

BenchBrowser: Retrieving Evidence for Evaluating Benchmark Validity

Harshita Diddee¹ Gregory Yauney^{2*} Swabha Swayamdipta³ Daphne Ippolito¹

¹Language Technologies Institute, Carnegie Mellon University ²Amazon

³Thomas Lord Department of Computer Science, University of Southern California

hdiddee@andrew.cmu.edu, gyauney@gmail.com, swabhas@usc.edu, daphnei@cmu.edu

Abstract

Do language model benchmarks actually measure what practitioners intend them to? Benchmark metadata often obscures how benchmarks operationalize a capability: a “poetry” benchmark may omit haikus, while “instruction-following” benchmarks may test for some arbitrary set of skills. This coarseness prevents practitioners from inspecting how capabilities are operationalized at the item level, leading to unclear estimates of model competence. We introduce BENCHBROWSER, an auditing infrastructure that retrieves use-case-relevant items from 20+ benchmarks, letting practitioners inspect the evidence behind benchmark claims. Through a human evaluation with ten NLP practitioners, we show that the retrieved items are relevant and help support two meta-evaluations: surfacing *content validity* threats, where advertised capabilities are unevenly represented across facets, and *convergent validity* threats, where nominally similar benchmarks induce unstable model rankings. By tying validity concerns to concrete examples, BENCHBROWSER provides interpretable evidence to calibrate interpretation of benchmark scores and improve future iterations of benchmarks. The [tool](#) and its [code](#) are publicly available.

1 Introduction

Practitioners often operate under a perilous assumption: that public benchmarks act as reliable proxies for their specific use-cases. Yet benchmarks differ widely in both scope and granularity. For instance, in some benchmarks each example probes a different capability, as with InfoBench (Qin et al., 2024) and AlpacaEval (Dubois et al., 2024). But in others, all examples test the same targeted capability, as in GSM8K (Cobbe et al., 2021) and NutriBench (Hua et al., 2024). This heterogeneity in design can compromise the *validity* of our evaluations, i.e., the degree to which the performance of a model

on a benchmark reflects its true competence in the capability of interest (Bean et al., 2025; Ye et al., 2025). To confidently deploy a model, a practitioner must then answer a difficult question: *Does this benchmark provide comprehensive evidence of a model’s competence for my specific use-case?*

Answering this question is difficult because the gap between a benchmark’s description and its content creates two distinct validity pitfalls. First, evaluations can lack *content validity*: the extent to which the items in a test adequately represent the entire domain of the capability being measured. This can be compromised by *sparse coverage*, where a practitioner probing for a multifaceted capability finds few relevant evaluation items in a broad evaluation suite, or by *scope-metadata mismatch*, where broad benchmark descriptions mislead. For example, BIG-bench’s hindu_knowledge task (Srivastava et al., 2023) tests deity hierarchy but doesn’t cover origins of spiritual texts—one of the many aspects that a practitioner might be interested in. Second, evaluations can often lack *convergent validity*: this occurs when models evaluated on different benchmarks measuring the same capability produce conflicting rankings. A model might excel at “reasoning” on science questions (AI2 ARC, Clark et al. 2018) but could fail to “reason” about consistencies in narrative summarization (FlawedFictions, Ahuja et al. 2025). Confronted with these *divergent operationalizations* of reasoning competence, the practitioner may be left unsure of which signal to trust to choose a competent model for their use-case (Eriksson et al., 2025).

Identifying these failures is slow and expensive because validity is both item-level and stakeholder-dependent. For one, manual audits of item-level capabilities do not scale to tens of thousands of benchmark items (Bean et al., 2025), and keyword search over dataset cards cannot reveal how a capability is operationalized inside individual bench-

*Work done prior to joining Amazon.

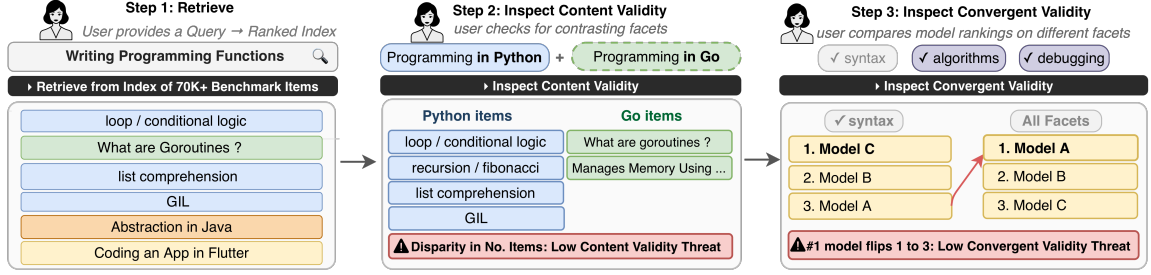


Figure 1: BENCHBROWSER retrieves items relevant to a practitioner’s use-case from an index of 70k+ benchmark items. The interface allows checking (a) **example-level representation** (amount of retrieved evidence, redundancy, metric metadata) and (b) **the stability of model ranks** across subsets of retrieved examples (as well as existing benchmarks) in order to surface threats to content validity and convergent validity, respectively.

mark items (Ma et al., 2025). Additionally, one practitioner’s annotations of “coding ability” may be based on purely syntactical standards, while another’s may acknowledge properties like concurrency; so static benchmark documentation is unlikely to cover such subjective and evolving capability annotations (Zeng et al., 2025). Finally, since validity concerns the interpretation and use of a benchmark score, it is practitioners, and not only benchmark designers, who have to inspect whether these items instantiate their intended construct and whether different operationalizations support consistent conclusions (Wallach et al., 2025). Therefore, this process must be accessible, fast, and friendly to iteration for evolving practitioner definitions of capabilities.

In this work, we introduce BENCHBROWSER, a retrieval-based tool for assessing evaluation validity (Figure 1 and §2). Given a natural-language use case, BENCHBROWSER retrieves semantically related items from tens of thousands of benchmark examples in under 5 seconds, enabling practitioners to gauge how specific items and their design (metric of success, domain coverage) impact the interpretation of benchmark scores for that use-case. We first validate that retrieved items are relevant to practitioners through a human evaluation with $N = 10$ NLP practitioners (§4 and §4.1).

We then use the retrieved evidence for two meta-evaluations: (a) surfacing content validity threats, where small facet changes expose large coverage disparities (§5). For instance, we show how swapping a single facet in a use-case like “programming in Python” to “programming in Go” results in a sharp drop in the number of available examples, demonstrating that competence estimates in “programming” may be disproportionately anchored to this specific language. Then we demonstrate how this evidence helps surface (b) convergent valid-

ity threats, where relevant retrieved items induce model rankings that diverge from human-validated benchmark rankings for the same use-case (§6). For example, when evaluating models on their ability to answer questions about the Hindu religion, the top-ranked models on the multiple-choice BIG-bench hindu_knowledge task can drop to nearly the worst-ranked when instead evaluated on prompt-completion type examples from HellaSwag (Zellers et al., 2019).

Overall, BENCHBROWSER surfaces evidence for practitioners to directly assess validity rather than relying on rigid annotation schema, enabling interactive evaluation of benchmark validity for novel use-cases.

2 BENCHBROWSER

Given a use-case, BENCHBROWSER retrieves the top- k relevant examples from a multi-benchmark database in three stages — generating query rewrites or *anchors* for retrieval, retrieving semantically close items for these anchors and filtering them using an LLM Judge. Unless otherwise stated, LLM-based rewriting and filtering use gpt-5-mini-2025-08-07.

2.1 Stage 1: Anchor Generation

Directly embedding practitioner use-cases for retrieval can often be insufficient: use-cases might be abstract (e.g., “instruction following”), individual benchmark examples may not be annotated with fine-grained capability annotations that the practitioner mentions in their use-case, or practitioner use-cases may not contain format cues (e.g., MCQ vs. long-form) which are inherent in benchmark items. Therefore, we try different query-rewriting strategies, a standard way to bridge query-retrieval item mismatch (Rizvi et al., 2004; Antonellis et al., 2008; Ma et al., 2023) to generate the *anchors* that

minimize the effects of these constraints for our retrieval (Noël et al., 1997). Prompts and examples are in Appendix A. Throughout the work, we compare performance with Original, a no-rewrite baseline that uses the raw use-case.

We test three other anchor generation methods: **(a) Rephrasing:** Underspecified topics or capabilities can attract a noisy set of examples with weak relevance (Krovetz and Croft, 1992; Chen et al., 2013), so we rewrite a use-case as $n = 3$ specific subtopics or component skills. **(b) Example Synthesis:** Omitting format cues from the retrieval anchor can significantly degrade retrieval performance (Noël et al., 1997), hence, we generate $n = 3$ synthetic evaluation examples in formats common in our benchmark database, such as long-form QA, MCQ, and fill-in-the-blank, so that retrieval anchors resemble indexed benchmark items. **(c) Shorthand:** Because many benchmark items do not explicitly name the capability they test, we also create a bounded-vocabulary representation of the form $\langle \text{skill} \rangle \ \& \ \langle \text{key}_1 \rangle \ \& \ \langle \text{key}_2 \rangle \ \& \ \langle \text{key}_3 \rangle$. Each Shorthand preserves the primary skill plus salient topical cues. Unlike the other anchor methods, Shorthand retrieval maps both the practitioner use-case and all indexed examples into this shared representation, making latent capability labels explicit and encouraging items with related skills to cluster despite surface-format differences. We implement this efficiently by fine-tuning a Llama-3-8B-Instruct model on GPT-generated (sample, shorthand) pairs and translating the benchmark database once (detailed pipeline in §A.1).

2.2 Stage 2: Semantic Retrieval

For each anchor, we compute semantic embeddings with BAAI/bge-en-icl (Xiao et al., 2024) and retrieve nearest-neighbor examples using Faiss (Douze et al., 2024). We chose this backbone after comparing standard BGE bi-encoders, BAAI/llm-embedder, a cross-encoder reranker, and bge-en-icl on topic queries (Appendix B). bge-en-icl gives the best effectiveness-latency tradeoff while being significantly faster than the reranker. We retrieve the top- k nearest examples per anchor, pool them, sort by similarity, and pass the top candidates to filtering. We also compare against BM25 (Robertson et al., 2009) and random retrieval to confirm meaningful gains arise from semantic retrieval over standard lexical baselines.

2.3 Stage 3: Filtering

The top- k pooled candidates are then scored by an LLM judge. The judge checks if the example has *topical* or *skill* overlap with the use-case, and whether success on the example would provide credible evidence of competence for the use-case. This step aims to eliminate only the most obviously unrelated examples and not to outright select the relevant examples, as the latter is a non-trivial aspect to judge and is ultimately the practitioner’s prerogative. The selection prompt is in Appendix B.3.

3 Evaluation Setup

We evaluate BENCHBROWSER on different *evaluation query sets* designed to stress-test retrieval across query specificity, from broad topics (“Science”) to practitioner-specific needs unlikely to appear in existing benchmarks (“convert my paper abstract into a teaser figure”). These sets provide an evaluation scaffold to test our system and we describe them briefly below, grouped by their origin. Full details and prompts are in Appendix C.

Synthetically Constructed Evaluation Queries. ($N = 158$) We synthetically construct *topics*, *skills*, and *applications* query sets; The *topics* set covers broad knowledge areas, such as “sports” or “law,” drawn from 58 BBH (Suzgun et al., 2023) and MMLU subjects (Hendrycks et al., 2021). The *skills* are cross-domain capabilities, such as “reasoning” or “creative writing,” derived from real-world Clio use-cases of Claude and rewritten as broad skill descriptions using claude-sonnet-4-5-20250929 (Tamkin et al., 2024). *Applications* use the same source but describe concrete deployed tasks, such as “studying for the SAT,” mirroring the specific user requests common in instruction-following benchmarks (Qin et al., 2024; Zhou et al., 2023; Wen et al., 2024).

Crowdsourced Evaluation Queries representing Novel Practitioner Intent. ($N = 21$) To test real use-cases that may be absent from existing benchmarks, we crowdsource a *novel* set from $N = 9$ NLP practitioners (e.g., “creating Matplotlib visualizations for my research”). This set helps evaluate whether BENCHBROWSER can surface useful evidence for practitioner needs not already encoded as benchmark subjects.

Manually Curated Evaluation Queries for Existing NLP Benchmarks. ($N = 20$) In order to

Method	Topics				Skills			Applications		
	Prec _{sys} @ <i>k</i>	Prec _{anchor} @ <i>k</i>	NDCG@ <i>k</i>	Recall@ <i>k</i>	Prec _{sys} @ <i>k</i>	Prec _{anchor} @ <i>k</i>	NDCG@ <i>k</i>	Prec _{sys} @ <i>k</i>	Prec _{anchor} @ <i>k</i>	NDCG@ <i>k</i>
Random	0.304*	0.478*	0.702*	0.029*	0.310*	0.396*	0.628*	0.045*	0.051*	0.231*
BM25	0.529*	0.748*	0.850*	0.191*	0.654*	0.780*	0.823*	0.320*	0.309*	0.648*
Original	0.763*	0.887*	0.921*	0.350*	0.768*	0.923	0.911*	0.572	0.722	0.870
Rephrasing	0.812*	0.896*	0.923*	0.345*	0.775	0.933	0.908*	0.563	0.687	0.849
Shorthand	0.884	0.970	0.955*	0.414	0.815	0.925	0.916	0.572	0.611*	0.808*
Example Synthesis	0.890	0.968	0.971	0.423	0.837	0.926	0.937	0.560	0.687	0.869

Table 1: Unified Retrieval Performance @ $K = 20$: Sign-Flip Tests vs. Best Method Best values and second-best values per column are highlighted in **green** and **gold**, respectively. * marks methods significantly below the best method in that column by paired sign-flip tests with Holm correction ($p < 0.05$). Horizontal line separates neural embedding-based methods (lower) from non-neural baselines (upper).

evaluate retrieval methods in a controlled setting, we construct a set of use-case queries that are already targeted by existing NLP benchmarks. For example, “Using meal descriptions to measure calorie intake” is paired with NutriBench (Hua et al., 2024), an existing test set measuring the nutritional awareness of language models. We refer to these paired benchmark test sets as *gold* test sets and exclude them from our retrieval index explicitly. The full query-to-task mapping is in Appendix D.3.

Retrieval metrics. We grade retrieved items with gpt-5-mini-2025-08-07 as *Relevant*, *Partially Relevant*, or *Irrelevant*. For precision and recall, *Relevant* and *Partially Relevant* count as positives; this follows standard graded-relevance information retrieval decisions for exploratory search tasks (Voorhees, 1998) where partial matches can still expose users to plausible, non-dominant operationalizations of their intent (Wullach et al., 2026). We report Prec_{anchor}@*k* for unfiltered anchor outputs, Prec_{sys}@*k* for the final filtered system output, and NDCG@*k* for ranking quality. Prec_{anchor}@*k* is used only to select the anchor method, while Prec_{sys}@*k* is our main practitioner-facing retrieval metric. For the *topics* set, we also report Recall@*k* on a restricted 28k-example database containing only the relevant gold sets of examples; all other metrics are computed over the full 70k-example benchmark database listed in Appendix C.1.

4 BENCHBROWSER Finds Relevant Examples

Anchor Comparison. Table 1 compares anchoring strategies for retrieving relevant examples. Across the *topics* and *skills* sets, rewriting-based anchors (Rephrasing, Shorthand, Example Synthesis) substantially outperform Random and BM25, often by more than 10 points in Prec_{anchor}@*k* and

Prec_{sys}@*k*, and generally improve over the direct Original baseline. Example Synthesis and Shorthand are the strongest overall: Example Synthesis is near-best across all three query sets, while Shorthand performs comparably. Gains in NDCG@*k* show that rewriting improves ranking quality, not only precision of top-ranked items. Recall@*k* is conservative for all methods on the *topics* set: we attribute this to the presence of multiple highly overlapping use-cases in this set (e.g. high-school vs. college chemistry) due to which examples that are generally relevant to one use-case may be counted as misses for a closely related one. As a robustness check for our automatic evaluations, we replicate the entire evaluation with another LLM judge, gemini-2.5-flash-lite, and observe strong, statistically significant agreement ($\rho = 0.795$ for Prec_{sys}@*k*; Table 10, Appendix E).

