# ResearchQA: Evaluating Scholarly Question Answering at Scale Across 75 Fields with Survey-Mined Questions and Rubrics

**Li S. Yifei***, **Allen Chang***, **Chaitanya Malaviya, Mark Yatskar**
University of Pennsylvania
{liyifei, cylumn}@seas.upenn.edu

**Data:** huggingface.co/datasets/realliyifei/ResearchQA
**Code:** github.com/realliyifei/ResearchQA
**Website:** cylumn.com/ResearchQA

## Abstract

Evaluating long-form responses to research queries is increasingly important for LLM agents, particularly emerging deep research systems. Such evaluation heavily relies on expert annotators, restricting attention to areas like AI where researchers can conveniently enlist colleagues. Yet, research expertise is abundant: survey articles consolidate knowledge spread across the literature. We introduce ResearchQA, a resource for evaluating LLM systems by distilling survey articles from 75 research fields into 21K queries and 160K rubric items. Queries and rubrics are jointly derived from survey sections, where rubric items list query-specific answer evaluation criteria, i.e., citing papers, making explanations, and describing limitations. 31 Ph.D. annotators in 8 fields judge that 90% of queries reflect Ph.D. information needs and 87% of rubric items warrant emphasis of a sentence or longer. We leverage ResearchQA to evaluate 18 systems in 7.6K head-to-heads. No parametric or retrieval-augmented system we evaluate exceeds 70% on covering rubric items, and the highest-ranking system shows 75% coverage. Error analysis reveals that the highest-ranking system fully addresses less than 11% of citation rubric items, 48% of limitation items, and 49% of comparison items. We release our data to facilitate more comprehensive multi-field evaluations.

## 1 Introduction

The rapid growth in research literature makes staying informed about advancements in many fields difficult (Price, 1963; Larsen and Von Ins, 2010). Large language model (LLM) tools, such as deep research systems (DeepMind, 2025; OpenAI, 2025) and scientific AI assistants (Skarlinski et al., 2024; Yang et al., 2024; Si et al., 2025; Singh et al., 2025), show potential to address this problem by meeting the information needs of both experts and non-experts. **However, evaluating long form answers to research queries is extremely challenging** (Xu et al., 2024). Several benchmarks have been proposed (i.a., Lee et al., 2023; Auer et al., 2023; Asai et al., 2024; Zhao et al., 2025), but are limited in size and primarily constrained to engineering domains (Table 1). Broader evaluation is necessary, but, as yet, has been unachievable because of a lack of affordable availability of appropriate experts.

In this paper, we introduce **ResearchQA, a literature-based resource for benchmarking research synthesis systems**. Our insight is to leverage academic surveys, whose role is to review a research field's foundational questions and synthesize relevant evidence (Kasanishi et al., 2023), for conveniently making comprehensive evaluation of research topics possible. We operationalize this idea by building a multi-stage pipeline to transform a pool of over 54K academic surveys into queries and answer *evaluation rubrics* (Lin et al., 2025; Sawada et al., 2025). Each rubric, derived jointly with queries by mining high quality survey sections, lists query-specific evaluation criteria, which may include citing papers, and making comparisons, or describing causal effects, among others (Figure 1). To demonstrate the effectiveness of literature-based evaluation, we release ResearchQA, containing **21.4K queries with 160k rubric items, across 75 research fields**, making it the most diverse and comprehensive benchmark of its kind to date (Tables 1 and 8).

We validate the quality of ResearchQA in a multi-field expert evaluation, spanning 31 Ph.D. level annotators with expertise from 8 fields. 90% of queries are judged as reflecting information needs of Ph.D. students and 86% are stylistically similar to how a Ph.D. student would have ex-

*Equal contribution.

pressed that need. Few queries are considered too open-ended, facilitating the opportunity for criteria-based evaluation over open interpretation. We also evaluate the quality of rubric items, finding that 87% of them address concepts that should be covered in at least a sentence by systems.

