confidence intervals and significance tests behind that overlap.

Noise decomposition. System-level swing (22.0%) encompasses both dialogue and judge noise; judge-only swing on frozen dialogues is 8.7%, meaning ∼60% of apparent instability is dialogue-resampling variance. Clarification exhibits the highest system-level swing (35.3%), confirming it is the noisiest rubric; Intent has the lowest (8.3%). Five-run averaging substantially absorbs dialogue-side variance; the residual 8.7% judge-only floor is the minimum noise achievable with the current judge.

6.3. Per-Rubric Analysis

Building on the noise decomposition above, where Clarification showed the highest system-level swing (35.3%), Figure 4 shows why the aggregate score alone is insufficient. Across ranks 1–7, Clarification is the least saturated rubric, ranging from 0.77 to 0.84. The other rubrics are near-saturated for ranks 1–6, while the rank-7 Gemma-4-26B-A4B Grounding score is 0.88. Grounding provides the widest spread across the full leaderboard and contributes strongly to separation below the leading band.

Figure 4: Per-rubric mean pass rates (GPT-4.1 judge). Clarification is least saturated among ranks 1–7; Grounding exhibits the widest overall spread.

Table 5
Behavioural auto-metrics (5 runs × 20 scenarios per model). Eleven roster entries have available metrics; the second provider instance of Gemma-4-26B-A4B lacks a complete row.

Model	Turns	MaxHit	Tool%	Cart%
Qwen3.5-122B-A10B	5.90	3%	100	99
Qwen3.5-35B-A3B	6.41	11%	100	96
Qwen3.5-27B	5.90	5%	100	98
Kimi-K2.5	5.85	7%	100	98
Gemma4-31B	6.33	10%	100	92
Kimi-K2.6	6.21	8%	100	93
Gemma-4-26B-A4B	6.80	21%	100	—
gpt-oss-120B	5.16	3%	100	98
gpt-oss-20B	5.39	4%	100	100
Qwen3-32B	6.15	8%	100	97
Qwen3-30B-A3B	5.05	3%	100	98

Grounding pass rates span 0.51 (Qwen3-30B-A3B) to 1.00 (Qwen3.5-35B-A3B / Qwen3.5-27B), the widest spread across the five rubrics.

Example (grounding failure). In scenario `gisela_ribeiro_3` with Qwen3-30B-A3B, the customer asks for "vintage-wash denim shirts"; the assistant describes shirts without calling `catalog_search`, then searches and receives only jeans—failing the Grounding rubric because prior claims were not traceable to tool outputs.

Saturation note. Under the reliable stack, rubric saturation largely disappears: Recommendation opens to a 37-point spread across models and Grounding to 51%, enabling finer separation. This suggests that the saturation observed under GPT-4.1 is partly a leniency artefact rather than a ceiling on model capability.

6.4. Automatic Behavioural Metrics

Beyond judge-assessed rubric scores, Table 5 reports automatic behavioural metrics computed directly from conversation traces, independent of any LLM judge. In the ten-model subset with accessible con-

Table 6
Cross-category pooled mean total scores over successful GPT-4.1 judgments (5 runs, 0–5 scale; unequal per-model $n = 87$–100). Rank within each category in parentheses.

Model	Shoes	Electronics	Beauty
Qwen3-32B	1.78 (1)	2.80 (1)	2.41 (1)
gpt-oss-120B	1.62 (2)	1.84 (3)	1.62 (5)
gpt-oss-20B	1.48 (3)	1.72 (4)	1.80 (2)
Gemma4-31B	1.43 (4)	2.12 (2)	1.75 (3)
Qwen3-30B-A3B	1.38 (5)	1.67 (5)	1.69 (4)

versation traces, every recorded conversation invokes `catalog_search` (Tool% = 100); the remaining table row comes from the internal trace summary. These traces are not part of the public release, so Table 5 cannot be independently reconstructed from the released artifacts. Cart completion rates are 92–100% where measured. Gemma-4-26B-A4B is the wordiest reported model (6.80 avg. turns, 21% max-turn hits) and has the highest Clarification pass rate (0.84). In the ten-model trace subset, turn count is moderately correlated with total score ($r = 0.52$, $n = 10$), but the association is not statistically significant; operational metrics therefore capture information not reducible to judge-assessed quality.