Tie Breaking. Because Example Synthesis and Shorthand are statistically indistinguishable in Prec_{sys}@*k*, our primary metric, across every query class (Table 1), we use two tie-breaking experiments to decide the anchoring strategy for BENCHBROWSER. For the *topics* set, we test whether model performances on retrieved examples preserves the ranking of models on the gold test sets for the *topics* set (Appendix D.1). For the *skills* and *applications* sets, which lack gold test sets, we adopt the standard pooling methodology adopted in TREC for tasks when exhaustive ground-truth is intractable (Voorhees, 1998; Zobel, 1998). For each method, we measure the overlap of retrieved examples with the union of examples judged relevant by all anchoring methods (Appendix D.2). Example Synthesis performs best in both experiments, so we use it for all subsequent analyses.

User Interaction Metrics. We measure the end-to-end latency of the tool over a traffic of ≈ 200

Use-case	M_H	M_{LLM}	ρ	κ_{HH}	κ_{H-LLM}	$AC2_{HH}$	$AC2_{H-LLM}$	%Rel
Topics	0.96	0.92 [†]	0.15 [†]	0.10	0.01	0.94	0.90	94.3
Skills	0.81	0.91 [‡]	0.18 [†]	0.11	0.13	0.60	0.81	71.7
Applications	0.74	0.75	0.49 [‡]	0.29	0.32	0.55	0.67	62.8
Known Valid.	0.58	0.88 [‡]	0.34 [‡]	0.29	0.09	0.40	0.47	44.5
Novel	0.70	0.67	0.27 [‡]	0.13	0.14	0.36	0.43	56.5

Table 2: Mean relevance of retrieved examples by human annotators (M_H) and LLM judge (M_{LLM}). [†] $p < 0.05$; [‡] $p < 0.0001$. ρ is Spearman’s rho. κ_{HH} and κ_{H-LLM} are Fleiss’s kappa for human-human and human-LLM agreement. $AC2_{HH}$ and $AC2_{H-LLM}$ are Gwet’s AC2. %Rel is the % judged relevant by humans.

concurrent requests to be 5.28 ± 1.48 s supporting its interactive use. (See Appendix F for a breakdown of statistics). We also confirm that for Example Synthesis, $Prec_{sys}@k$ remains largely stable as k grows for at least the *topics* set and *skills* set (Appendix D.4). The relevance for *applications* set expectedly degrades more sharply, likely because highly contextual use-cases have fewer matching examples. Collectively, these results indicate that BENCHBROWSER can return reasonably sized pools of relevant items at interactive speeds.

4.1 Human Evaluation

Setup. We recruited $N = 10$ NLP practitioners to annotate the top-20 examples returned by the tool across 71 use cases. Each example was labeled by three annotators using the same three-point relevance scale as the LLM judge (recruitment details are in Appendix G). Annotators labeled all *novel* and *known validation* use-cases to measure LLM-human relevance calibration on the primary evaluation sets. They also labeled the top-20 examples for the ten highest-scoring *topics*, *skills*, and *applications* queries under the LLM judge to verify that high automatic relevance scores do not merely reflect LLM agreeableness (Jain et al., 2025; Arabzadeh and Clarke, 2025). This yields 4,260 human relevance judgments across 71 use cases.

Study Results. We map relevance labels to numeric scores to compare human and LLM judgments: *Relevant* = 1, *Partially Relevant* = 0.5, and *Irrelevant* = 0. Table 2 shows that mean human relevance is close to mean LLM-judge relevance for the *topics*, *applications*, and *novel* sets. The largest discrepancy occurs in the *known validation* set, likely due to a small number of esoteric benchmark tasks (such as BIG-bench Hard’s *novel_concepts*) that are difficult to compress into practitioner-style prompts (Appendix G.1).

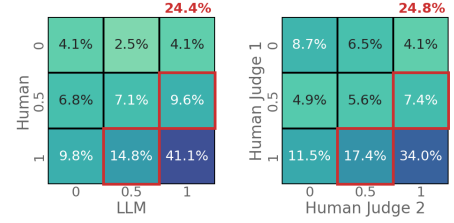


Figure 2: Distributions of example relevance judgments for the *novel* set shows most disagreements are between *Partially Relevant* and *Relevant*.

Table 2 also reveals that relevance judgments are strongly skewed toward the *Relevant* class. Because this class imbalance makes standard correlation metrics like ρ and κ unreliable (Bexkens et al., 2018; Gaudreault and Branco, 2024), we adopt Gwet’s AC2 as our primary agreement metric since it is more robust to class imbalance (Gwet, 2014). Agreement between different human annotators as measured with AC2 is moderate to high for most query sets, though low to moderate AC2 for the *known validation* and *novel* sets indicates a degree of subjectivity. Agreement between humans and LLM judges is moderate to high for all five query sets, with AC2 ranging from 0.43 for the *novel* set to 0.90 for the *topics* set.

Disagreement Analysis. The primary driver of disagreement is calibration between *Partially Relevant* and *Relevant* examples. Figure 2 illustrates this pattern for the lowest-agreement *novel* set: both human-human and human-LLM disagreements occur mainly between *Partially Relevant* and *Relevant*, indicating that lower AC2 is driven primarily by calibration near the *Relevant/Partially Relevant* boundary, not by catastrophic misses. This pattern holds for the other query sets as well (Appendix G.1). These calibration differences are consistent with prior work on subjective annotation tasks (Adams et al., 2023; McDonnell et al., 2016; Shirani et al., 2021; El Dehaibi and MacDonald, 2020), reinforcing the value of the practitioner-driven item-level inspection that BENCHBROWSER enables. Appendix G.1 provides agreement breakdowns and annotator comments supporting the challenge of distinguishing partial from full relevance.

5 Surfacing Content Validity Threats

Having established that BENCHBROWSER retrieves relevant item-level evidence, we now use it to surface content validity threats: cases where

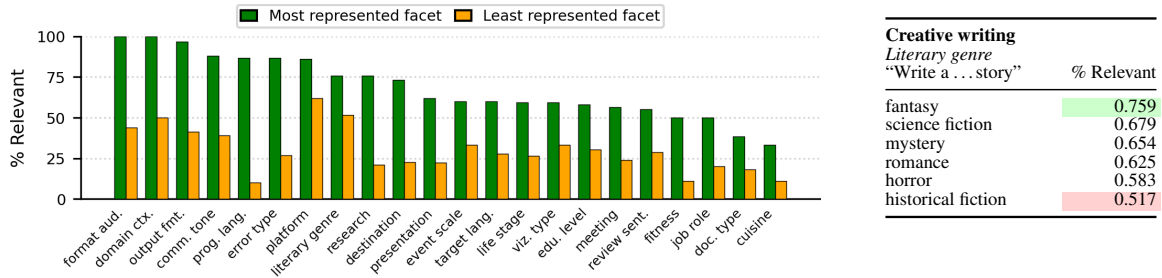


Figure 3: Facet-level disparities indicate low content validity. Left: in nearly all skill families, facets can have large differences in number of relevant retrieved examples. Right: one skill family from the bar plot: the disparity between retrieved evidence for “literary genre” between “fantasy” and “historical fiction”. Appendix H.1 provides more examples of such dispersion across additional skills.

benchmark items instantiate a capability too narrowly. Content validity concerns whether an evaluation adequately represents the facets of the ability it claims to measure (Strauss and Smith, 2009; Freiesleben and Zezulka, 2025; Wallach et al., 2025). When this threat is not surfaced, aggregate benchmark scores can appear to measure a broad capability while actually reflecting only a narrow operationalization. For example, if examples for “code generation” mostly involve short Python snippets, then performance on those examples says little about code generation in other languages or contexts. A natural way to probe this threat is facet analysis (Landsheer and Boeije, 2010): hold the broad capability fixed, such as “code generation,” while varying a single facet, such as programming language, and then inspect benchmark coverage. BENCHBROWSER operationalizes this workflow by retrieving evidence for practitioner-specified facets, allowing users to compare which variants of a capability are well represented, under-represented, or represented only through overlapping generic items.

Experimental Setup. This workflow requires skill families that enumerate variants of the same broad capability. We use the *skills* set as seeds and prompt `claude-sonnet-4-5-20250929` to produce 20 skill families, each with six closely related variants differing along one interpretable axis (120 total; e.g., a code generation skill family with programming language facet, a creative writing skill family with literary genre facet). We manually verify that variants are lexically separable so that changes in query wording do not confound retrieval. The right panel of Figure 3 shows a skill family with facets.

Evaluation. A fully principled human assessment would require agreement-checked judgments over both facet relevance and item relevance; this

is costly and cognitively burdensome because both are stakeholder-dependent, and related facets often retrieve overlapping near-duplicates (Strauss and Smith, 2009; Cartwright et al., 2019; Henley and Piorkowski, 2024). Accordingly, we use the LLM judge validated in §4.1 as a relevance proxy in this section. We compute the percentage of retrieved examples deemed *Relevant*, excluding *Partially Relevant* examples to avoid overstating coverage. Large disparities between facets within a skill family provide evidence of possibly low content validity.

Findings. The left panel of Figure 3 summarizes the percentage of *Relevant* examples for the best- and worst-represented facets of each skill family. Closely related use-case variants often yield very different numbers of *Relevant* examples, suggesting substantial spread in the retrievable evidence for different facets. The right panel gives one concrete example of this disparity demonstrating how for a skill like “creative writing”, testing “fantasy” story writing yields a large number of *Relevant* examples, whereas equally legitimate but less conventional facets like writing “historical fiction” receive much lower coverage despite targeting the same broad capability. Appendix H finds the same trends with a lexical BM25 baseline, suggesting that these disparities reflect benchmark composition rather than an artifact of the semantic retriever.

BENCHBROWSER’s interface allows users to similarly retrieve evidence across facets they deem relevant for their assessment (Step 2, Figure 1). Through this, BENCHBROWSER aims to help practitioners gauge both, *if* (low number of items retrieved) and *how diversely* (low number of unique samples) operationalize a practitioner’s capability of interest.

6 Surfacing Convergent Validity Threats

With the rapid proliferation of benchmarks, multiple benchmarks now often claim to evaluate the same capability (Sun et al., 2025; Zhang et al., 2025). This invites an often-untested assumption: benchmarks measuring the same skill should support similar conclusions about which models are better for that skill. *Convergent validity* formalizes this expectation as an agreement between model rankings induced by nominally similar evaluations (Strauss and Smith, 2009; Bean et al., 2025). When this agreement fails, benchmark results can yield contradictory conclusions (Diddee and Ippolito, 2024; Zhou et al., 2026) which can hinder actionable insights about which models are competent for a practitioner’s capability of interest. BENCHBROWSER helps investigate such failures by retrieving relevant examples for a practitioner’s use-case, allowing the practitioner to test whether rankings induced by existing benchmarks transfer to the retrieved sets of relevant examples.

Experimental setup. For this analysis, we need use-cases which are already measured by existing public benchmarks, so we use the *known validation* set (§3). For each use-case u , we evaluate $n = 8$ models on its pre-defined gold evaluation set G_u and the example set retrieved by BENCHBROWSER R_u . Each example is scored with its source benchmark metric, such as accuracy, exact match, or win rate, and scores are aggregated to induce a model ranking. An induced ranking is the ordering of the same eight models by performance on either G_u or R_u . Model and metric details are in Appendix I.

To separate retrieval-induced ranking changes from finite-sample instability, we compare rankings using two Kendall’s τ estimates averaged over 100 Monte Carlo trials:

- $\tau_{\text{ret}}(u)$ tests whether the retrieved set R_u induces the same model ordering as the gold benchmark G_u at a matched evaluation budget. In each trial, we compare the ranking induced by R_u with the ranking induced by $|R_u|$ subsampled examples from G_u .
- $\tau_{\text{gold}}(u)$ is a gold-only control for how much the benchmark’s own ranking changes under the same budget reduction. In each trial, we compare the full gold set ranking with a ranking induced by $|R_u|$ subsampled gold examples. This provides a task-specific ceiling on achievable rank agreement when evaluating models on fewer ex-

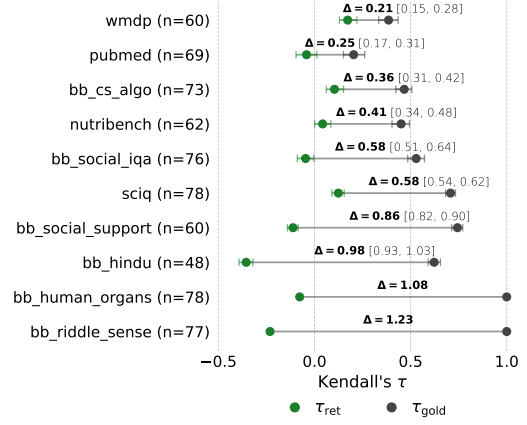


Figure 4: Kendall’s τ correlations of model rankings on gold test sets vs. retrieved sets for *known validation* set use-cases with mean human relevance ≥ 0.5 . Wider bars correspond to greater rank divergence $\Delta = \tau_{\text{gold}} - \tau_{\text{ret}}$. n : number of retrieved examples. 95% bootstrap confidence intervals for the mean over trials.

amples (Yauney et al., 2026).¹

We define rank divergence as $\Delta(u) = \tau_{\text{gold}}(u) - \tau_{\text{ret}}(u)$; large Δ indicates that retrieved relevant items induce ranking changes beyond gold-set sampling effects. To be extra cautious, we report only use-cases whose retrieved examples have mean *human relevance* at least *Partially Relevant*.

Findings. Figure 4 shows that rank divergence varies sharply across use-cases. Some retrieved sets preserve benchmark-induced model comparisons, while others produce large divergence or ranking inversions, suggesting that topical relevance alone is insufficient for establishing convergent validity.

Manual inspection suggests three recurring drivers of these inversions, reflecting the practical ways in which practitioners might encounter poor convergent validity (Table 3). First, *rubric shift* occurs when retrieved examples match the topic but change the format or scoring criterion, such as moving from short factual questions to longer, constraint-heavy, or open-ended prompts. Such metric changes can systematically alter model rankings, especially for win-rate-based generative evaluations that are sensitive to calibration, sometimes even spurious, artifacts (Campbell and Fiske, 1959; Dubois et al., 2024; Gao et al., 2024b; Zheng et al., 2025). Second, *divergent operationalization* occurs when benchmarks that advertise the same capability test different task families, yield-

¹Appendix I reports same-cardinality gold–gold correlations as an additional sanity check.

Observed Cause 1: Rubric Shift*Example Query: “Hindu Religion”***BIG-bench Hard (Hindu Knowledge):** Which of the following gods are associated with a flying elephant**HellaSwag:** Philosophy and Religion: How to learn about ancient hinduism ...

Observed Cause 2: Divergent Operationalization*Example Query: “Solving Puzzles”***BIG-bench Hard (Colored Objects):** On the desk, you see items in a row: a gold textbook, a purple puzzle, a teal necklace, and a silver pencil. How many non-gold items do you see to the right of the pencil?**AlpacaEval:** Hi, I’m trying to solve a crossword puzzle, but I’ve never done one of these before. Can you help?