We explore whether rubrics can improve LLM-as-a-Judge accuracy. We sample research queries and ask the same annotators to vote for the higher response quality between two retrieval-augmented systems. Expert preferences are aggregated using majority voting and find that human accuracy to these labels is 84%, which approximates an upper bound for LLM-as-a-Judge accuracy. Proprietary and open-sourced judges that directly predict preference show 71% and 64% accuracy respectively, likely because they lack appropriate expert knowledge. To bridge this knowledge gap, we score each response from 0 to 4 for each rubric item's *coverage* (Figure 1). To our excitement, we find that agreement increases to 74% and 69% when coverage and direct predictions are combined in an ensemble LLM-as-a-Judge protocol.

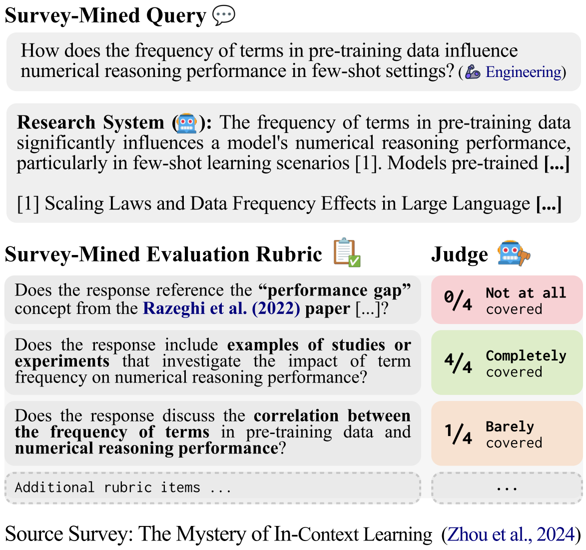


Figure 1: An example RESEARCHQA query and evaluation rubric. The query, mined from Zhou et al. (2024), instructs a research system to generate a long-form answer. An automatic evaluator creates an absolute measure of answer quality via a rubric with up to 8 items. The first rubric item cites Razeghi et al. (2022).

**With RESEARCHQA, we extensively evaluate frontier systems, revealing significant performance gaps between how different inference pipelines address scholarly queries.** We consider 18 systems across four tiers of test-time inference effort: language models using solely parametric memory, naive retrieval on Google Scholar, production-level retrieval optimized for consumer use, and deep research APIs. Each system is judged on two expert-validated evaluation metrics: (1) average rubric coverage %, and (2) Elo score using our ensemble LLM-as-a-Judge. The results reveal four noteworthy observations:

*All systems struggle to cover rubric items.* No parametric or retrieval system we evaluate exceeds 70% coverage. The highest-performing system we evaluate is Perplexity's deep research (Perplexity, 2025), with 75.29% coverage, indicating room for improvement.

*Models selectively benefit from retrieval.* When only relying on parametric memory, Claude-Sonnet-4 (Anthropic, 2024) ranks lower than Gemini-2.5-Pro (Comanici et al., 2025) (30% win rate), but when both models use retrieval Claude-Sonnet-4 ranks higher (75% win rate).

*Retrieval optimization is critical.* We observe that a naive retrieval implementation is often insufficient to improve coverage. On average, the same models differ by 7.6% coverage between naive and production retrieval.

*Deep research systems are significantly advantaged for scholarly queries.* Strikingly, Perplexity's deep research obtains 82% win rate over the next best system, revealing substantial gaps between the helpfulness of research-oriented tools.

Finally, we conduct an error breakdown of unaddressed rubric items, revealing that systems can improve across many criteria: common error cases include citing key works (under-addressed in 89% of cases), describing limitations (52% of cases), and making comparisons (51% of cases).