6.5. Tool-Calling Ablation (Secondary Finding)

As an additional finding separate from the primary leaderboard above, we probe the causal effect of tool-calling access using two additional models. Disabling tool calling reduces total scores by 2.2–2.8 rubric passes while grounding collapses by 0.74–0.77, supporting a causal interpretation: native tool calling is a prerequisite for acceptable performance on this benchmark. Full scores are in Appendix A.

6.6. Cross-Category Generalisation

To test whether findings from the jeans worked example hold under a different scenario protocol, we replicate the evaluation on shoes (Athletic), electronics (Headphones & Earbuds), and beauty (Skincare) from Amazon Reviews 2023. Each category uses 20 GPT-4.1-generated scenarios with the same tool interface, customer simulation, and 5-rubric judge—without the model-breaking filter of Section 3.2. Table 6 reports results for a five-model subset spanning the full score range of the primary evaluation. Of 1,500 attempted model–run–scenario combinations, 1,433 received valid GPT-4.1 judgments; the pooled means below use the available successful judgments, with per-model denominators ranging from 87 to 100.

Absolute scores (1.4–2.8) are lower than on jeans (3.5–4.8). This is not evidence that jeans is a harder catalog. The two studies used different scenario-selection protocols: jeans used the model-breaking filter of Section 3.2, while shoes, electronics, and beauty used unfiltered GPT-4.1-generated scenarios. The scores are therefore not directly comparable, and the jeans numbers are a worked example of the shopping-loop protocol rather than a proof that the framework generalises across e-commerce. The Qwen3-32B order change below is a lesson about how we pick scenarios, not a claim that raw scores transfer.

Qwen3-32B ranking flip. A notable finding is that Qwen3-32B ranks 11th of 12 on jeans under GPT-4.1 but *first* on all three cross-category domains. We attribute this to the scenario composition: our model-breaking filter selects predominantly Clarification-failure scenarios, and Qwen3-32B scores only 0.68 on Clarification vs. 0.77–0.84 for the top cluster, making it specifically vulnerable to these curated scenarios. Standard GPT-4.1-generated scenarios on other categories do not target this weakness, allowing Qwen3-32B to recover and demonstrate competitive recommendation and grounding capabilities. This finding illustrates that scenario curation choices can materially affect relative rankings—a methodological insight that motivates the model-breaking filter design itself.

Cross-category rank correlation. Spearman ρ between the three unfiltered categories ranges from 0.40 to 0.70 ($n = 5$ models). We treat this as directional only: $n = 5$ is too small for statistical significance, the jeans protocol is different, and we do not claim that scores transfer. As a missing-data sensitivity check, restricting each category to model–run–scenario cases completed by all five models (94 cases per model for shoes, 62 for electronics, and 83 for beauty) yields $\rho = 0.70$ for all three category pairs. The qualitative direction is stable, but the exact pooled correlations are sensitive to missingness.

7. Discussion and Limitations

LLM judges and human calibration. Our judge-reliability programme (Section 4.3) shows that GPT-4.1, while the most self-reproducible single judge (8% frozen-dialogue swing), is far too lenient on hard cases ($\kappa = -0.13$ against human gold). The reliable stack reduces swing to 9.2% and reaches 88.3% accuracy against human labels on this slice—below an always-fail dummy at 89.3%, so $\kappa = 0.273$ is the more informative agreement figure. We recommend the reliable stack for future iterations of this benchmark. This has a direct implication for anyone adapting this framework to a new assistant or catalog: a single off-the-shelf judge model should not be trusted at face value, and at minimum a second, independently-sourced judge should be run alongside it before treating a leaderboard position as fact.