Observed Cause 3: Context Shift (Recall → Applied)*Example Query: “Writing Programs”***HumanEval:** Check if in given list of numbers, are any two numbers closer to each other than given threshold.**InfoBench:** ... you’ve came up with an algorithm for solving the coding question your interviewer gave you but

Table 3: Observed causes of ranking inversions; gold and blue highlights indicate an example is from the gold set or the retrieved set, respectively.

ing construct non-equivalence. For example, for a use-case like “solving puzzles”, BIG-bench Hard’s `riddle_sense` may be relevant but too specific to riddles to engender the same ranking as a benchmark testing logic puzzles, even though they both test related reasoning skills. Finally, a *context shift* can be observed when a use-case’s evaluation shifts from purely testing recall to testing it’s applied use.

7 Related Work

Diagnosing low validity for AI benchmarks.

Recent audits stress grounding AI evaluation in validity measurements (Alaa et al., 2025; Salaudeen et al., 2025; Ye et al., 2025), revealing that LLM benchmarks often underspecify links between scores, measured constructs, and their operationalizations (Dietz et al., 2025; Reuel-Lamparth et al., 2024; Wallach et al., 2025). This underspecification degrades reliability, especially for multifaceted capabilities. Since gathering validity evidence is costly—requiring expert annotation (Bean et al., 2025), explicit operationalization (Zeng et al., 2025; Bohacek et al., 2026), or stakeholder definitions (Wallach et al., 2025; Guerdan et al., 2025)—BENCHBROWSER targets this bottleneck by retrieving inspectable, item-level evidence for practitioner-defined use cases, enabling systematic meta-evaluations without fixed taxonomies.

Improving open-ended evaluation. While little work addresses validity inspection directly, open-ended evaluation methods—including custom benchmark synthesis (Shashidhar et al., 2025; Pombal et al., 2025), item-level annotation (Zeng et al., 2025; Li et al., 2025; Kim et al., 2025), and psychometric composition (Hofmann et al., 2025)—instantiate relevant methodologies. Most notably, Ghosh et al. (2025) dynamically rank query-conditioned items for custom use-cases. Unlike BENCHBROWSER, they focus on inducing a ranking for the use-case as opposed to assessing if retrieved query-conditioned items actually instantiate the intended construct. Similarly, while Bohacek et al. (2026) probe internal competency gaps using SAEs—restricted to what model internals realize—we focus on empowering practitioners to elicit these gaps directly. By reusing *vetted* benchmark items (Liu et al., 2024), we map existing operationalizations and avoid synthetic generation biases (Huang et al., 2024; Chen et al., 2025) that can often degrade purely synthetic testsets (Pombal et al., 2025; Shashidhar et al., 2025). We anticipate that such inspection, *prior*, to evaluation prevents curating redundant, non-discriminative test data (Maheshwari et al., 2024) and calibrates expectations for stable competence estimates in emerging capabilities (Diddee and Ippolito, 2024).

8 Conclusion

Validating LLM benchmarks for emerging, multifaceted capabilities requires moving beyond aggregate scores to directly audit exactly *what* is being measured and *how*. To facilitate this, we present BENCHBROWSER, a meta-evaluation tool that supports cost-effective, transparent validity inspection for practitioner-defined use-cases. Rather than relying on any single evaluation suite (Veuthey et al., 2025; Son et al., 2024; Chern et al., 2024), BENCHBROWSER surfaces atomic evidence across multiple benchmarks, exposing the diverse ways abstract capabilities are actually operationalized. Our evaluations, corroborated by human annotators assessing the relevance of retrieved data, show that BENCHBROWSER accurately retrieves relevant evidence for assessing coverage and representation. Making sources, rubrics, and framings inspectable allows practitioners to rapidly inspect content and convergent validity, revealing the specific subsets, outliers, or artifacts driving ranking inversions. In doing so, BENCHBROWSER engenders more stable, con-

textualized capability claims and directly guides where future benchmark development should focus to address underrepresented facets.

Limitations

Our human evaluation is limited in scale: to keep annotation costs manageable, we restrict the number and diversity of use-cases and annotators. All annotators were located in the United States. Although we collect 4,260 relevance judgments, precision and coverage estimates may not fully generalize beyond the evaluated use-cases. An end-to-end human evaluation of the tool would require a larger time and resource budget to compensate annotators for several iterations of tool use, which was outside the scope of this project. Such an evaluation is an important direction for future work.

The content-validity analysis in §5 does not elicit or verify facets with annotators, because doing so would require an additional layer of costly human validation. Because these facets are generated by a language model, they may reflect biased or incomplete facet selection. We expect that an LLM judge can provide a useful preliminary signal of representational disparities, and expect that this signal will be strengthened with human supervision during the tool’s actual use. Consequently, the analysis in this section is intended primarily to demonstrate potential representational disparities for practitioner inspection, rather than to make actionable claims about which facets require additional benchmark coverage.

While BENCHBROWSER is intended for diagnosing benchmark validity, its underlying retrieval mechanism could, in theory, be used to cherry-pick evaluation sets that make a given model look best.

Finally, BENCHBROWSER can surface only the validity concerns reflected in its indexed benchmark items and evaluated model outputs. Expanding coverage to more benchmark suites, models, and outputs from commercial models required by some metrics is resource-intensive. As a result, any conclusions drawn from the tool are bounded by the current database and the feasible set of evaluated models.

Acknowledgments

This research was supported by a Gemini Credits Grant. We thank Fernando Diaz, whose course first introduced us to the concept of validity and its central role in measurement. We are grateful

to Joachim Baumann for insightful feedback that helped improve the clarity of this draft. We also thank Kabir Ahuja and Valentin Hofmann for helpful discussions on validity, Varsha Kishore for helpful discussions about retrieval, and members of SS’s and DI’s groups for their comments on improving the clarity of the work. Finally, we thank Saujus Vaduguru and Kyzyl Monteiro for their assistance in organizing the user study for BENCHBROWSER.

References

- Griffin Adams, Alex Fabbri, Faisal Ladhak, Eric Lehman, and Noémie Elhadad. 2023. From sparse to dense: GPT-4 summarization with chain of density prompting. In *Proceedings of the 4th New Frontiers in Summarization Workshop at EMNLP*, pages 68–74.
- Kabir Ahuja, Melanie Sclar, and Yulia Tsvetkov. 2025. Finding flawed fictions: Evaluating complex reasoning in language models via plot hole detection. In *COLM*.
- Ahmed Alaa, Thomas Hartvigsen, Niloufar Golchini, Shiladitya Dutta, Frances Dean, Inioluwa Deborah Raji, and Travis Zack. 2025. Medical large language model benchmarks should prioritize construct validity. In *ICML Position Papers Track*.
- Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In *NAACL*.
- Ioannis Antonellis, Hector Garcia-Molina, and Chih-Chao Chang. 2008. [Simrank++: query rewriting through link analysis of the clickgraph](#). In *WWW*.
- Negar Arabzadeh and Charles LA Clarke. 2025. Benchmarking LLM-based relevance judgment methods. In *SIGIR*, pages 3194–3204.
- Andrew M. Bean, Ryan Othniel Kearns, Angelika Romanou, Franziska Sofia Hafner, Harry Mayne, Jan Batzner, Negar Foroutan, Chris Schmitz, Karolina Korgul, Hunar Batra, Oishi Deb, Emma Beharry, Cornelius Emde, Thomas Foster, Anna Gausen, Mar’ia Grandury, Simeng Han, Valentin Hofmann, Lujain Ibrahim, and 23 others. 2025. [Measuring what matters: Construct validity in large language model benchmarks](#). In *NeurIPS Datasets and Benchmarks Track*.
- Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *EMNLP*.
- Rens Bexkens, Femke MAP Claessen, Izaak F Kodde, Luke S Oh, Denise Eygendaal, and Michel PJ van

- den Bekerom. 2018. The kappa paradox. *Shoulder & Elbow*, 10(4):308–308.
- Matyas Bohacek, Nino Scherrer, Nicholas Dufour, Thomas Leung, Christoph Bregler, and Stephanie CY Chan. 2026. Uncovering competency gaps in large language models and their benchmarks. In *ICML*.
- Donald T Campbell and Donald W Fiske. 1959. Convergent and discriminant validation by the multitrait-multimethod matrix. *Psychological bulletin*, 56(2):81.
- Mark Cartwright, Graham Dove, Ana Elisa Méndez Méndez, Juan P. Bello, and Oded Nov. 2019. [Crowdsourcing multi-label audio annotation tasks with citizen scientists](#). In *CHI*.
- Long Chen, Dell Zhang, and Mark Levene. 2013. Question retrieval with user intent. In *SIGIR*.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde De Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.
- Yen-Shan Chen, Jing Jin, Peng-Ting Kuo, Chao-Wei Huang, and Yun-Nung Chen. 2025. LLMs are biased evaluators but not biased for retrieval augmented generation. In *ACL Findings*.
- Steffi Chern, Ethan Chern, Graham Neubig, and Pengfei Liu. 2024. [Can large language models be trusted for evaluation? Scalable meta-evaluation of LLMs as evaluators via agent debate](#). *ArXiv preprint arXiv:2401.16788*.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? Try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A Smith, and Matt Gardner. 2021. A dataset of information-seeking questions and answers anchored in research papers. In *NAACL*.
- Harshita Diddee and Daphne Ippolito. 2024. Chasing random: Instruction selection strategies fail to generalize. In *NAACL Findings*.
- Laura Dietz, Oleg Zendel, Peter Bailey, Charles Clarke, Ellese Cotterill, Jeff Dalton, Faegheh Hasibi, Mark Sanderson, and Nick Craswell. 2025. LLM-evaluation tropes: Perspectives on the validity of LLM-evaluations. In *SIGIR Conference on Innovative Concepts and Theories in Information Retrieval*.
- Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The Faiss library. *arXiv preprint arXiv:2401.08281*.
- Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. 2024. Length-controlled AlpacaEval: A simple way to debias automatic evaluators. In *COLM*.
- N El Dehaibi and EF MacDonald. 2020. Investigating inter-rater reliability of qualitative text annotations in machine learning datasets. In *Proceedings of the Design Society: DESIGN Conference*, volume 1, pages 21–30. Cambridge University Press.
- Denis Emelin, Ronan Le Bras, Jena D. Hwang, Maxwell Forbes, and Yejin Choi. 2021. [Moral stories: Situated reasoning about norms, intents, actions, and their consequences](#). In *EMNLP*.
- Maria Eriksson, Erasmo Purificato, Arman Noroozian, Joao Vinagre, Guillaume Chaslot, Emilia Gomez, and David Fernandez-Llorca. 2025. Can we trust AI benchmarks? An interdisciplinary review of current issues in AI evaluation. *arXiv preprint arXiv:2502.06559*.
- Timo Freiesleben and Sebastian Zenz. 2025. The benchmarking epistemology: Construct validity for evaluating machine learning models. *arXiv preprint arXiv:2510.23191*.
- Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac’h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2024a. [A framework for few-shot language model evaluation](#).
- Yicheng Gao, Gonghan Xu, Zhe Wang, and Arman Cohan. 2024b. [Bayesian calibration of win rate estimation with LLM evaluators](#). In *EMNLP*.
- Jean-Gabriel Gaudreault and Paula Branco. 2024. Empirical analysis of performance assessment for imbalanced classification. *Machine Learning*, 113(8):5533–5575.
- Adhiraj Ghosh, Sebastian Dziadzio, Ameya Prabhu, Vishaal Udandara, Samuel Albanie, and Matthias Bethge. 2025. ONEbench to test them all: Sample-level benchmarking over open-ended capabilities. In *ACL*.
- Luke Guerdan, Devansh Saxena, Stevie Chancellor, Zhiwei Steven Wu, and Kenneth Holstein. 2025. Measurement as bricolage: Examining how data scientists construct target variables for predictive modeling tasks. *arXiv preprint arXiv:2507.02819*.
- Kilem L Gwet. 2014. *Handbook of inter-rater reliability: The definitive guide to measuring the extent of agreement among raters*. Advanced Analytics, LLC.

- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *ICLR*.
- Austin Z Henley and David Piorkowski. 2024. Supporting annotators with affordances for efficiently labeling conversational data. *arXiv preprint arXiv:2403.07762*.
- Valentin Hofmann, David Heineman, Ian Magnusson, Kyle Lo, Jesse Dodge, Maarten Sap, Pang Wei Koh, Chun Wang, Hannaneh Hajishirzi, and Noah A Smith. 2025. Fluid language model benchmarking. In *COLM*.
- Andong Hua, Mehak Preet Dhaliwal, Ryan Burke, and Yao Qin. 2024. [NutriBench: A dataset for evaluating large language models on nutrition estimation from meal descriptions](#). In *ICLR*.
- Dong Huang, Jie M Zhang, Qingwen Bu, Xiaofei Xie, Junjie Chen, and Heming Cui. 2024. Bias testing and mitigation in LLM-based code generation. *ACM Transactions on Software Engineering and Methodology*.
- Suryaansh Jain, Umair Z Ahmed, Shubham Sahai, and Ben Leong. 2025. Beyond consensus: Mitigating the agreeableness bias in LLM judge evaluations. *arXiv preprint arXiv:2510.11822*.
- Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *EMNLP*.
- Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*.
- Jushaan Singh Kalra, Xinran Zhao, To Eun Kim, Fengyu Cai, Fernando Diaz, and Tongshuang Wu. 2025. MoR: Better handling diverse queries with a mixture of sparse, dense, and human retrievers. In *EMNLP*.
- Eunsu Kim, Haneul Yoo, Guijin Son, Hitesh Patel, Amit Agarwal, and Alice Oh. 2025. BenchHub: A unified benchmark suite for holistic and customizable LLM evaluation. *arXiv preprint arXiv:2506.00482*.
- Robert Krovetz and W Bruce Croft. 1992. Lexical ambiguity and information retrieval. *ACM Transactions on Information Systems*, 10(2):115–141.
- JA Landsheer and HR Boeije. 2010. In search of content validity: Facet analysis as a qualitative method to improve questionnaire design: An application in health research. *Quality & Quantity*, 44(1):59–69.
- Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, and 1 others. 2024. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *ICML*.
- Xiang Lisa Li, Farzaan Kaiyom, Evan Zheran Liu, Yifan Mai, Percy Liang, and Tatsunori Hashimoto. 2025. AutoBench: Towards declarative benchmark construction. In *ICLR*.
- Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *ACL*.
- Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2021. LogiQA: a challenge dataset for machine reading comprehension with logical reasoning. In *IJCAI*.
- Yu Lu Liu, Su Lin Blodgett, Jackie Cheung, Q. Vera Liao, Alexandra Olteanu, and Ziang Xiao. 2024. [ECBD: Evidence-centered benchmark design for NLP](#). In *ACL*.
- Da Ma, Gonghu Shang, Zhi Chen, Libo Qin, Yijie Luo, Lei Pan, Shuai Fan, Lu Chen, and Kai Yu. 2025. Task-specific data selection for instruction tuning via monosemantic neuronal activations. In *NeurIPS*.
- Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting in retrieval-augmented large language models. In *EMNLP*.
- Gaurav Maheshwari, Dmitry Ivanov, and Kevin El Hadad. 2024. Efficacy of synthetic data as a benchmark. *arXiv preprint arXiv:2409.11968*.
- Tyler McDonnell, Matthew Lease, Mucahid Kutlu, and Tamer Elsayed. 2016. [Why is that relevant? collecting annotator rationales for relevance judgments](#). In *HCOMP*.
- Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? A new dataset for open book question answering. In *EMNLP*.
- Marie-Pascale Noël, Wim Fias, and Marc Brysbaert. 1997. About the influence of the presentation format on arithmetical-fact retrieval processes. *Cognition*, 63(3):335–374.
- José Pombal, Nuno M Guerreiro, Ricardo Rei, and André FT Martins. 2025. Zero-shot benchmarking: A framework for flexible and scalable automatic evaluation of language models. In *COLM*.
- Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuansheng Wu, Fei Liu, Pengfei Liu, and Dong Yu. 2024. InFoBench: Evaluating instruction following ability in large language models. In *ACL Findings*.
- Anka Reuel-Lamparth, Amelia Hardy, Chandler Smith, Max Lamparth, Malcolm Hardy, and Mykel J Kochenderfer. 2024. Betterbench: Assessing AI benchmarks, uncovering issues, and establishing best practices. In *NeurIPS*.
- Shariq Rizvi, Alberto Mendelzon, Sundararajao Sudarshan, and Prasan Roy. 2004. Extending query rewriting techniques for fine-grained access control. In *ACM SIGMOD*.