## 2 RESEARCHQA: Mining Queries and Rubrics from Academic Surveys

RESEARCHQA is a large-scale research question answering dataset, consisting of **21.4K queries across 75 research fields**. Research fields span 7 domains: Health Sciences & Medicine (7.5K queries), Life & Earth Sciences (4.9K), Engineering & Computer Science (4.7K), Physical Sciences (2.5K), Social Sciences (1.4K), Humanities (362), and Economics (55). Table 1 shows that RESEARCHQA contains similar desiderata of related benchmarks, while expanding on queries and research field diversity. Table 8 shows the number of queries for each field.

| Scholarly QA Benchmark | # Scholarly Queries | # Scholarly Fields & Research Domains | Abstractive Eval. Format | Multi-Doc Reasoning | Evaluation Rubrics | Auto-Generated |
|---|---|---|---|---|---|---|
| QASPER (1) | 5.0K | 1 (NLP ← 🦾) | ✗ Extractive | ✗ | ✗ | ✗ |
| QASA (2) | 1.8K | 1 (AI ← 🦾) | ✗ Extractive | ✗ | ✗ | ✗ |
| PubMedQA (3) | 1.0K | — (🫁) | ✗ Yes/No | ✗ | ✗ | ✓ |
| SciQA (4) | 2.5K | 1 (CS ← 🦾) | ✗ Extractive | ✗ | ✗ | ✓ |
| KIWI (5) | 0.2K | 1 (NLP ← 🦾) | ✓ Abstractive | ✓ | ✗ | ✗ |
| SciDQA (6) | 2.9K | 1 (AI ← 🦾) | ✗ Extractive | ✓ | ✗ | ✓ |
| SciQAG (7) | 188.0K | 20 (C&MS ← 🔭) | ✗ Extractive | ✗ | ✗ | ✓ |
| ScholarQABench (8) | 3.0K | — (🫁🦾🔭) | ✓ Abstractive | ✓ | ✓ | ✗ |
| SciArena† (9) | 8.2K | — (🫁🧬🦾🔭🏛🎭📈) | ✓ Abstractive | ✓ | ✗ | ✗ |
| RESEARCHQA (Ours) | 21.4K | 75 (🫁🧬🦾🔭🏛🎭📈) | ✓ Abstractive | ✓ | ✓ | ✓ |

Table 1: Comparison of RESEARCHQA to related benchmarks. Icons: 🫁 Health Sciences & Medicine; 🧬 Life & Earth Sciences; 🦾 Engineering & CS; 🔭 Physical Sciences; 🏛 Social Sciences; 🎭 Humanities; 📈 Economics. (1) Dasigi et al. (2021); (2) Lee et al. (2023); (3) Jin et al. (2019); (4) Auer et al. (2023); (5) Xu et al. (2024); (6) Singh et al. (2024); (7) Wan et al. (2024); (8) Asai et al. (2024); (9) Zhao et al. (2025) (†Concurrent work). Scholarly fields are marked as blank "—" when field descriptions are incomplete or metadata are missing.

To build RESEARCHQA, we create a multi-stage pipeline to generate queries and rubrics from survey articles (Figure 2). We describe the selection of top publication venues in §2.1 and survey articles in §2.2. We present query generation in §2.3 and rubric generation in §2.4. Finally, we describe dataset splits in §2.5. Further details about pipeline implementation are in Appendix A. To balance the cost and performance of pipeline creation, we select $\mathcal{M} \leftarrow$ `gpt-4.1-mini` whenever an LLM is used to generate or filter data. We discuss model alternatives in Appendix B and present a cost breakdown of different model $\mathcal{M}$ alternatives for the data pipeline in Table 10.

## 2.1 Extract Top Venues from Research Fields

To focus on high-quality surveys, we identify the top-20 publishing venues ranked by h5-index for each field in Google Scholar. Fields on Google Scholar can be overly specific (e.g., Wood Science & Technology) or general (e.g., Health & Medical Sciences (general)), so we manually redistribute from 257 to 94 fields. This process results in 660 venues that are used to retrieve survey articles.

## 2.2 Extract Survey Articles

We retrieve survey articles using keyword search on three sources: Crossref,[1] Semantic Scholar,[2] and S2ORC (Lo et al., 2020). In total, 615K candidate articles are returned, 134K of which are downloadable full-text articles, and 54K which are automatically classified to represent a true survey article versus merely containing survey-related keyword(s) in the title.