Serving infrastructure as a confound. Self-hosted and third-party-API deployments of the same model are not guaranteed to be identical in behaviour: quantisation, prompt templating, and tool-schema wiring can differ by provider even when the weights are unchanged. Our own leaderboard contains direct evidence of this: Gemma-4-26B-A4B, run via SiliconFlow and via self-hosting with the same weights, scores 4.65 and 4.59 respectively (Table 4); the 0.06/5 gap attributable purely to serving location. This is a caution against reading any self-hosted-vs.-third-party score gap in this benchmark as evidence about the underlying model alone. Teams comparing models across providers should therefore hold serving configuration constant wherever possible, or explicitly report it alongside model identity, since the two are otherwise easy to conflate.

Limitations. Several limitations should temper how broadly these findings are generalised. *First*, the human gold set underlying our judge-reliability calibration is preliminary: a single reviewer, 103 hard rubric cells (11 pass / 92 fail); an always-fail dummy scores 89.3%, above the reliable stack's 88.3% accuracy, so κ rather than raw accuracy is the right summary. Multi-reviewer adjudication is in progress and is the primary open gap in the evaluation. *Second*, the model-breaking scenario filter deliberately selects non-representative scenarios to maximise discriminative power, so absolute jeans scores do not represent average shopping-assistant quality in deployment. The cross-category experiments used a different, unfiltered protocol and are not a like-for-like hardness comparison; jeans is the worked example, and those three categories do not prove a general e-commerce framework. *Third*, in the cross-category experiments GPT-4.1 serves as scenario generator, customer simulator, and judge simultaneously, a circularity that may introduce self-preference bias not yet disentangled; future work should assign these roles to different model families. *Fourth*, provider timeout rates ranged from 0% (self-hosted) to 51.7% for the Qwen3.5 trio on DeepInfra before migration and recovery; we report only completed conversations, which could introduce subtle bias if timeouts correlate with scenario difficulty, though timeout rates show no significant correlation with model scores in our data, and models with the highest pre-recovery timeout rates (the Qwen3.5 trio) rank at the top of the leaderboard after recovery. *Fifth*, the evaluated model roster reflects what was available to our evaluation infrastructure at the time of the study, filtered only by the tool-calling requirement, rather than a systematic or exhaustive sample of publicly available models; results should not be read as ranking every model a practitioner might consider. *Finally*, the five rubrics were refined specifically for tool-using e-commerce assistants and have not been independently validated on other assistant types, which may require rubric adaptation.

Table 7
Practical takeaways for teams building or evaluating conversational commerce agents.

#	Topic	Recommendation	Key evidence
1	Judge design	Use a cross-family multi-judge reliable stack with super-judge arbitration	GPT-4.1 alone: $\kappa = -0.13$ vs. human gold; reliable stack: 88.3% accuracy vs. 89.3% always-fail dummy, $\kappa = 0.273$, swing 24.5%→9.2%
2	Model comparisons	Never trust a single run; use multi-run bootstrap confidence intervals	Highest-scoring models form overlapping bootstrap bands, not a claimed ranking (Section 6.1)
3	Rubric focus	Track Clarification and Grounding alongside the aggregate score	Clarification is least saturated in ranks 1–7; grounding has the widest overall spread
4	Cross-category transfer	Treat transfer evidence as preliminary; jeans is a worked example under a different protocol from the other categories	$\rho = 0.40$–0.70 ($n = 5$, directional); top-tier models not yet run cross-category (Section 6.6)

Practical takeaways. Taken together, the discussion above has direct operational consequences: judge choice and aggregation method are not neutral implementation details but change which model looks best; single-run comparisons among top-tier models are not reliable enough to act on; grounding failures are common enough that any deployment should budget for verifying assistant claims against actual system state, not just conversation fluency. Table 7 summarises the preceding discussion as a compact decision guide for practitioners building or evaluating conversational commerce agents.