- Stephen Robertson, Hugo Zaragoza, and 1 others. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389.
- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. WinoGrande: An adversarial winograd schema challenge at scale. In *AAAI*.
- Olawale Salaudeen, Anka Reuel, Ahmed Ahmed, Suhana Bedi, Zachary Robertson, Sudharsan Sundar, Ben Domingue, Angelina Wang, and Sanmi Koyejo. 2025. Measurement to meaning: A validity-centered framework for AI evaluation. *arXiv preprint arXiv:2505.10573*.
- Alireza Salemi, Surya Kallumadi, and Hamed Zamani. 2024. Optimization methods for personalizing large language models through retrieval augmentation. In *SIGIR*.
- Tefko Saracevic. 2007. Relevance: A review of the literature and a framework for thinking on the notion in information science. part II: Nature and manifestations of relevance. *Journal of the American Society for Information Science and Technology*, 58(13):1915–1933.
- Sumuk Shashidhar, Clémentine Fourier, Alina Lozovskia, Thomas Wolf, Gokhan Tur, and Dilek Hakkani-Tür. 2025. YourBench: Easy custom evaluation sets for everyone. In *COLM*.
- Amirreza Shirani, Giai Tran, Hieu Trinh, Franck Dernoncourt, Nedim Lipka, Paul Asente, Jose Echevarria, and Thamar Solorio. 2021. Learning to emphasize: Dataset and shared task models for selecting emphasis in presentation slides. In *Content Authoring and Design (CAD21) Workshop at AAAI*.
- Guijin Son, Dongkeun Yoon, Juyoung Suk, Javier Aula-Blasco, Mano Aslan, Vu Trong Kim, Shayekh Bin Islam, Jaume Prats-Cristià, Lucía Tormo-Bañuelos, and Seungone Kim. 2024. MM-Eval: A multilingual meta-evaluation benchmark for LLM-as-a-judge and reward models. *arXiv preprint arXiv:2410.17578*.
- Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adri Garriga-Alonso, and 1 others. 2023. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *TMLR*.
- Milton E. Strauss and Gregory T. Smith. 2009. [Construct validity: advances in theory and methodology](#). *Annual Review of Clinical Psychology*, 5:1–25.
- Jiankai Sun, Chuanyang Zheng, Enze Xie, Zhengying Liu, Ruihang Chu, Jianing Qiu, Jiaqi Xu, Mingyu Ding, Hongyang Li, Mengzhe Geng, and 1 others. 2025. A survey of reasoning with foundation models: Concepts, methodologies, and outlook. *ACM Computing Surveys*, 57(11):1–43.
- Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, and 1 others. 2023. Challenging BIG-bench tasks and whether chain-of-thought can solve them. In *ACL Findings*.
- Alex Tamkin, Miles McCain, Kunal Handa, Esin Durmus, Liane Lovitt, Ankur Rath, Saffron Huang, Alfred Mountfield, Jerry Hong, Stuart Ritchie, and 1 others. 2024. Clio: Privacy-preserving insights into real-world AI use. *arXiv preprint arXiv:2412.13678*.
- Jaime Raldua Veuthey, Zainab Ali Majid, Suhas Harisharan, and Jacob Haimès. 2025. MEQA: A meta-evaluation framework for question & answer LLM benchmarks. *arXiv preprint arXiv:2504.14039*.
- David Vilares and Carlos Gómez-Rodríguez. 2019. HEAD-QA: A healthcare dataset for complex reasoning. In *ACL*.
- Ellen M Voorhees. 1998. Variations in relevance judgments and the measurement of retrieval effectiveness. In *SIGIR*.
- Hanna Wallach, Meera Desai, A Feder Cooper, Angelina Wang, Chad Atalla, Solon Barocas, Su Lin Blodgett, Alexandra Chouldechova, Emily Corvi, P Alex Dow, and 1 others. 2025. Position: Evaluating generative AI systems is a social science measurement challenge. In *ICML Position Papers Track*.
- Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. [Crowdsourcing multiple choice science questions](#). In *Workshop on Noisy User-generated Text*. Association for Computational Linguistics.
- Bosi Wen, Pei Ke, Xiaotao Gu, Lindong Wu, Hao Huang, Jinfeng Zhou, Wenchuang Li, Binxin Hu, Wendy Gao, Jiaying Xu, and 1 others. 2024. Benchmarking complex instruction-following with multiple constraints composition. *NeurIPS*.
- Tomer Wulach, Ori Shapira, and Amir DN Cohen. 2026. The overlooked role of graded relevance thresholds in multilingual dense retrieval. In *ACL Findings*.
- Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muenighoff, Defu Lian, and Jian-Yun Nie. 2024. C-Pack: Packaged resources to advance general chinese embedding. In *SIGIR*.
- Gregory Yauney, Shahzaib Saqib Warraich, and Swabha Swayamdipta. 2026. How reliable is language model micro-benchmarking? In *ICLR*.
- Haoran Ye, Jing Jin, Yuhang Xie, Xin Zhang, and Guojie Song. 2025. Large language model psychometrics: A systematic review of evaluation, validation, and enhancement. *arXiv preprint arXiv:2505.08245*.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *ACL*.

- Zhiyuan Zeng, Yizhong Wang, Hannaneh Hajishirzi, and Pang Wei Koh. 2025. EvalTree: Profiling language model weaknesses via hierarchical capability trees. In *COLM*.
- Peitian Zhang, Zheng Liu, Shitao Xiao, Zhicheng Dou, and Jian-Yun Nie. 2024. [A multi-task embedder for retrieval augmented LLMs](#). In *ACL*.
- Zicheng Zhang, Xiangyu Zhao, Xinyu Fang, Chunyi Li, Xiaohong Liu, Xiongkuo Min, Haodong Duan, Kai Chen, and Guangtao Zhai. 2025. Redundancy principles for MLLMs benchmarks. In *ACL*.
- Xiaosen Zheng, Tianyu Pang, Chao Du, Qian Liu, Jing Jiang, and Min Lin. 2025. [Cheating automatic LLM benchmarks: Null models achieve high win rates](#). In *ICLR*.
- Hongli Zhou, Hui Huang, Ziqing Zhao, Lvyuan Han, Huicheng Wang, Kehai Chen, Muyun Yang, Wei Bao, Jian Dong, Bing Xu, and 1 others. 2026. Lost in benchmarks? Rethinking large language model benchmarking with item response theory. In *AAAI*.
- Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. 2023. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*.
- Justin Zobel. 1998. How reliable are the results of large-scale information retrieval experiments? In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 307–314.

A Anchor Generation Methods

A.1 Shorthand Anchor

Many benchmarks (e.g., WINOGRANDE, BIG-bench) lack fine-grained capability annotations, making it difficult to identify which skills are being tested at the datapoint level. To ensure semantic proximity among functionally similar samples across diverse formats, we design structured textual representations called *shorthands*.

Shorthand format. A shorthand follows the template $\langle \text{skill} \rangle \& \langle \text{key}_1 \rangle \& \langle \text{key}_2 \rangle \& \langle \text{key}_3 \rangle$, where $\langle \text{skill} \rangle$ denotes the primary cognitive ability required, and keys encode salient lexical or contextual attributes that enhance retrieval precision.

Pipeline.

1. **Seed Data construction.** To train a model to efficiently translate our index into the Shorthand representation, we needed to generate a small, high quality translation dataset where a example corresponds to its equivalent shorthand. To achieve a diverse set of examples to act as this set - we first embed 10 random subsets of our entire index using the BAAI bge-en-icl embedding model. Each subset contains 1000 examples (without replacement). We, then, run t-SNE on each subset separately and k-means per batch (with $k = 10$). Finally, we pick, both, the $n = 250$ nearest-to-centroid and $n = 250$ farthest-from-centroid examples in each cluster. Choosing these allows us to cover both, the prototypical and well represented tasks as well as the least represented tasks for the initial seed data construction. This allows us to generate a dataset consisting of 5000 examples.
2. **Finetuning Data Synthesis.** We then use gpt-5-mini-2025-08-07 to produce shorthands for the 5000 examples. We use the following prompt for this conversion.

Shorthand Generation Prompt ""You are tasked with converting natural language text into a concise shorthand format which can be used to identify other datapoints having similar capabilities. The format for creating this shorthand is: $\langle \text{skill} \rangle \& \langle \text{key}_1 \rangle \& \langle \text{key}_2 \rangle \& \langle \text{key}_3 \rangle$ where:
- $\langle \text{skill} \rangle$ encodes the primary cogni-

tive/technical ability needed to accomplish the task underlying the text. Ask yourself: "What is the primary cognitive/technical ability needed to accomplish the task underlying the text?" Some examples (NOT EXHAUSTIVE): creative_writing, coding, equation_solving, coreference_resolution, strategy_planning, logical_reasoning, factual_recall, reading_comprehension, scientific_visualization, graphics, social_interaction, movie_trivia, statistical_analysis, etc.

- $\langle \text{key}_1 \rangle, \langle \text{key}_2 \rangle, \langle \text{key}_3 \rangle$ are 1-3 SPECIFIC and DISTINCTIVE features that maximize retrieval precision; For adding each feature/key, ask yourself: - "What are the specific document types, topics, and/or entities in this text" that need to be preserved for retrieving similar datapoints? - Is there any specific topic underlying being asked by the user query: "machine learning" "statistics" "poetry" etc - include that as a key. - Is there any specific output format mentioned in the query: "research papers", "clinical cases", "financial reports" "emails" etc - include that as a key. - Is the the query about a specific person, organization, or entity - include that as a key. If there are multiple entities, or placeholders like "X", "Y", "Z" - include the broad category of the entity (e.g., "actor", "writer", "sportsperson" "female") and number of entitites as a key. - Is the user looking for a specific task like: "sorting_ascending", "counting_3_objects", "implementing_neural_network"- include that as a key. - If the text includes a mathematical formula: include the concept type of the formula (e.g., "differential_equations", "linear_algebra", "probability_distribution") - If the user has only specificed a single topic - use that as the key. DO NOT ADD A SKILL UNLESS ITS SPECIFIED. - If the user has only specified a single skill/task - use that skill/task as the key. DO NOT ADD A DIFFERENT TASK UNLESS ITS SPECIFIED. - Use underscores for multi-word concepts (e.g., "machine_learning", "social_skills", "new_york") - Make

sure that no key adds a new generic domain/skill/topic to the query. **STRICTLY FOLLOW THIS RULE.** - Separate each key with an ampersand. - Return exactly 1 shorthand for a user query wrapped in XML tags: `<shorthand></shorthand>`. - **IMPORTANT:** Only include keys, if they add unique, searchable value. If there are no more distinctive features, USE FEWER KEYS rather than padding with generic terms. Your final outputs should be of the form: `<shorthand> </shorthand>` Here are some examples of shorthands:

3. **Model Finetuning.** We then finetune Llama-3-8B-Instruct via on (example, Shorthand) pairs. We fine-tuned the base model using 4-bit quantization (QLoRA) with a maximum sequence length of 4096 tokens. Parameter Efficient Fine-Tuning (PEFT) was applied via Low-Rank Adaptation (LoRA) with a rank of $r = 64$, a scaling factor $\alpha = 32$, and a dropout rate of 0. The LoRA adapters targeted the query, key, value, output, gate, up, and down projection modules. Training was performed for 2 epochs using the 8-bit AdamW optimizer with a learning rate of 1×10^{-4} and a linear decay schedule with a warmup ratio of 0.03. We utilized a per-device batch size of 2 with 8 gradient accumulation steps, resulting in an effective batch size of 16. To ensure reproducibility, the random seed was set to 3407.
4. **Index translation.** We use our finetuned version of the model to translate all benchmark examples from our benchmark database into their respective shorthand notation. This step yields cost-efficient annotation, constrains the vocabulary, and enforces consistent skill verbalization across all the examples in the transformed benchmark database. Some examples from the final benchmark database are provided below.

Once this translated benchmark database is generated, we create the FAISS (Douze et al., 2024) index for this benchmark database to use for retrieval when Shorthand is used as an anchor generation strategy.

A.2 Anchor Generation Prompts

Rephrasing Baseline.

Segment Generation Prompt

""Given a user query, create **three** brief refinements^{**} of that query that are appropriate for retrieving examples representing the interest highlighted in the query. Use the following rules:

- If the query is compositional, i.e., it contains multiple concepts, make different operationalizations of each concept. For example: if the query is 'scientific visualizations for scientific papers' - each refinement should focus on a different sub-part of this query: like scientific visualizations, visualizations for scientific papers and tools for visualizations.
- If the query is about some broad capability, enlist different sub-topics that test for that capability. For example: if the query is 'trivia' - each refinement can include common topics included in trivia questions like 'celebrity history', 'geography', 'world history', etc.
- If the query is about some broad topic, enlist different related domains and sub-topics that are related to that topic. For example: if the query is 'biology' - each refinement can include different related domains like 'genetics', 'ecology', 'dermatology', etc.
- If the query is about a specific task, generate different contextual forms of that task. For example: if the query is 'Creative Writing' - each refinement can include different types of creative writing like 'poetry', 'short stories', 'novels', etc.
- If the query describes a scenario - 'Cooking indian food' - list different activities that can be performed in that scenario. For example: 'indian spices', 'indian food recipes' and 'indian desserts'.
- If the query is a proper Noun - 'Harry Potter' - generate at least one query that highlights the broad task associated with that proper noun, for example: 'fantasy literature' or 'magic', and one that lists the most popular form of that proper noun, for example, 'Harry Potter Books'.
- If the query is a question - 'What is the Capital of France' - generate at least one query that highlights the topic of that question. For example: 'Capitals of different countries', 'Facts about France' and 'Paris'.
- If it falls under neither of these categories, for example: 'Can Elephants fly' - generate queries

that cover reasonable aspects of that query: For example: ‘Animals that can fly’, ‘Elephant facts’ and ‘Flying facts’.

* Make sure that one refinement is not a subset of another refinement. * Make sure that at least one refinement highlights the broad domain of the query. For example, if the query is ‘chess’ - one refinement should include ‘board games’. * Avoid making overly complicated refinements by using simple words and phrases. * Focus on what would attract different types of relevant test cases or documents from existing test benchmarks. ALWAYS Return your responses in the following XML format; each refinement should be in a separate <refinement> tag: <refinements> <refinement> </refinement> </refinements> DO NOT INCLUDE ANY OTHER TEXT IN YOUR RESPONSE or any reasoning in your response. The user query is: query

Example Synthesis Baseline.

Testcase Generation Prompt
 ""Given a user query, generate 3 testcases that is representative of the user’s interest that can help evaluate an AI Model on its competence on the users intent. The rules for generating the testcases are as follows: - The testcase should be STRICTLY between 1-2 lines and not very verbose. - Each testcase can be of a different format (Long Form, MCQ, Fill in the blank, etc.) - Only include the question/input for a testcase and not the expected output. - Return each testcase wrapped in a separate <testcase> </testcase> tag. Your final outputs should be of the form: <testcases> <testcase> Some question related to the query </testcase> <testcase> Some question related to the query </testcase> <testcase> Some question related to the query </testcase> </testcases> DO NOT INCLUDE ANY OTHER TEXT IN YOUR RESPONSE or any reasoning in your response. - The user query is: {query} ""

A.3 Examples of Anchors

The table provides an example of all the anchors produced for the use-case: “understanding geopolitical tensions in Southeast Asia”.