**Article search.** For each of 660 unique venues from Google Scholar, we retrieve publications with title keywords: *survey*, *literature review*, *a review*, *an overview*, and *meta-analysis*.

**Article filters.** To optimize precision in our search, we apply filters to remove erroneous articles (e.g., *survey* can be used in the context of field observations, not literature reviews). We prompt $\mathcal{M}$ to classify articles as literature reviews (F1=.80, Prec=.87), removing those judged not to be a literature review, i.e., work aimed at synthesizing and reviewing existing literature.

## 2.3 Generate Queries from Survey Content

We identify medium-length survey sections with multiple citations, then generate questions and reference answers using grouped sentences from each section. Queries are filtered by keywords and $\mathcal{M}$ to ensure they are standalone and without excessive variability in appropriate answers (F1=.86, Prec=.75). In total, 319K of 886K sections are used, yielding 21K queries after filtering.

**Survey section filters.** Sections are removed if section titles suggest they are not part of the main body (e.g., *abstract*). Next, we remove sections too short to be informative ($<$ 3 sentences or $<$ 800 characters), too long ($>$ 300K characters), or lacking citations ($<$ 3 inline citations).

**Query and reference answer generation.** We few-shot prompt $\mathcal{M}$ to extract hierarchical sum-

[1] https://www.crossref.org/
[2] https://www.semanticscholar.org

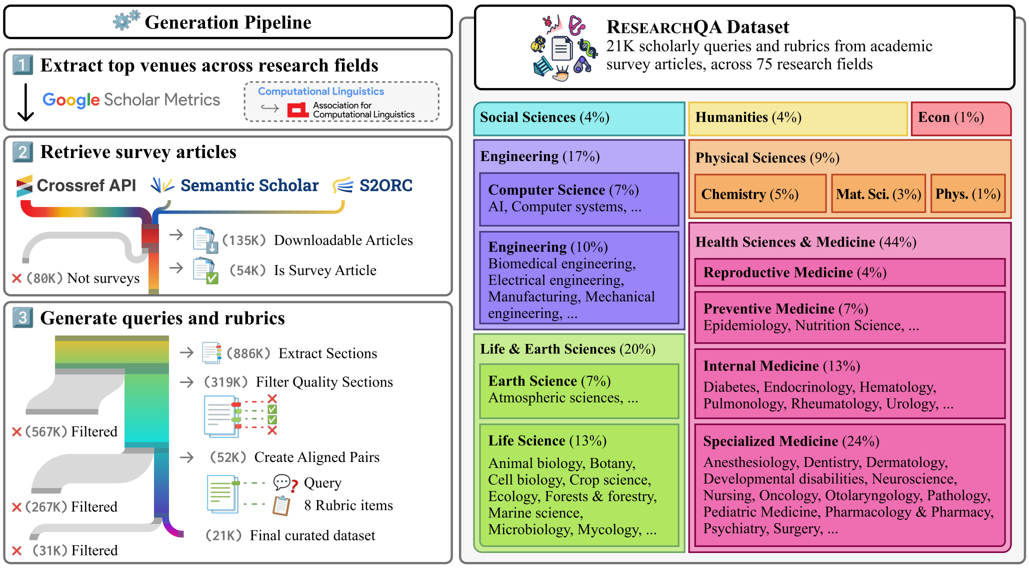


Figure 2: **(Left) ResearchQA generation stages:** We identify top-20 venues from each field in Google Scholar, retrieve survey articles from available databases, and generate queries and rubrics from survey sections. Throughout generation, we employ appropriate filtering mechanisms to ensure data quality. **(Right) ResearchQA test split field distribution:** Queries in the test split span 75 research fields from 7 domains, with high representation in Health Sciences & Medicine, Life & Earth Sciences, and Engineering.

maries (Christensen et al., 2014) from section content. Each summary consists of tree-based structure of questions with supporting sentences found in the section content (Benz and Jasinskaja, 2017; Wu et al., 2023), helping to generate queries that integrate multiple sources of supporting information (Figure 7). We prompt $\mathcal{M}$ with examples, using section content and the summary to generate a query and reference answer. Prompts instruct that queries must be supported by at least 3 sentences and reference answers summarize supporting sentences without introducing new evidence. Like Jansen et al. (2025), each query is paired with a knowledge cut-off date of the source article.