8. Conclusion and Future Work

We presented a reproducible framework for benchmarking LLMs across the full conversational shopping loop and demonstrated it in a first-in-class 12-model study. The primary jeans evaluation is a worked example rather than a universal e-commerce ranking; its empirical value is to expose overlapping performance bands, rubric-specific differences, and substantial judge dependence. Applying the framework across five independent runs, we found that (i) the highest-scoring models form overlapping bootstrap confidence bands under a GPT-4.1 judge, so fine-grained rank among them is not a claimed result, (ii) a multi-judge reliable stack with super-judge arbitration reduces evaluation swing from 24.5% to 9.2%; accuracy against a preliminary, imbalanced human gold set is 88.3%, below an always-fail dummy at 89.3% ($\kappa = 0.273$), and (iii) a smaller cross-category study under a different scenario protocol yields only directional rank-order evidence ($\rho = 0.40$–0.70, $n = 5$; Section 6.6). A key methodological finding is that GPT-4.1 alone is too lenient on hard cases ($\kappa = -0.13$ vs. human labels); we recommend a cross-family judge panel with super-judge arbitration for future iterations.

For practitioners, three findings translate directly into deployment guidance. (1) **Track Clarification alongside the aggregate score**: it is the least saturated rubric among ranks 1–7 under GPT-4.1. (2) **Track Grounding fidelity**: it has the widest spread across the leaderboard and helps separate score bands. (3) **Never trust a single run**: run-to-run variance is large enough that single-run comparisons among top models are unreliable, so bootstrap confidence intervals over multiple runs are necessary before declaring a winner (Table 7 summarises this and the judge-design guidance above).

Future directions include: (i) multi-reviewer human validation to robustly calibrate the judge stack, (ii) expanding the model lineup to frontier closed-source families, (iii) running the top-tier models across all product categories, and (iv) extending the benchmark to multilingual shopping conversations.

Benchmark code, the jeans catalog, the 20 frozen scenarios, and a minimal runner are at https://github.com/rezolved/conversational-commerce-benchmark-framework. The catalog is derived from the publicly available Amazon Reviews 2023 dataset [8].

Declaration on Generative AI

During the preparation of this work, the authors used AI assistants to help write benchmark, analysis, and figure-generation code, and to help draft and edit portions of the manuscript text. The experimental design, findings, and conclusions are the authors' own.

References

[1] D. Carvalho, S. Krivic, T. Tang, S. Ahmad, User-state verification in conversational commerce: Detecting journey hallucinations via trace invariants, in: Proceedings of the 34th ACM Conference on User Modeling, Adaptation and Personalization (UMAP '26), 2026. doi:10.1145/3774935.3812704.

[2] R. Li, S. E. Kahou, H. Schulz, V. Michalski, L. Charlin, C. Pal, Towards deep conversational recommendations, in: Advances in Neural Information Processing Systems, Curran Associates, Red Hook, NY, 2018, pp. 9748–9758.

[3] S. A. Hayati, D. Kang, Q. Zhu, W. Shi, Z. Yu, INSPIRED: Toward sociable recommendation dialog systems, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 8142–8152.

[4] Y. Qin, S. Liang, Y. Ye, K. Zhu, L. Yan, Y. Lu, et al., ToolLLM: Facilitating large language models to master 16000+ real-world APIs, in: Proceedings of the Twelfth International Conference on Learning Representations (ICLR), OpenReview.net, Vienna, Austria, 2024.

[5] X. Liu, H. Yu, H. Zhang, Y. Xu, X. Lei, H. Lai, et al., AgentBench: Evaluating LLMs as agents, in: Proceedings of the Twelfth International Conference on Learning Representations (ICLR), OpenReview.net, Vienna, Austria, 2024, pp. 1–28.

[6] Y. Jin, et al., Shopping MMLU: A massive multi-task online shopping benchmark for large language models, in: Advances in Neural Information Processing Systems (Datasets and Benchmarks Track), Curran Associates, Red Hook, NY, 2024, pp. 18062–18089.

[7] J. Huang, S. Wang, L. Ning, W. Fan, S. Wang, D. Yin, Q. Li, Towards next-generation recommender systems: a benchmark for personalized recommendation assistant with LLMs, in: Proceedings of the Nineteenth ACM International Conference on Web Search and Data Mining (WSDM), Association for Computing Machinery, New York, NY, 2026, pp. 217–226. doi:10.1145/3773966.3777954.