Original: understanding geopolitical tensions in Southeast Asia
Rephrasing: South China Sea maritime disputes
Example Synthesis: Discuss the factors driving contemporary geopolitical tensions in Southeast Asia
Shorthand: geopolitical_analysis & southeast_asia

Table 4: Four methods for producing retrieval anchors.

B Embedding-Based Semantic Retrieval Details

B.1 In Context Examples for the Embedding Models (used during Original, Example Synthesis and Rephrasing Index Creation)

This subsection provides the In-Context Examples used for the BGE-en-icl model in Table 5.

use-case	Relevant example
Verify scientific claims	Claim: AdaBERT achieves inferior performance while significantly worsening the efficiency by 12.7x. Evidence: The dataset contains 3772 word pairs. The accuracy of ConVecs 70.02% is not significantly different from the accuracy of SimDiffs (72.4%). Label: Negative/Refuted.
Blogging	Write a blog announcing the opening of a new mall in the town of Serenity.
Unit Testing Code	Question: An anagram is the result of rearranging the letters of a word to produce a new word. Note: anagrams are case insensitive Complete the function to return true if the two arguments given are anagrams of each other; return false otherwise. Examples: foefet is an anagram of toffee. ut_id: 0 code - import unittest class TestAreAnagramsFunction(unittest.TestCase): This class contains unit tests for the are_anagrams function. def test_anagram(self): Test that two anagrams return True. self.assertTrue(are_anagrams("foefet", "toffee")) def test_not_anagram(self): Test that two non-anagrams return False.

Table 5: ICL Examples used for the BGE-en-icl model.

Note: All records share the instruction: "Given an activity, retrieve relevant records that have the same underlying concepts and topics as the given activity"

Retriever Type	Prec _{sys} @ <i>k</i>	Recall@ <i>k</i>	NDCG	Time (s)
bge-base-v1.5	0.208	0.004	0.291	0.048
bge-large-v1.5	0.332	0.004	0.353	0.044
llm-embedder	0.488	0.012	0.537	0.029
bge-en-icl	0.694	0.218	0.684	1.040
bge-reranker	0.664	0.176	0.632	14.030

Table 6: Retriever Performance Comparison: bge-en-icl achieves the highest Prec_{sys}@*k*, Recall@*k* and NDCG@*k* with reasonable latency.

B.2 Comparing Embedding-Models and Retriever Configurations for the Retriever Backbone

We measure the Prec_{sys}@*k* and Recall@*k* on the raw user use-cases from the *topics* set on several widely used dense retriever backbones spanning (i) standard BGE bi-encoders: [BAAI/bge-base-en-v1.5](#) and [BAAI/bge-large-en-v1.5](#) which are popular semantic embedding construction models; (ii) [BAAI/llm-embedder](#), a popular instruction-tuned/LLM-derived embedding model suited for diverse retrieval augmented generation applications especially involving long, abstract queries (Zhang et al., 2024); and (iii) a cross-encoder reranker [BAAI/bge-reranker-base](#), which jointly models query-example interactions rather than independent embeddings. The reranker’s inclusion also allows us to test whether higher compute per query can compensate for the representation gap induced by broad, under-specified practitioner use-cases. Finally, we include the (iv) [BAAI/bge-en-icl](#), an embedding construction model that is designed to better encode instructional intent and task structure using guidance from In-Context Learning examples demonstrating ideal recall. We leverage the latter to diversify our retriever’s recall of diversely formatted examples by including the same in the in-context learning examples we generate for the case. §B.1 outlines the in-context examples used when generating the embeddings for the BAAI/bge-en-icl model. All values reported here at *k* = 20.

Table 6 shows that bge-en-icl provides the best overall trade-off: it achieves the highest system precision (Prec_{sys}@*k* = 0.694), Recall@*k* (0.218), and ranking quality NDCG@*k*, substantially outperforming both generic BGE encoders and the stronger llm-embedder baseline (Prec_{sys}@*k* = 0.488, Recall@*k* = 0.012). Notably, while the bge-reranker is competitive in accuracy (Prec_{sys}@*k* = 0.664, recall=0.176), its latency is an order of magnitude higher (14.03s vs. 1.04s),

making it impractical as a default backbone for the interactive needs for our tool. In contrast, en-icl yields the strongest effectiveness metrics while remaining within a reasonable per-query compute budget, and we therefore adopt bge-en-icl as our default retriever backbone in BENCHBROWSER.

B.3 Filtering selection prompt

SELECTION PROMPT: You are evaluating whether a test case is relevant for assessing an AI model’s ability to help with a specific user task. You are given: - <user_intent> Some topic, a description of some scenario, a capability or an application that some user is interested in <user_intent> - <test_case> Some question/test case that might be used to evaluate that system. </test_case>.

A test case is RELEVANT if correctly answering it would demonstrate capabilities that directly support the user’s intent. Consider the following:

- **Knowledge Overlap:** Does the test require knowledge domains that overlap with what the user needs?
- **Skill Transfer:** Would the reasoning, analysis, or problem-solving skills needed for this test directly apply to the user’s task?
- **Practical Utility** If an AI can handle this test case well, would that increase confidence it can help the user with their specific goal?

Rate the relevance as one of the following:

- 1: Test case is relevant to the user’s intent.
- 0: Test case maybe relevant to the user’s intent but not directly relevant.
- -1: Test case is not relevant to the user’s intent

Provide your rating (only 1,0 or -1) wrapped in the <score> and </score> tags.

C Evaluation Setup and Datasets

C.1 Benchmarks Included in BENCHBROWSER

In Table 7, we include all the benchmarks that are included in BENCHBROWSER’ benchmark

database. In Table 8 we include the metric used by each benchmark. The exact breakdown (all the individual MMLU and Big Bench Hard datasets) along with information about their metrics can also be accessed on the website, under the Convergent Validity tab ('Existing Benchmarks').

C.2 Evaluation Categories

Motivation Skill-based testing is directly aligned with the traditional motivation for constructing LLM evaluation benchmarks, so practitioners directly benefit from being able to use BENCHBROWSER's analysis to anticipate the need to construct a new skill-oriented benchmark. Topics and application-oriented testing is more implicitly relevant to discovering new facets where evaluating LLMs could help discover new strengths and weaknesses. These set of queries can help practitioners decide new contexts to include in future benchmarks which are increasingly focusing on maximizing coverage and topical diversity of real-world user intents.

Topics use-cases. An index is constructed using only BIG-bench Hard (BBH) and MMLU test sets, covering 58 use-cases (tasks and subjects). MMLU college-level subsets are excluded due to topic redundancy (e.g., "Human Aging", "High School Biology", "Clinical Knowledge"). Each topic-based query tests for focused conceptual alignment on a single subject area, enabling controlled measurement of retriever precision on narrow topical domains.

Skills use-cases. To represent practitioner-facing evaluation intents, we derive 50 skill-based use-cases from the Clio analysis of real-world model usage (Tamkin et al., 2024). High-level skills (e.g., "instruction-following", "critical thinking", "creative writing") are prompted to claude-sonnet-4-5-20250929 for naturalistic phrasing. The evaluation index aggregates ~67k examples from 20 modern benchmarks such as IFEval (Zhou et al., 2023), InfoBench (Qin et al., 2024), MMLU, and BBH, enabling retrieval testing over both long-form and MCQ styles. This setup evaluates whether the retriever can generalize across benchmark formats and domains.

Applications use-cases. These 50 Clio-seeded synthetic prompts simulate concrete tasks like "studying for the SAT" or "debugging HP printer errors." They assess how well retrievers locate eval-

uation items relevant to practical scenarios. Precision@20, NDCG@20, and yield are reported per application, using the same 20-benchmark index as the skills setup. Such testing is crucial for understanding real-world generalization and identifying missing evaluation coverage across practical tasks.

Novel use-cases. We circulated a Google Form surveying NLP practitioners (graduate students within AI-focused field) to seek novel use-cases that they were interested in evaluating LLMs on. The participants were allowed to make more than one submission and up to 3 submissions. Each participant was requested to provide at least one peer-reviewed research paper that demonstrates the common utility of the use-case of their interest to the broader scientific community, ensuring the relevance and validity of our evaluation scenarios. This generated a set of 21 use-cases. Some use-cases were processed to omit personal identifiers - for example, 'recommend me food that can satisfy my dietary constraints' was modified to 'recommending food that satisfies dietary restrictions'.

C.3 Prompts for Generating Usecases

We use the Clio (Tamkin et al., 2024) paper to seed the usecases for which we test BENCHBROWSER. This is done to ensure that we don't pick usecases that are too disconnected from real-world needs.

Generating Usecases for the skills set For generating the skill set we use the high-level task categories emerging from a random sample of 1M Claude.ai conversations as described in the paper (Table 7, Appendix F in (Tamkin et al., 2024)).

You are given a list of high-level use case categories for which Claude is used by real-world users. Your task is to identify the underlying fundamental skills that users need when engaging in tasks in these categories.

For each category, think about:

- What are the core cognitive abilities required? (e.g., reasoning, analysis, creativity)
- What technical competencies are involved? (e.g., coding, data manipulation, system design)
- What interpersonal or communication skills are needed? (e.g., persuasion, empathy, cultural awareness)

Benchmark	Citation	Benchmark	Citation
IFEval	Zhou et al. (2023)	logiqa	Liu et al. (2021)
MMLU	Hendrycks et al. (2021)	mathqa	Amini et al. (2019)
TruthfulQA	Lin et al. (2022)	openbookqa	Mihaylov et al. (2018)
HellaSwag	Zellers et al. (2019)	qasper	Dasigi et al. (2021)
ARC	Clark et al. (2018)	triviaqa	Joshi et al. (2017)
Winogrande	Sakaguchi et al. (2020)	webqs	Berant et al. (2013)
BBH	Suzgun et al. (2023)	AlpacaEval	Dubois et al. (2024)
gsm8k	Cobbe et al. (2021)	ComplexBench	Wen et al. (2024)
headqa	Vilares and Gómez-Rodríguez (2019)	Infobench	Qin et al. (2024)
humaneval	Chen et al. (2021)		

Table 7: Benchmarks Used for Example Retrieval

Category	Key Used	Processors / Benchmarks
Custom Metric	<i>Varied</i>	complexbench (point_judges), infobench (DRFR), ifeval (inst_level_strict_acc), llmbar (gold label)
Accuracy	acc	nutribench, truthfulQA, arc, mmlu, winogrande, hellaswag, headqa, logiqa, mathqa, openbookqa, sciq, pubmed, wmdp, moral_stories, big-bench tasks
	exact_match	triviaqa
	pass@1	humaneval
Exact Match	exact_match	bbh, gsm8k, webqs
F1 Score	f1/f1_abstractive	qasper
Win Rate	win	mtbench, koala, vicuna, lfqa, alpacaeval

Table 8: Summary of evaluation metrics used across benchmarks.

- What metacognitive or strategic capabilities are at play? (e.g., planning, decision-making, problem diagnosis)

Generate a list of 50 broad, transferable skills that represent these fundamental capabilities. These skills should be:

- **Atomic:** Each skill should represent a single, measurable capability
- **Cross-domain:** Skills should be applicable across multiple use case categories
- **Evaluable:** Each skill should be something that can be tested or benchmarked
- **Granular enough to be distinctive:** Avoid overly generic terms like "intelligence" or "ability"

Return your answer wrapped in XML tags.

Example: `<skill>cultural adaptation</skill>`

Input - Use Case Categories:

- Web and Mobile App Development
- Content Creation and Communication
- Academic Research and Writing
- Education and Career Development
- Advanced AI/ML Applications
- Business Strategy and Operations
- Language Translation
- DevOps and Cloud Infrastructure
- Digital Marketing and SEO
- Data Analysis and Visualization

Generating Usecases for the *applications* set

For the *applications* set, we use the ground-truth categories from an evaluation dataset of 19,476 synthetic chat transcripts that outline more specific contexts under which Claude is utilized. We ask claude-sonnet-4-5-20250929 to generate applications on the basis of the provided categories using

the following prompt.

You are given a list of broad categories of tasks for which Claude is used by real-world users. Create a list of 50 applications based on the given categories. Do not make the compositional applications i.e., applications that combine too many tasks. Return your answer wrapped in XML tags "usecase" "usecase" Example: "usecase" improving home energy efficiency "usecase" Example: "usecase" recommending Netflix series for me "usecase"

- Software development questions
- Elementary school homework help
- Technology troubleshooting
- Health and fitness advice
- Questions about geopolitics
- Parenting and childcare tips
- Language learning and translation help
- Financial planning and investment
- Theological and philosophical questions
- Environmental science and sustainability
- Book discussions and literary analysis
- Sports rules and strategy questions
- Cooking and recipe inquiries
- Job application questions
- Home improvement and DIY projects
- Pet care and animal behavior
- Romantic relationship advice
- Movie and TV show recommendations
- Music theory and instrument learning
- Tourism and travel questions

C.4 Evaluation Metrics

We use the following metrics for evaluation:

- (a) $\text{MethodPrecision@k}_t = \frac{|\{d_i: i \leq k \wedge S_t(d_i) \in \{1, 0\}\}|}{k}$ - This quantifies the precision of the ranking methods i.e.,

the number of relevant or partially relevant retrieved examples before the relatedness filtering within BENCHBROWSER.

- (b) $\text{SystemPrecision@k}_t = \frac{|\{d_i: i \leq k \wedge L_t(d_i) \in \{\text{REL}, \text{PART}\}\}|}{k}$ - This quantifies the precision of BENCHBROWSER i.e., the number of relevant or partially retrieved examples by BENCHBROWSER after the relatedness filtering.

- (c) $\text{Intersection@k}_t = \frac{\text{hits}_t(k)}{k}$ For the tie-breaking experiment, we also report the Intersection@k_t which helps identify which method recalls the highest number of examples among all the examples returned by all the methods.

- (d) **Recall@k**: In the Bounded Recall setup, measures how many of the returned examples actually belong to the expected gold test set for that use case:

$$\text{Recall@k} = \frac{|\{d \in \mathcal{R}_k \cap \mathcal{G}\}|}{|\mathcal{G}|} \quad (1)$$

where $\mathcal{G}$ is the gold test set for the given use case.

- (e) **Common cutoff**: $K_c = \max\{k : |T_k| \geq 0.9 \cdot |T|\}$
- (f) **Correlation with Model Ranking**: Wherever gold sets are available compute the correlation between the rankings of models when evaluated on a validation set (gold test set) versus our retrieved test set of the same advertised intent.

- (g) **NDCG@k (Normalized Discounted Cumulative Gain)**: Measures the quality of ranking by considering both relevance and position, with higher-ranked relevant documents contributing more to the score:

$$\text{NDCG@k} = \frac{\text{DCG@k}}{\text{IDCG@k}} \quad (2)$$

where DCG@k (Discounted Cumulative Gain) is defined as:

$$\text{DCG@k} = \sum_{i=1}^k \frac{2^{\text{rel}(d_i)} - 1}{\log_2(i + 1)} \quad (3)$$

and IDCG@k (Ideal Discounted Cumulative Gain) is the DCG@k score of the perfect ranking. Here, $\text{rel}(d_i)$ is the relevance score of

document d_i at position i , and the logarithmic discount factor $\log_2(i + 1)$ reduces the contribution of lower-ranked documents.

C.5 Evaluation Prompts

The key distinction between our $\text{Prec}_{\text{sys}}@k$ and $\text{Prec}_{\text{anchor}}@k$ is guided by the distinction between our evaluation prompt used by the LLM Judge for the Filtering Stage (referred to as the Selection Prompt) and the LLM Judge used for the final evaluation. While the $\text{Prec}_{\text{anchor}}@k$ evaluation aims at ascertaining which proportion of examples ranked by any anchoring method (Example Synthesis, Shorthand, etc) are related to the user’s intent, the $\text{Prec}_{\text{sys}}@k$ evaluation prompt is strongly situated in a practitioner’s intent of actually using the retrieved examples for a downstream task. We make this distinction assuming that practitioners might be lenient in claiming relatedness but not necessarily find a testcase useful for their specific intent if the test case is only broadly related to their task.