**Query filters.** We aim for queries that are (1) *standalone*, i.e., understandable by experts without extra decontextualization (Choi et al., 2021); and (2) *low in answer variability*, i.e., different experts are likely to provide similar ratings so that model responses can be more easily scored. To enforce these criteria, we use $\mathcal{M}$ to score standaloneness and answer variability of each query on a scale from 1 to 10. Queries scoring $< 7$ for standaloneness or $> 4$ for answer variability are removed.[3] Further, we discard queries that have keywords indicating context dependence (e.g., *the paper*, *this study*). We then remove queries with reference answers that are too short ($< 800$ characters). Finally, we re-apply $\mathcal{M}$ to remove queries that are misassigned to a field, which can occur when mining queries from multi-field venues.

### 2.4 Rubric Generation

Rubrics $\mathbf{R} = \{R_i\}_{i=1}^{K}$ consist of $K$ automatically generated rubric items $R_i$ that each evaluate aspects of answer quality. Below, we present the design choices and method for rubric creation.

**Desiderata.** Rubrics are distilled from survey articles, adopting limitations from the articles themselves. For example, citations and analysis in the survey article can be subjective. Consequently, there may be gaps in evaluation criteria, e.g., an important missing $R_i$, and not all rubric items may be important. We address these potential limitations through two design choices. (1) We generate and evaluate three rubric types: survey rubrics ($\mathbf{R}_S$), parametric rubrics ($\mathbf{R}_P$), and hybrid rubrics that merge the two ($\mathbf{R}_H \subset (\mathbf{R}_S \cup \mathbf{R}_P)$), covering obvious gaps in rubrics (Wadhwa et al., 2025). (2) To create $\mathbf{R}_H$, we deduplicate, rerank, and remove hallucinations from the union set $\mathbf{R}_S \cup \mathbf{R}_P$, which intends to remove unimportant or flawed rubric items. The top-8 hybrid rubric items are retained, leaving $\sim$7.5 rubric items on average.

**Rubric item generation.** We few-shot prompt $\mathcal{M}$, using question-answer pairs from ScholarQABench (Asai et al., 2024), to generate rubric

[3] Thresholds validated by NLP experts on initial dev. data.