[8] Y. Hou, J. Li, Z. He, A. Yan, X. Chen, X. Fu, J. McAuley, Bridging language and items for retrieval and recommendation: Benchmarking LLMs as semantic encoders, in: Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, 2026, pp. 3251–3265. Amazon Reviews 2023 dataset: https://amazon-reviews-2023.github.io/.

[9] K. Zhou, Y. Zhou, W. X. Zhao, X. Wang, J.-R. Wen, Towards topic-guided conversational recommender system, in: Proceedings of the 28th International Conference on Computational Linguistics (COLING), International Committee on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 4128–4139.

[10] Y. Liu, D. Iter, Y. Xu, S. Wang, R. Xu, C. Zhu, 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), Association for Computational Linguistics, Singapore, 2023, pp. 2511–2522.

[11] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, I. Stoica, Judging LLM-as-a-judge with MT-Bench and chatbot arena, in: Advances in Neural Information Processing Systems (Datasets and Benchmarks Track), Curran Associates, Red Hook, NY, 2023, pp. 1–29.

[12] S. Kim, J. Suk, S. Longpre, B. Y. Lin, J. Shin, S. Welleck, G. Neubig, M. Lee, K. Lee, M. Seo, Prometheus 2: An open source language model specialized in evaluating other language models,

in: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Miami, FL, 2024, pp. 4334–4353.
[13] S. Yao, H. Chen, J. Yang, K. Narasimhan, WebShop: Towards scalable real-world web interaction with grounded language agents, in: Advances in Neural Information Processing Systems, Curran Associates, Red Hook, NY, 2022, pp. 1–15.
[14] X. Wang, et al., MINT: Evaluating LLMs in multi-turn interaction with tools and language feedback, in: Proceedings of the Twelfth International Conference on Learning Representations (ICLR), OpenReview.net, Vienna, Austria, 2024, pp. 1–20.
[15] G. Mialon, C. Fourrier, C. Swift, T. Wolf, Y. LeCun, T. Scialom, GAIA: A benchmark for general AI assistants, in: Proceedings of the Twelfth International Conference on Learning Representations (ICLR), OpenReview.net, Vienna, Austria, 2024, pp. 1–36.
[16] BitGN and COLIBRIX ONE, Agentic e-commerce challenge (ECOM1), 2026. URL: https://bitgn.com/challenge/ecom, accessed 2026-05-20.
[17] Y. Zhang, et al., Qwen3 embedding: Advancing text embedding and reranking through foundation models, arXiv preprint arXiv:2506.05176 abs/2506.05176 (2025).
[18] D. Mezzetti, txtai: All-in-one open-source embeddings database, 2024. URL: https://github.com/neuml/txtai.
[19] B. Efron, R. J. Tibshirani, An Introduction to the Bootstrap, Chapman & Hall, New York, 1993.

A. Tool-Calling Ablation: Full Results

As an additional finding separate from the primary leaderboard (Section 6.5), we probe the causal effect of tool-calling access using two additional models (A, ∼120B; B, ∼20B), each evaluated with tool calling enabled (TC) and disabled (no-TC) on the same served instance. Table 8 reports mean total scores and grounding pass rates (5 runs × 20 scenarios each).

Table 8
Tool-calling ablation on two additional models. TC: tool-calling enabled; no-TC: `catalog_search` disabled. Params: approximate size.

Variant	Params	Mean (/5)	Grounding
Ablation Model A (TC)	∼120B	4.49	0.90
Ablation Model A (no-TC)	∼120B	2.29	0.16
Ablation Model B (TC)	∼20B	4.24	0.83
Ablation Model B (no-TC)	∼20B	1.46	0.06

Disabling tool calling reduces total scores by 2.2–2.8 rubric passes while grounding collapses from 0.83–0.90 to 0.06–0.16, supporting a causal interpretation: native tool calling is a prerequisite for acceptable performance on this benchmark. We expect this effect to hold for any tool-using model in this setting, not just the two instances tested here.