We corroborate this assumption with annotators during our human evaluation. For most comparisons, we stick to $\text{Prec}_{\text{sys}}@k$, as that is the practically useful metric that practitioners would expect maximized.

EVALUATION PROMPT: ""You are evaluating whether a test case is relevant for assessing an AI model’s ability to help with a specific user task. You are given: - <user_intent> Some topic, a description of some scenario, a capability or an application that some user is interested in </user_intent> - <test_case> Some question/test case that might be used to evaluate that system. </test_case> Your evaluation criteria is: A test case is **RELEVANT** if successfully answering it would demonstrate capabilities that directly support the user’s intent. Consider the following:

- **Knowledge Overlap:** Does the test require knowledge domains that overlap with what the user needs?
- **Skill Transfer:** Would the reasoning, analysis, or problem-solving skills needed for this test directly apply to the user’s task?
- **Practical Utility** If an AI can handle this test case well, would that increase confidence it can help the user with their specific

goal?

Rate the relevance as one of the following:

- **RELEVANT:** If an AI can answer this test case correctly, it would strongly imply that it can help the user with their specific goal.
- **PARTIAL RELEVANT:** If an AI can answer this test case correctly, it would somewhat imply that it can help the user with their broad goal i.e., with the topic or the skill needed by the user.
- **IRRELEVANT:** Answering this question, would not provide any useful indication about if the AI can fulfill the user’s intent.

Provide your rating and wrapped in the <label> and </label> tags.

D Additional Evaluation of Anchor Generation Methods for Retrieval

D.1 Tie-breaking: Computing Correlation between Model Performances on Retrieved Set and Gold Test Set for *topics* set

Measuring the correlation between the performance of a model on a gold test set that is explicitly designed for a use-case with the performance of the model on the set of retrieved examples can be a strong sanity check on the utility of the retrieved examples. Accordingly, for the *topics* set, we compute the performance of $N = 8$ models on the 58 BBH and MMLU testsets and then compute their correlation with the performance of the models on the sets of retrieved examples for the same use-cases. We use the following models for our evaluation: We use the following eight models for our evaluation: Llama-3.1-8B-Instruct, Llama-3.1-Tulu-3-8B, Mistral-7B-Instruct-v0.2, Qwen1.5-14B-Chat, Qwen2.5-7B-Instruct, c4ai-command-r7b-12-2024, gemma-2-9b-it, and gemma-3-4b-it. Since all the benchmarks in this set use either accuracy or exact string match as their metric for reporting accuracy, we aggregate the per-sample accuracy or exact-string match ($\in [0, 1]$) and then normalize by the number of retrieved examples to obtain each model’s performance on the retrieved set. As a baseline, we also plot the correlation between the performance

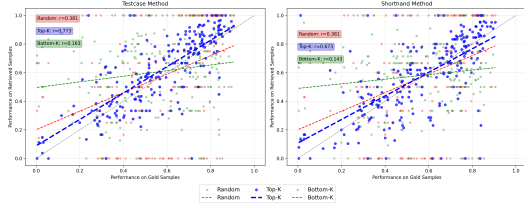


Figure 5: We observe that performance on the retrieved testset and the gold test set (BBH and MMLU for the *topics* set set) show reasonable to high correlation.

of the models on the bottom- k and a random slice of the benchmark index as a way of calibrating random correlation.

Figure 5 demonstrates that the performance of the models on the retrieved sets and gold sets have high correlations: highest for the Example Synthesis anchor method.

D.2 Tie-breaking: Estimating Overlap with Approximated Gold Set for *applications* set and *skills* set

Use-cases in the *application* and *skill* sets do not have pre-defined gold sets of human-curated examples. Therefore, we cannot calculate the recall@ k metric for these use-cases. Instead, we compare the different retrieval methods using a metric we call *intersection@ k* . This metric measures the overlap between a method’s relevant examples and the full set of examples that are deemed relevant by any method. To do this, we first run each retrieval method for a relatively large value of $k = 80$ and then take the union of all the examples that are judged as relevant by any method. Then, for a method m with ranked list $R_m(q) = [d_1, d_2, \dots]$, we compute the intersection of the examples up to rank k retrieved by BENCHBROWSER with the union set and divide by k .

Figure 6 shows that the *testcase* method consistently has higher intersection@ k values across all three sets of use-cases.

To investigate the cause of the spike observed at the tail end of the intersection (Figure 6), we compute the support fraction across k which denotes what fraction of the queries actually contribute to the precision at a particular value of k . Formally, for method m , the *support* is $s_m(k) = |\{q : |G(q)| > 0 \wedge |R_m(q)| \geq k\}|$, and the support fraction is $\hat{s}_m(k) = s_m(k)/Q$. We define $K_{\text{common}}(m) = \max\{k : \hat{s}_m(k) \geq 0.9\}$, so points beyond K_{common} are backed by fewer than 90% of queries. In Figure 6: the green area hence marks

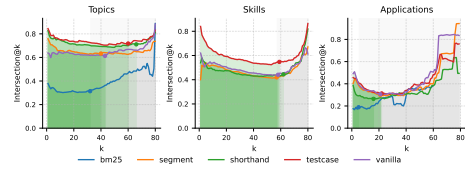


Figure 6: intersection@ k in approximated recall setup: Gray region indicates precision for fewer than 90% of queries. For each method we compute K_{common} : the largest k for which $\geq 90\%$ of tasks contribute a value at every position $\leq k$ (i.e., their lists are at least that long). The area up to K_{common} under that method’s curve is filled green (trusted, high task support); the tail beyond K_{common} is filled grey (lower support). A dot marks (K_{common} , Precision at K common.) Here: *segment* denotes Rephrasing, *shorthand* denotes Shorthand, *testcase* denotes Example Synthesis and *vanilla* indicates Original.

the common area versus the greyed out region indicates the region where intersection@ k is contributed by less than 10% of the queries; Through this we yield a more stable comparison between the macro averages at large k by avoiding dominance by a small, unrepresentative subset of easy queries where achieving high precision is naively possible. Ultimately, this experiment further provides evidence of Example Synthesis’ reasonable proficiency and generality in being able to (a) retrieve relevant examples and (b) retrieve the largest set of relevant examples for any use-case.

D.3 known validation set use-cases and their corresponding Gold Test Sets

Table 9 specifies the use-cases designed for the *known validation* set.

D.4 Precision of BENCHBROWSER as K is increased

Figure 7 shows $\text{Prec}_{\text{sys}}@k$ for the best-performing anchor (Example Synthesis) as we vary the number of retrieved examples, k . For *topics* set and *skills* set, precision remains relatively stable as k grows, indicating that BENCHBROWSER can return larger pools of examples without rapidly flooding practitioners with irrelevant items. This may be especially useful when practitioners want to sample a broader, larger set of examples for using doing fine-grained selection over or using as an alternate test set for custom use-cases for which gold test sets don’t exist.

In both experiments, performance on the *applications* set set is noticeably lower across all methods

Usecase	Gold Test Set
Learn how to measure my carbohydrate intake and make my meals more balanced	nutribench (Hua et al., 2024)
Test my knowledge of different sports and their respective players	bb_sports
Help me learn more about the Hindu religion	bb_hindu
Test me on different topics in science	sciq (Welbl et al., 2017)
Help me answer questions for my biomedical research	pubmed (Jin et al., 2019)
Help me mitigate different biological, cyber, and chemical attacks	wmdp (Li et al., 2024)
Explain the ethically right choice to make under different kind of situations	moral_stories (Emelin et al., 2021)
Can help me write one-line python codes	bb_code_line
Helps me find common substrings between strings	bb_cs_algo
Assists me in fact checking my claims	bb_fact_checker
Tests me on the properties of human organs	bb_human_organs
Teaches me how to use metaphors in daily language	bb_identify_odd
Learn how to identify common misconceptions	bb_misconceptions
Helps me resolve moral dilemmas	bb_moral_permissibility
Teaches me how to identify the underlying concept or theme among a few things	bb_novel_concepts
Can solve riddles for me	bb_riddle_sense
Can help me understand the social implications of my actions	bb_social_iqa
Help me figure out if someone supports my argument	bb_social_support
Tests my emotional intelligence	bb_strange_stories

Table 9: Prompts with Corresponding Gold Datasets used for the *Known Validation* Set. All datasets prefixed with “bb” are included in the Big Bench Dataset (Srivastava et al., 2023).

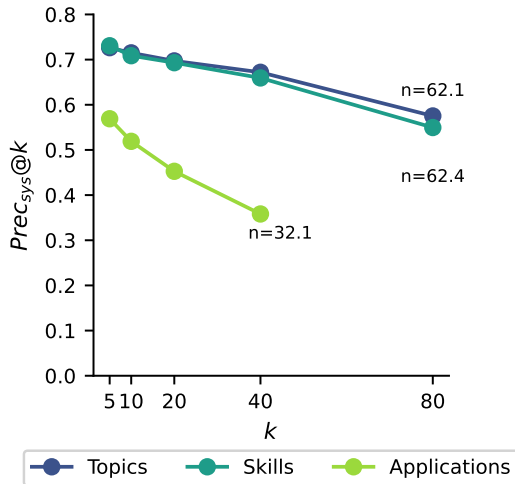


Figure 7: System precision ($\text{Prec}_{\text{sys}}@k$) as a function of k for *topics* set, *skills* set, and *applications* set. For *topics* set and *skills* set, precision remains high as k increases, while *applications* set exhibits a sharper decline.

(Table 1, Figure 7), reflecting the difficulty of retrieving examples for highly contextual use-cases from existing benchmarks. This drop in precision is not surprising since *applications* set use-cases often bundle multiple skills and entities (e.g., domain, user role, format, constraints), only some of which may be followed or *available* in the examples included in the benchmarks. Despite this, Example Synthesis has high performance in this set too where the non-rewrite based Original baseline actually becomes the most competitive retrieval

method. In alignment with prior work, this parity between the Original and Example Synthesis based anchoring highlights the complexity in devising and choosing a single rewrite method that can maintain retrieval performance across a wide variety of practitioner use-cases (Kalra et al., 2025; Salemi et al., 2024).

E Automatic Judge Calibration with Another Automatic Judge

To further validate the sanity of our automatic relevance judgments we also replicate the automatic precision computation in Table 1 with another automatic judge i.e., gemini-2.5-flash-lite and report the spearman correlation between the scores enlisted by this model and our designated judge, gpt-5-mini in Table 10.

Category	Score	$\text{Prec}_{\text{sys}}@k$	$\text{Prec}_{\text{sys}}@k$
Overall	0.629 [‡]	0.700 [‡]	0.795 [‡]
<i>applications</i> set	0.613 [‡]	0.738 [‡]	0.725 [‡]
<i>known validation</i> set	0.549 [‡]	0.485 [‡]	0.604 [‡]
Skills	0.603 [‡]	0.626 [‡]	0.747 [‡]
Topics	0.576 [‡]	0.562 [‡]	0.622 [‡]

Table 10: Spearman correlation coefficients at $k=10$ and $k=20$ respectively. First $\text{Prec}_{\text{sys}}@k$ column corresponds to $k=10$ and the 2nd one corresponds to $k=20$. [‡] indicates $p < 0.001$; [†] indicates $p < 0.05$.

The resulting Spearman correlations are consistently positive and statistically significant (Ta-

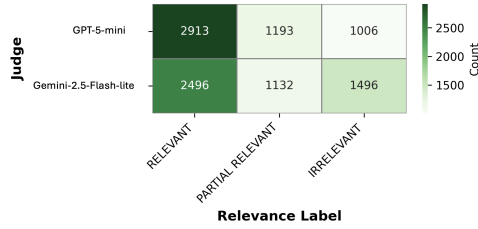


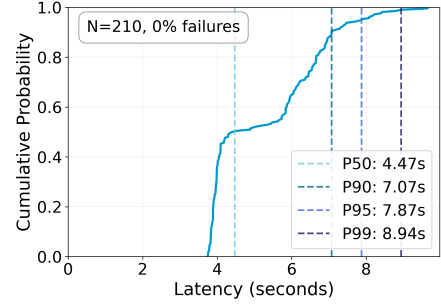
Figure 8: Marginal relevance-label distributions assigned by GPT-5-mini (designated judge) and Gemini-2.5-flash-lite over the same evaluated top- k retrieved items.

ble 10), indicating that items judged as more relevant by our designated judge also tend to be judged as more relevant by the alternative judge. This provides evidence that the relative relevance ordering induced by our automatic evaluation is stable under judge substitution, supporting the robustness of our aggregate retrieval comparisons. Figure 8 also reports the marginal label distributions assigned by each judge across all evaluated items. Both judges use all three labels and share the same qualitative pattern—RELEVANT is most frequent, followed by PARTIAL RELEVANT but they differ in strictness: gpt-5-mini assigns RELEVANT more often ($\approx 57\%$) and gemini-2.5-flash-lite assigns fewer RELEVANT labels ($\approx 49\%$). Together with the strong item-level rank agreement, this suggests that the judges largely agree on which items are comparatively more vs. less relevant, while disagreeing primarily in the decision boundary between RELEVANT and IRRELEVANT (i.e., Gemini is more conservative). This pattern is consistent with *reasonable calibration* for our use case as aggregate precision trends seem stable (at least across method) under an alternate judge.

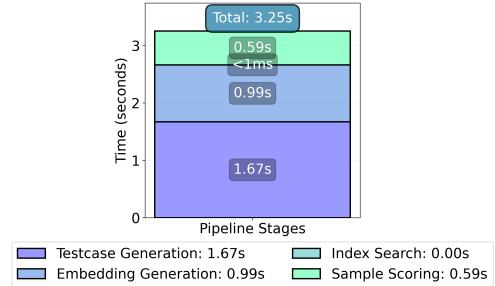
F System Infrastructure and Profiling

F.1 Hosting Infrastructure

BENCHBROWSER is deployed as a Google Cloud Run service in region northamerica-northeast1, provisioned with **4 GiB RAM** and **1 vCPU**. To reduce cost, we set the minimum number of instances to **0**, so requests may incur cold-start overhead; each request has a **120s** deadline (requests exceeding this limit are terminated). Cloud Run request concurrency is set to 80. To avoid GPU-backed embedding computation, we outsource embeddings to the [DeepInfra](#) API, using BAAI/bge-en-icl at a rate of **\$0.0100/Mtoken**. Our reported end-to-end



(a) End-to-end latency distribution.



(b) Mean stage-time breakdown.

Figure 9: **Latency of the BENCHBROWSER web demo.** (top) Empirical CDF of request latency over 210 runs of the public BENCHBROWSER demo ($k=35$), with percentile markers (p50/p90/p95/p99). Median latency is 4.47s; tail latency remains below 8.95s at p99 (0 failures). (Right) Average time spent in each pipeline stage, showing testcase generation (1.67s; 31.7%), embedding generation (0.99s; 18.8%), and sample scoring (0.59s; 11.2%) as the dominant contributors; the remaining time corresponds to end-to-end overhead.

latency includes the external embedding API call (network + inference) in addition to backend processing but **does not** include the time to warm start the container (which can be anywhere between 60 to 180 seconds depending upon cloud resource availability as well as asset load time).

F.2 Latency Profiling

We report end-to-end latency for the public BENCHBROWSER demo over 210 requests (1 iteration per query; no failures) in Figure 9. Latency is 5.28 ± 1.48 s (mean $\pm$ stdev), with p50 = 4.47 s, p90 = 7.08 s, p95 = 7.90 s, and p99 = 8.95 s (min 3.75 s, max 9.68 s). A stage-level breakdown attributes 1.67 s (31.7%) to testcase generation, 0.99 s (18.8%) to embedding generation, and 0.59 s (11.2%) to sample scoring; the remaining ~ 2.02 s ($\sim 38\%$) is unattributed overhead (primarily request/response handling, serialization, networking, and other pipeline glue).