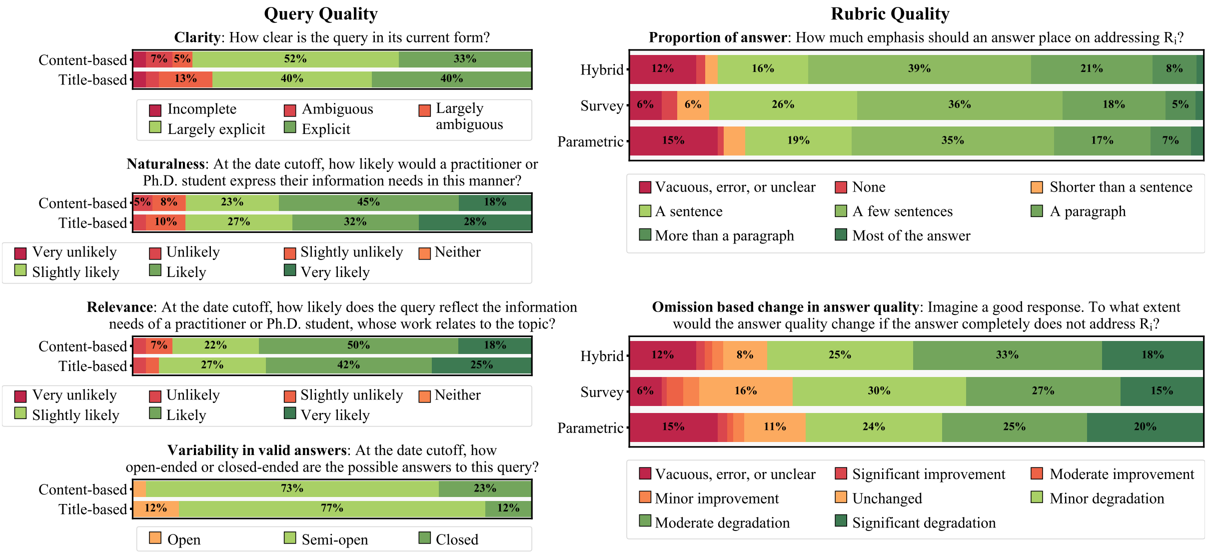


Figure 3: RESEARCHQA query and rubric quality ratings by 31 Ph.D. level experts.

items. $\mathbf{R}_S$ is conditioned on both the query and reference answer; $\mathbf{R}_P$ is conditioned on only the query. Each rubric item is created from one of three prompts designed to increase diversity of rubric item topics: information-based, depth-based, and citation-based items. Information-based items ask for specific statements, findings, opinions, or comparisons. Depth-based items ask for elaboration or explanation. Citation-based items ask whether answers cite a specific article. To create $\mathbf{R}_S$ and $\mathbf{R}_P$ with diverse items, we sample 4 information-based items, 2 depth-based items, and 2 citation-based items.

**Hybrid rubric construction.** Using $\mathcal{M}$, we deduplicate repeating rubric items that represent identical criteria and rerank rubric items based on their importance for answer evaluation. To remove hallucinations, we remove rubric items referencing papers that cannot be automatically matched to a Google Scholar article. Post-filters, about 61% of hybrid rubric items come from $\mathbf{R}_S$.

**Baseline rubric construction.** We additionally compare against a generic rubric ($\mathbf{R}_G$) that does not take query specifics into account. Rubric items are sampled from ScholarQABench (Asai et al., 2024) and SciArena (Zhao et al., 2025) metrics: Correctness, Citations, Coverage, Relevance, Organization, Usefulness, Attribution, and Examples. Table 11 presents the full list of rubric items.

### 2.5 Dataset statistics and splits

We create train ($\mathcal{D}_{\text{train}}$), validation ($\mathcal{D}_{\text{validation}}$), and test ($\mathcal{D}_{\text{test}}$) splits. $\mathcal{D}_{\text{test}}$ (3.7K queries) samples 50 queries from each field with $\geq$ 50 queries (75 of 94 fields), so that each field is sufficiently represented. $\mathcal{D}_{\text{validation}}$ (703 queries) samples up to 10 queries from remaining fields (74 of 75), and $\mathcal{D}_{\text{train}}$ is made up by the remaining 16.9K queries, which are for supporting the community in developing and tuning research systems.

| Section | Annotation Type | PA |
|---|---|---|
| §3.1 | Clarity | .72 |
| | Naturalness | .73 |
| | Relevance | .83 |
| | Variability in Valid Answers | .88 |
| §3.2 | Proportion of Answer | .91 |
| | Omission Based Change in Answer Quality | .94 |
| §3.3 | Rubric Coverage | .73 |
| | Pairwise Preference | .69 |

Table 2: Inter-annotator agreement across experts.

## 3 Expert Validation

This section presents the method and results of the expert validation study. We collect expert annotations for three judgement types: query quality (§3.1), rubric quality (§3.2), and scoring and preferences on system outputs (§3.3). Judgements are made on Likert-scale and multi-category labels, visualized in Figure 3, and are binarized by meaning for downstream analysis.

**Annotator recruitment.** We recruited annotators via Ph.D. email lists. 45 Ph.D. students and 1 postdoctoral staff registered to participate in annotations, and 31 ultimately completed the annotation task. Annotators' expertise span diverse fields: Natural Language Processing (15 experts), Computer Vision (6), Biomedicine (4), Linguistics (3), Physics (2), Genetics (1), Economics (1), and Psychology (1). Annotators were compensated $25 per hour of annotation.

**Annotation agreement.** We measure inter-annotator agreement (IAA) as binary pairwise agreement (PA) in Table 2. While tasks can be subjective, IAA is similar to prior work exploring expert judgements for scientific QA (e.g., pairwise preference is .69 vs .70 PA in Asai et al. (2024)).

### 3.1 Query Quality

**Setup.** Queries are judged on clarity, naturalness, relevance, and variability in valid answers. We compare queries from our pipeline (content-based) against an ablated pipeline that generates solely from survey titles (title-based), controlling for comparisons on the same query topic. This analysis isolates the effect of pipeline components used to enhance query generation and filtering.

**Label binarization.** We binarize labels by positive and negative meaning. Clarity merges *Largely explicit* and *Explicit*; Naturalness merges *Slightly likely* to *Very likely*; Relevance merges *Slightly likely* to *Very likely*; and Variability in valid answers merges *Semi-open* and *Closed*.

**Survey content yields unambiguous queries, eliciting answers without too much variability.** Experts rate ∼85% of content-based queries as *Largely explicit* or *Explicit*, indicating that queries generated from our pipeline have mostly specific and easy interpretations. Likely, content-generated queries have specific interpretations due to grounding in a reference text. Additionally, content-based queries have more *Semi-open* or *Closed-ended* queries (96% versus 89%), which supports valid answers that can be directly compared. By contrast, *Open-ended* queries might elicit a large number of possible answers. For example, *"What distinguishes robustness under distribution shift from domain adaptation and transfer learning in NLP?"* can yield answers that experts find difficult to rank and compare.

**Queries support the information needs of researchers and are naturally expressed.** Queries generated from both full and ablated pipelines have similar relevance and naturalness. 94-90% are rated as *Slightly likely* or stronger to reflect researchers' information needs, and 86% of queries *Slightly likely* or stronger to be expressed in that way by a researcher. These ratings also indicate room for improvement, because queries rarely are rated as *Very likely* to be relevant or naturally expressed.

### 3.2 Rubric Quality

**Setup.** Rubric items are judged on the number of sentences that should be used to address the item in an answer and how its omission affects answer quality (Figure 3). Additionally, annotators can flag a number of errors of a rubric item $R_i$: (1) $R_i$ is difficult to judge; (2) $R_i$ is unclear; (3) $R_i$ is an empty or unspecific assessment (e.g., it is a mere rephrasing of the query);[4] (4) $R_i$ contains an error (e.g., it cites a non-existent paper). The presence of any of these flags voids the quality of the rubric item. We compare among hybrid, survey, and parametric rubrics.

**Label binarization.** In "Proportion of answer measurement", we merge *None* and *Shorter than a sentence*. In "Omission based change in answer quality" we merge *Significant improvement* to *Minor improvement*.

**Hybrid rubrics are likely to contain criteria worth describing in multiple sentences and improve answer quality.** Across rubric types, rubric items should be described in *A sentence* or more 84-86% of the time and their omission would degrade answer quality 69-74% of the time. Hybrid rubrics items are rated to be greatly important at a higher frequency, specifically those causing *Moderate degradation* or *Significant degradation* to answer quality when removed.

**Hybrid rubrics contain few non-existent papers and vacuous restatements of the query.** Parametric rubrics have the highest rate of *vacuous, error, or unclear* items (15%). Experts note that parametric rubric items contain nonexistent paper titles, causing statements about their coverage in answers to be unanswerable or ambiguous.

### 3.3 System Output Preferences and Evaluation Protocol Validation

**Setup.** We collect expert judgements on head-to-head answers (Figure 8) to perform a meta-evaluation on LLMs using rubrics to approximate

[4] A rule-based system is also employed to detect unspecific rubric items, where rubric items need to include at least one substantive word not present in the original query.

expert judgements. Experts provide two types of annotations: (1) Rubric coverage: Experts rate how well each answer covers each rubric item on a 5-point scale (0 = *Not at all*, 4 = *Completely*). (2) Pairwise preference: Experts compare two answers side-by-side (in random order) and select: *Left* is better, *Right* is better, *Tie*, or *Both bad*. Majority voting determines the final label.