G Human Evaluation of BENCHBROWSER

We obtained an Institutional Review Board (IRB) exemption under the 2018 Common Rule (45 CFR § 46.104(d)) to conduct our study with $N = 10$ annotators. Annotators were compensated with a DoorDash food order (up to \$20 USD) for approximately one hour of participation. Eligibility criteria required that participants be at least 18 years old, fluent English speakers, currently residing in the United States, and have prior experience working on at least one NLP project.

In Figure 10 we include the instructions provided to the annotators and provide a screenshot of the interface used by annotators to provide relevance judgments for the retrieved samples (Figure 11).

G.1 Detailed Analysis of Human–Automatic Disagreement

Known Validation outliers. A closer analysis of the Known Validation set reveals that the large mean difference between human and automatic relevance scores (Table 2) is driven by six use-cases with unusually large per-use-case deltas (0.40–0.75), with humans consistently assigning lower relevance. These use-cases are drawn from benchmarks that test abstract or esoteric capabilities, making it difficult to “retrofit” a concise practitioner-style use-case. For instance, the most extreme deviation (mean difference 0.75) arises from BIG-bench’s `novel_concepts` task (Srivastava et al., 2023), whose original goal is to “*measure the ability of models to uncover an underlying concept that unites several ostensibly disparate entities, which hopefully would not co-occur frequently.*” We simplify this to a retrieval use-case of “*identifying the underlying concept for a given set of items.*” Annotator comments (Table 11) indicate that, without the full benchmark context, this simplified use-case encourages annotators to form narrow, idiosyncratic interpretations. Examples that are consistent with the original benchmark designer’s intent, but not with the annotator’s inferred interpretation, are then marked as *Irrelevant*, increasing the human–automatic gap. We observe similar issues for other abstract tasks in this set, suggesting that retrofitting highly specialized benchmarks into short use-cases can systematically bias human judgments downward.

Inter-annotator agreement. We also investigate the relatively low Fleiss’s κ for several sets (Ta-

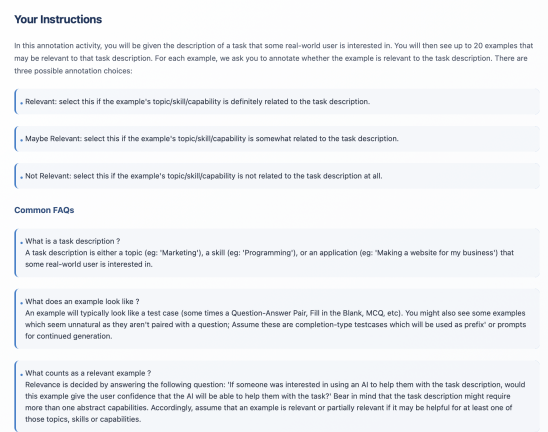


Figure 10: Interface showing the annotation instructions to the annotators.

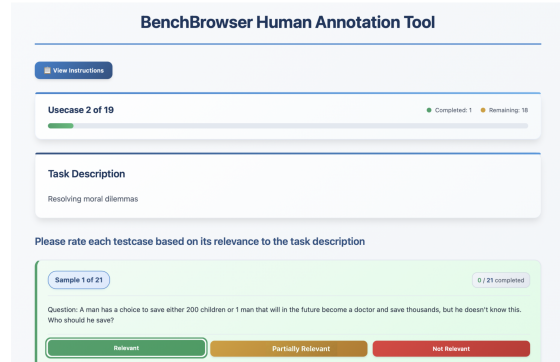


Figure 11: Annotation Interface for BENCHBROWSER

ble 2). Figure 12 plots confusion matrices for (a) the LLM judge versus the majority human label and (b) pairs of human annotators. In both cases, most disagreements occur between *Relevant* and *Partially Relevant* relevant labels rather than between *Relevant* and *Irrelevant*, indicating that annotators and the LLM generally agree on whether an item is “in the ballpark” and differ mainly on how strict to be. Table 11 illustrates this with example comments from annotators on disputed items: both annotators often articulate plausible but different interpretations of what counts as “relevant” to a given use-case (e.g., whether sentiment classification suffices for “detecting emotional support in conversations”). This pattern matches prior findings that inter-annotator agreement is modest for inherently subjective constructs such as quality, relevance, or interest (Adams et al., 2023; McDonnell et al., 2016; Shirani et al., 2021), and suggests that our IAA values mainly reflect the intrinsic ambiguity of the task rather than poorly specified instructions.

Table 11: Comments from Annotators over samples where we observe disagreement: Annotator A is always the annotator assigning a high relevance (1) and Annotator B is always the annotator assigning a low relevance (0).

Sample	Annotator Comment
use-case: Learn Differential Equations	
Prompt: Antiderivar la siguiente función $f(x)=(x+1)(2x-1)$	A says: “solving DEs requires computing antiderivatives;” B says: “The question is testing both translation and maths; within maths, differential equation is not explicitly tested so on average I this isn’t relevant”.
use-case: Detecting emotional support in conversations	
Group chat excerpt: Owner: @Team - Aren’t you too unpunctual? We agreed on 10 o’clock yesterday Prompt: Determine the emotion of the owner’s last message. Choose from [negative, neutral, positive].	A says : The task is sentiment classification which is relevant to detecting emotions. B says: The task description is specifically asking for cases that test emotional support, while the question tests emotion classification with predefined categories which aren’t relevant to emotional support.
use-case: Identifying the underlying concept for a given set of items	
All of the following support the endosymbiotic theory <i>EXCEPT</i> - Mitochondria and chloroplasts divide ... - Mitochondria and chloroplasts have ribosomes	Annotator B: Given the broad nature of "concepts, factual knowledge," technically anything in a domain could share underlying concepts. This seems too abstract and insufficiently related to the specific task so I can consider this partially relevant at best". <i>Annotator A marks as 'relevant' but does not justify in the comments.</i>

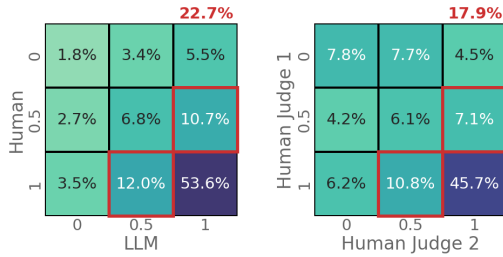


Figure 12: Intra and Inter-Label distributions between LLM and human annotators on the examples evaluated from all the sets. Most disagreements occur between *partially relevant* and *relevant* categories rather than polar opposites, implying calibration rather than conceptual divergence.

H Additional Details for Content Validity

We manually verify that the 120 use-cases for demonstrating the utility of BENCHBROWSER in §5 are constructed such that induced variations are lexically distinct. Assuming a very rigid relevance criterion of which deems that “a example is relevant only if it explicitly states the induced variation” (following which it is already checked for semantic relevance at the filtering stage) - even a lexical baseline is expected to give similar trends in the variation of the spread of relevant examples across variations of the same usecase.

Accordingly, we also construct a lexical BM25 baseline which only retrieves an example if it explicitly mentions the variation’s keyword. Though stringent, this has been shown to be a realistic base-

line of an annotator judging relevance for use-cases that they are either not interested in or are not experts in (Voorhees, 1998; Saracevic, 2007). Figure 13 confirms that the qualitative trends from Figure 3 hold under this lexical baseline, suggesting that our findings are due to benchmark composition rather than an artifact of the semantic retriever.

We confirm that the lexical baseline also demonstrate the same trend in spread of relevance, reiterating that the variation in relevant count spread observed is an artifact of the properties of the use-case and not due to lossy semantic retrieval. Ultimately, this confirms that some facets are underserved compared to other dominant facets.

H.1 Example of Use-Cases showing Spread in Relevant Count across Variations

Table 13 (placed at the end of the appendix to avoid hindering readability of existing content) includes examples of the task descriptions and consequent yields of examples across variations along the same axes.

I Additional Details for Convergent Validity

We use the following eight models for our evaluation: Llama-3.1-8B-Instruct, Llama-3.1-Tulu-3-8B, Mistral-7B-Instruct-v0.2, Qwen1.5-14B-Chat, Qwen2.5-7B-Instruct,

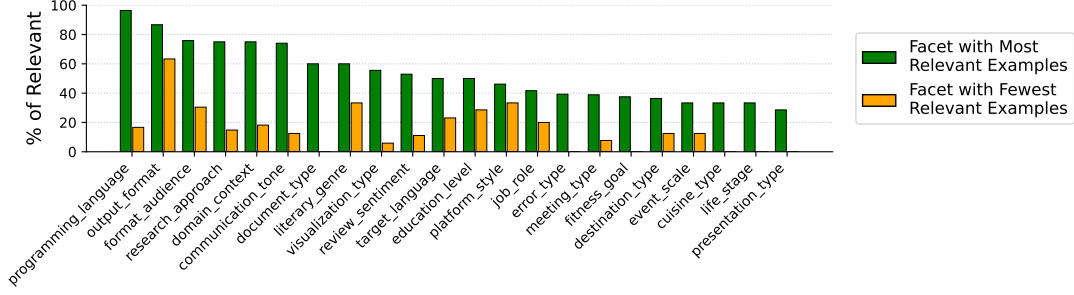


Figure 13: BM25 retrieval results for calibrating the variation in content validity as reported by BENCHBROWSER

c4ai-command-r7b-12-2024, gemma-2-9b-it, and gemma-3-4b-it.

Accuracy Aggregation for Model Performance

Estimates Model performance is evaluated using the specific metric designated by each benchmark, as summarized in Table 8. The majority of benchmarks supported by the Eval Harness (Gao et al., 2024a) provide per-sample accuracy based on strict string matching or binary accuracy. We utilize these values directly, noting that they are bounded between 0 and 1 (representing a binary exact match or correct prediction). For benchmarks employing Win-Rate as a metric, we generate model inferences and compute the win-rate against a fixed set of standard responses generated by the Ministral-8B-Instruct-2410 model. In this comparison, a win against the reference model is encoded as 1, while a tie or loss is encoded as 0 (serving as an accuracy equivalent). For the remaining benchmarks requiring custom evaluation protocols, we employ the specific metrics and libraries provided by the original authors. For instance, InfoBench (Qin et al., 2024) is evaluated using the provided code for its designated DRFR metric. These per-sample scores are subsequently aggregated to derive the final model performance estimates.

To control for any variance in retrieval results induced due to difference in prompting language, we write 5 task descriptions per use-case and retrieve examples for each. Then we pick the prompt that has the high automatic precision and have those annotated by human annotators. To account for the change the Kendall’s τ due to the size of the retrieved set and measure the internal heterogeneity of the gold set, we compare the Kendall’s τ across multiple configurations, described in detail below.

1.1 Contextualizing Kendall’s Tau

For each task, we select a single retrieval prompt shared across all models: the prompt with the highest automatic retrieval precision. We then evaluate all models on the same retrieved subset for that task. Let a_m^{ret} denote model m ’s mean accuracy on its retrieved items (with n_m retrieved items), and let $\bar{a}_m^{\text{gold}}$ denote its mean accuracy on the full gold evaluation set. We compare the following Kendall’s τ correlations per task (averaged over $S = 100$ Monte Carlo trials) in order to account for the fact that the retrieved sets are often much smaller than the gold sets:

- (i) **Same-cardinality retrieved–gold: τ_{ret} .** We compare the retrieved ranking $\{a_m^{\text{ret}}\}_m$ to a gold ranking estimated at *matched evaluation budget*. In each trial s , for every model m we subsample n_m examples from its gold pool, compute the subsampled gold accuracy $a_m^{\text{gold},(s)}$, and compute

$$\tau_{\text{ret}}^{(s)} = \tau\left(\{a_m^{\text{ret}}\}_m, \{a_m^{\text{gold},(s)}\}_m\right).$$

τ_{ret} is meant to ascertain if the retrieved set induces the same model ordering as the gold set, if both were evaluated with the same number of examples.

- (ii) **Internal gold stability (full vs. subset): τ_{gold} .** To quantify how much rankings can fluctuate due to *finite-sample effects alone*, we measure the stability of gold rankings when reducing the gold set’s size. In each trial s we subsample n_m gold examples per model, compute $a_m^{\text{gold},(s)}$, and compute

$$\tau_{\text{gold}}^{(s)} = \tau\left(\{\bar{a}_m^{\text{gold}}\}_m, \{a_m^{\text{gold},(s)}\}_m\right).$$

Thus, τ_{gold} provides a task-specific ceiling on rank agreement attributable to reduced evaluation budget.

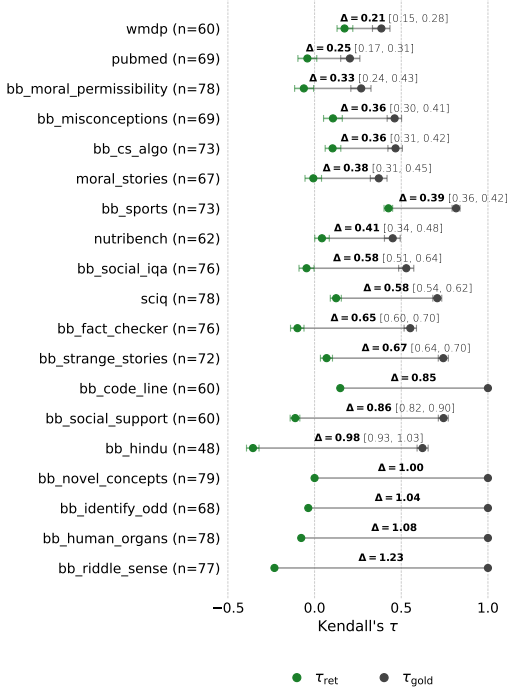


Figure 14: τ_{gold} shows how model rank correlation decreases when selecting a subset of the gold evaluation set; τ_{ret} measures additional divergence introduced by retrieval. n is the number of retrieved examples for each task.

In each trial s , we compute $\Delta^{(s)} = \tau_{\text{gold}}^{(s)} - \tau_{\text{ret}}^{(s)}$. Means of Δ with 95% bootstrap confidence intervals are in Figure 14. If a task has more retrieved examples than its gold set, no subsetting occurs so τ_{gold} is 1 and error bars are not meaningful.

Fully same-cardinality sets. As an additional control, we also report Kendall’s τ correlations in a setting with completely matched evaluation budgets for all sets that are compared. For each task, let n_{gold} be the number of examples in the gold evaluation set, and let $k = \min(\frac{1}{2}n_{\text{gold}}, n_{\text{ret}})$. Then for each model, we recompute the previous quantities where all sets have cardinality k . To start with, we sample a uniformly random subset of the gold evaluation set of size n_{gold} and split it into two subsets, each of size k .

- (i) τ'_{ret} . We subsample k examples from the retrieved pool and use the first set of k examples from the gold pool to compute $\tau'_{\text{ret}}^{(s)}$ in the same manner as τ_{ret} above.
- (ii) **Gold–gold repeatability (subset vs. subset): τ'_{sanity} .** We use both of the size- k subsets of the gold set, compute the correspond-

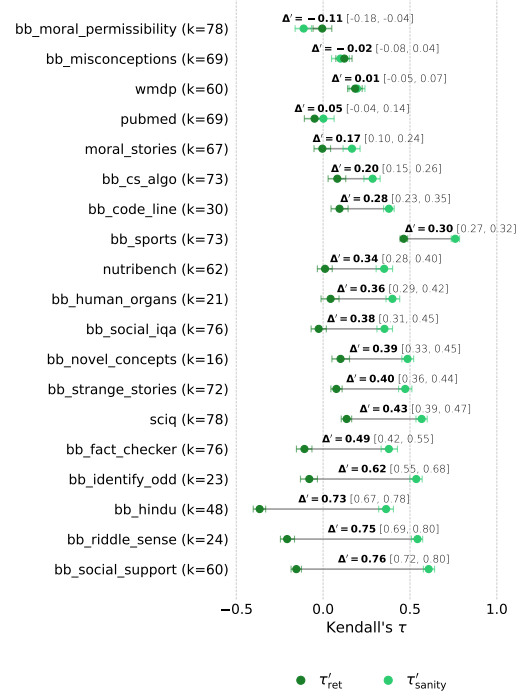


Figure 15: τ'_{sanity} shows how model rank correlation decreases when selecting a subset of the gold evaluation set; τ'_{ret} measures additional divergence introduced by retrieval. Here all evaluation sets used for model rankings have the same cardinality k in a given task.

ing accuracies, and compute

$$\tau'_{\text{sanity}}^{(s)} = \tau\left(\{a_m^{\text{gold},(s_1)}\}_m, \{a_m^{\text{gold},(s_2)}\}_m\right).$$

This isolates the intrinsic uncertainty in rankings when the evaluation budget is smaller (without referencing the full gold ranking).

In each trial s , we compute $\Delta'^{(s)} = \tau'_{\text{sanity}}^{(s)} - \tau'_{\text{ret}}^{(s)}$. Means of Δ' with 95% bootstrap confidence intervals are in Figure 15.

Results. Figures 14 and 15 plot the means of these various quantities for all tasks. Table 12 at the end of the appendix gives other summary statistics for the distributions. τ_{gold} and τ_{sanity} quantify how reliable model ordering is even within the original benchmark at the relevant sample size. This helps us confirm that the low τ_{ret} we observe is not due to inherent heterogeneity in the gold set. However, τ_{ret} is often much lower (sometimes near-zero or negative), which we posit is indicative of divergent operationalizations and rubric mismatches in the retrieved sets vs the original gold evaluation sets. Even when comparing against the more stringent τ'_{sanity} , we see that τ'_{ret} is lower, indicating low convergent validity.

Task	Quantity	Mean	Std	p10	p25	p50	p75	p90	Quantity	k	Mean	Std	p10	p25	p50	p75	p90
wmdp	τ_{ret}	0.17	0.23	-0.11	0.00	0.17	0.36	0.45	τ'_{ret}	60	0.18	0.22	-0.11	0.04	0.18	0.36	0.45
pubmed	τ_{ret}	-0.04	0.27	-0.44	-0.19	-0.04	0.11	0.25	τ'_{ret}	69	-0.05	0.29	-0.40	-0.25	-0.07	0.11	0.33
bb_moral_permissibility	τ_{ret}	-0.06	0.28	-0.40	-0.25	-0.07	0.11	0.23	τ'_{ret}	78	-0.01	0.28	-0.40	-0.18	0.04	0.14	0.36
bb_misconceptions	τ_{ret}	0.11	0.26	-0.21	-0.07	0.14	0.29	0.43	τ'_{ret}	69	0.12	0.24	-0.18	-0.01	0.14	0.29	0.43
bb_cs_algo	τ_{ret}	0.10	0.23	-0.23	-0.04	0.11	0.29	0.40	τ'_{ret}	73	0.08	0.27	-0.26	-0.11	0.08	0.29	0.46
moral_stories	τ_{ret}	-0.01	0.25	-0.37	-0.18	0.00	0.11	0.33	τ'_{ret}	67	-0.00	0.25	-0.29	-0.18	0.00	0.14	0.29
bb_sports	τ_{ret}	0.43	0.12	0.29	0.36	0.43	0.50	0.57	τ'_{ret}	73	0.46	0.11	0.29	0.40	0.47	0.55	0.62
nutribench	τ_{ret}	0.04	0.22	-0.25	-0.11	0.04	0.21	0.33	τ'_{ret}	62	0.01	0.21	-0.29	-0.16	0.04	0.12	0.27
bb_social_iqa	τ_{ret}	-0.05	0.22	-0.33	-0.19	-0.04	0.11	0.21	τ'_{ret}	76	-0.03	0.23	-0.33	-0.18	-0.04	0.11	0.33
sciq	τ_{ret}	0.12	0.16	-0.08	0.03	0.11	0.21	0.33	τ'_{ret}	78	0.14	0.15	-0.08	0.04	0.15	0.22	0.33
bb_fact_checker	τ_{ret}	-0.10	0.20	-0.36	-0.22	-0.13	0.01	0.15	τ'_{ret}	76	-0.11	0.22	-0.36	-0.22	-0.13	0.01	0.14
bb_strange_stories	τ_{ret}	0.07	0.18	-0.19	-0.04	0.07	0.21	0.29	τ'_{ret}	72	0.08	0.16	-0.11	-0.01	0.11	0.21	0.26
bb_code_line	τ_{ret}	0.15	0.00	0.15	0.15	0.15	0.15	0.15	τ'_{ret}	30	0.09	0.25	-0.20	-0.07	0.07	0.30	0.44
bb_social_support	τ_{ret}	-0.11	0.14	-0.29	-0.21	-0.11	0.00	0.07	τ'_{ret}	60	-0.15	0.15	-0.36	-0.29	-0.14	-0.07	0.01
bb_hindu	τ_{ret}	-0.36	0.19	-0.57	-0.47	-0.36	-0.21	-0.11	τ'_{ret}	48	-0.37	0.19	-0.57	-0.47	-0.36	-0.25	-0.18
bb_novel_concepts	τ_{ret}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	τ'_{ret}	16	0.10	0.24	-0.21	-0.08	0.08	0.25	0.39
bb_identify_odd	τ_{ret}	-0.04	0.00	-0.04	-0.04	-0.04	-0.04	-0.04	τ'_{ret}	23	-0.08	0.25	-0.39	-0.24	-0.11	0.07	0.25
bb_human_organisms	τ_{ret}	-0.08	0.00	-0.08	-0.08	-0.08	-0.08	-0.08	τ'_{ret}	21	0.04	0.27	-0.31	-0.16	0.08	0.23	0.38
bb_riddle_sense	τ_{ret}	-0.23	0.00	-0.23	-0.23	-0.23	-0.23	-0.23	τ'_{ret}	24	-0.21	0.23	-0.52	-0.35	-0.22	0.00	0.08
wmdp	τ_{gold}	0.39	0.25	0.04	0.22	0.42	0.57	0.69	τ'_{sanity}	60	0.20	0.24	-0.11	0.04	0.18	0.36	0.49
pubmed	τ_{gold}	0.20	0.28	-0.19	0.04	0.23	0.41	0.54	τ'_{sanity}	69	0.00	0.31	-0.40	-0.22	0.00	0.19	0.44
bb_moral_permissibility	τ_{gold}	0.27	0.31	-0.16	0.04	0.28	0.52	0.69	τ'_{sanity}	78	-0.11	0.23	-0.43	-0.29	-0.11	0.04	0.19
bb_misconceptions	τ_{gold}	0.46	0.21	0.14	0.36	0.50	0.57	0.71	τ'_{sanity}	69	0.10	0.26	-0.25	-0.12	0.11	0.29	0.43
bb_cs_algo	τ_{gold}	0.47	0.20	0.15	0.36	0.50	0.62	0.71	τ'_{sanity}	73	0.28	0.23	0.00	0.15	0.29	0.44	0.56
moral_stories	τ_{gold}	0.37	0.24	0.04	0.21	0.36	0.55	0.67	τ'_{sanity}	67	0.17	0.26	-0.12	0.04	0.15	0.33	0.45
bb_sports	τ_{gold}	0.82	0.12	0.64	0.71	0.84	0.91	0.97	τ'_{sanity}	73	0.76	0.12	0.62	0.69	0.77	0.85	0.91
nutribench	τ_{gold}	0.45	0.23	0.18	0.33	0.46	0.62	0.71	τ'_{sanity}	62	0.35	0.24	0.07	0.18	0.33	0.49	0.65
bb_social_iqa	τ_{gold}	0.53	0.22	0.21	0.40	0.56	0.69	0.77	τ'_{sanity}	76	0.35	0.23	0.04	0.21	0.37	0.50	0.65
sciq	τ_{gold}	0.71	0.13	0.55	0.62	0.71	0.79	0.89	τ'_{sanity}	78	0.57	0.17	0.37	0.47	0.58	0.67	0.77
bb_fact_checker	τ_{gold}	0.55	0.18	0.36	0.43	0.57	0.64	0.72	τ'_{sanity}	76	0.38	0.24	0.11	0.21	0.36	0.57	0.65
bb_strange_stories	τ_{gold}	0.74	0.15	0.55	0.67	0.76	0.84	0.91	τ'_{sanity}	72	0.47	0.20	0.21	0.34	0.48	0.63	0.74
bb_code_line	τ_{gold}	1.00	0.00	1.00	1.00	1.00	1.00	1.00	τ'_{sanity}	30	0.38	0.16	0.18	0.29	0.40	0.49	0.59
bb_social_support	τ_{gold}	0.74	0.14	0.57	0.64	0.76	0.86	0.93	τ'_{sanity}	60	0.61	0.16	0.40	0.50	0.64	0.71	0.79
bb_hindu	τ_{gold}	0.62	0.17	0.40	0.50	0.62	0.75	0.86	τ'_{sanity}	48	0.36	0.22	0.07	0.25	0.37	0.49	0.65
bb_novel_concepts	τ_{gold}	1.00	0.00	1.00	1.00	1.00	1.00	1.00	τ'_{sanity}	16	0.49	0.18	0.24	0.34	0.49	0.62	0.72
bb_identify_odd	τ_{gold}	1.00	0.00	1.00	1.00	1.00	1.00	1.00	τ'_{sanity}	23	0.54	0.18	0.30	0.40	0.54	0.67	0.75
bb_human_organisms	τ_{gold}	1.00	0.00	1.00	1.00	1.00	1.00	1.00	τ'_{sanity}	21	0.40	0.21	0.15	0.24	0.38	0.55	0.69
bb_riddle_sense	τ_{gold}	1.00	0.00	1.00	1.00	1.00	1.00	1.00	τ'_{sanity}	24	0.54	0.17	0.33	0.43	0.55	0.69	0.74
wmdp	Δ	0.21	0.34	-0.22	-0.07	0.21	0.50	0.67	Δ'	60	0.01	0.32	-0.36	-0.21	0.00	0.19	0.42
pubmed	Δ	0.25	0.37	-0.30	0.05	0.27	0.49	0.73	Δ'	69	0.05	0.46	-0.56	-0.26	0.04	0.40	0.60
bb_moral_permissibility	Δ	0.33	0.48	-0.33	0.02	0.40	0.76	0.90	Δ'	78	-0.11	0.37	-0.55	-0.36	-0.07	0.14	0.32
bb_misconceptions	Δ	0.36	0.29	-0.01	0.14	0.43	0.57	0.71	Δ'	69	-0.02	0.32	-0.37	-0.21	0.00	0.19	0.36
bb_cs_algo	Δ	0.36	0.27	0.00	0.22	0.38	0.57	0.67	Δ'	73	0.20	0.32	-0.18	0.00	0.25	0.41	0.55
moral_stories	Δ	0.38	0.35	-0.09	0.20	0.37	0.59	0.81	Δ'	67	0.17	0.35	-0.32	-0.02	0.24	0.38	0.60
bb_sports	Δ	0.39	0.16	0.15	0.29	0.43	0.52	0.57	Δ'	73	0.30	0.14	0.11	0.21	0.29	0.36	0.49
nutribench	Δ	0.41	0.35	0.00	0.15	0.38	0.64	0.87	Δ'	62	0.34	0.33	-0.08	0.12	0.37	0.57	0.70
bb_social_iqa	Δ	0.58	0.34	0.14	0.30	0.62	0.83	1.00	Δ'	76	0.38	0.38	-0.11	0.08	0.41	0.69	0.86
sciq	Δ	0.58	0.23	0.29	0.43	0.59	0.73	0.86	Δ'	78	0.43	0.22	0.19	0.30	0.43	0.58	0.71
bb_fact_checker	Δ	0.65	0.27	0.36	0.50	0.64	0.79	0.93	Δ'	76	0.49	0.33	0.07	0.29	0.49	0.71	0.91
bb_strange_stories	Δ	0.67	0.15	0.48	0.62	0.69	0.81	0.84	Δ'	72	0.40	0.22	0.11	0.26	0.43	0.57	0.64
bb_code_line	Δ	0.85	0.00	0.85	0.85	0.85	0.85	0.85	Δ'	30	0.28	0.31	-0.08	0.08	0.30	0.49	0.66
bb_social_support	Δ	0.86	0.21	0.64	0.79	0.90	1.07	1.07	Δ'	60	0.76	0.19	0.50	0.68	0.78	0.87	0.96
bb_hindu	Δ	0.98	0.27	0.59	0.80	1.03	1.21	1.31	Δ'	48	0.73	0.29	0.32	0.55	0.77	0.90	1.12
bb_novel_concepts	Δ	1.00	0.00	1.00	1.00	1.00	1.00	1.00	Δ'	16	0.39	0.32	-0.03	0.19	0.44	0.58	0.79
bb_identify_odd	Δ	1.04	0.00	1.04	1.04	1.04	1.04	1.04	Δ'	23	0.62	0.33	0.14	0.40	0.62	0.88	1.01
bb_human_organisms	Δ	1.08	0.00	1.08	1.08	1.08	1.08	1.08	Δ'	21	0.36	0.33	-0.04	0.11	0.37	0.54	0.82
bb_riddle_sense	Δ	1.23	0.00	1.23	1.23	1.23	1.23	1.23	Δ'	24	0.75	0.29	0.38	0.53	0.79	0.94	1.12

Table 12: Distributions of the convergent validity quantities discussed in Appendix I across 100 Monte Carlo trials. k for the right-hand quantities is the average evaluation set size that was chosen.

math word problems <i>Domain Context</i> “solve finance calc. problem”	Prec@k	instruction following <i>Output Format</i> “format response as Markdown”	Prec@k	creative writing <i>Literary Genre</i> “write fantasy story”	Prec@k
finance calculation	1.000	Markdown	0.967	fantasy	0.759
travel distance	0.933	JSON	0.867	science fiction	0.679
construction material	0.900	table	0.633	mystery	0.654
cooking measurement	0.679	YAML	0.448	romance	0.625
inventory management	0.500	bullet list	0.414	horror	0.583
				historical fiction	0.517

email composition <i>Communication Tone</i> “write follow-up email”	Prec@k	debugging code <i>Error Type</i> “debug syntax error”	Prec@k	code generation <i>Programming Language</i> “write function in Python”	Prec@k
follow-up	0.880	syntax error	0.867	Python	0.867
formal business	0.808	runtime error	0.714	C++	0.741
urgent request	0.577	logic error	0.690	Rust	0.478
apologetic	0.476	security vulnerability	0.393	JavaScript	0.346
casual friendly	0.462	performance issue	0.357	Java	0.286
thank you	0.391	memory leak	0.269	Go	0.100

trip planning <i>Destination Type</i> “plan trip to multiple cities”	Prec@k	legal documents <i>Document Type</i> “draft partnership agreement”	Prec@k	workout design <i>Fitness Goal</i> “design workout for rehab.”	Prec@k
trip to multiple cities	0.733	partnership agreement	0.385	rehabilitation	0.500
trip under budget cap	0.640	licensing agreement	0.370	weight loss	0.273
trip to Asia	0.467	employment contract	0.333	strength	0.261
trip to Europe	0.400	non-disclosure agreement	0.250	muscle gain	0.208
trip to rural village	0.267	privacy policy	0.231	endurance	0.185
weekend trip only	0.227	terms of service	0.182	flexibility	0.111

Table 13: Dispersion in number of *Relevant* samples retrieved across facets of the same use-case. Most use-cases show substantial spread: some variants (green) have many *Relevant* examples, while others (red) are barely represented. Placed here for Section H.