Answers are generated from `gpt-4.1-mini` and `gemini-2.5-flash` using their providers' embedding models[5] to retrieve relevant passages as input context. We retrieve the top-20 arXiv papers via Google search on the given query and constrain search by the appropriate date cutoff. Papers are chunked into 1000 character passages, and the top-20 passages ranked by embedding similarity score are input as context.

**Label binarization.** Rubric coverage merges (0 = *Not at all*, 1 = *Barely*) and (2 = *Moderately*, 3 = *Mostly*, 4 = *Completely*). For pairwise preference, we drop *Tie* and *Both bad*, keeping only direction labels, i.e., *Left* or *Right*. PA are 0.73 and 0.69 respectively. Because agreement for pairwise preference is low, we collect up to 4 expert judgements and keep only those that had a majority voted direction label (89.7% of cases with direction judgements). Human annotators with direction label preferences have 84% accuracy agreement with the majority vote label.

**Evaluator LLMs.** Open-source evaluators (`prometheus-2-8x7b`; Kim et al., 2024) and proprietary LLMs (`claude-4-sonnet` and `gpt-4.1-mini`) are validated as automatic evaluators. For each evaluator, we compute agreement to expert annotations on rubric coverage and pairwise preference, which represent absolute and relative measures of answer quality respectively.

We prompt an LLM to produce rubric coverage in one call, inputting a system answer $A$ and rubric $\mathbf{R} = \{R_i\}_{i=1}^{K}$ to quantify the number of rubric items covered in $A$ on a 5-point scale, $\text{Coverage} : A \times \mathbf{R} \rightarrow \{0, 1, 2, 3, 4\}^K$. The prompt describes the ends of the scale with labels from the user study (0 = *Not at all*, 4 = *Completely*).

We compare two types of automatic judges to predict pairwise preference. The **Direct Judge** ($\text{J}_{\text{direct}}$) is prompted with the query and two system answers and asked "Which response is better?". Both answer orderings are evaluated to reduce positional bias (Shi et al., 2024). However, direct comparison may lack appropriate knowledge useful to predict expert preferences. To address this gap, we introduce **Ensemble Judge** ($\text{J}_{\text{ensemble}}$), which uses both $\text{J}_{\text{direct}}$ and $\text{Coverage}$ to predict preferred answers: the answer with a larger sum of $\text{Coverage}$ (0-4 scale) and a full 4 points for each $\text{J}_{\text{direct}}$ comparison, is marked as the preferred answer. Formally, this sum can be expressed as:

$$4(\mathbb{I}_A + \mathbb{I}'_A) + \textstyle\sum_{c \in \text{Coverage}(A, \mathbf{R})} c \qquad (1)$$

where $\mathbb{I}_A$ indicates 1 if $A$ is preferred by $\text{J}_{\text{direct}}$ and 0 otherwise; $\mathbb{I}'_A$ implements $\text{J}_{\text{direct}}$ with answers input in the reverse order. $\text{J}_{\text{ensemble}}$ prefers the answer with the larger sum or outputs a tie.

[5] `text-embedding-3-large`, `text-embedding-004`

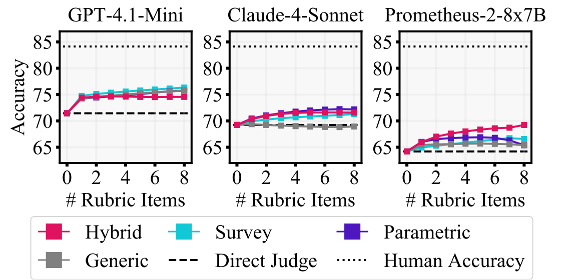


Figure 4: A comparison of how much rubrics can aid different evaluators in making predictions that agree with plurality human labels (y-axis) as a function of rubric size (x-axis). All direct judges benefit from integration of rubrics through the hybrid judge, substantially reducing their disagreement with human experts.

**LLM-Human agreement on Rubric Coverage.** All evaluator LLMs show fair Pearson correlation with expert annotations on rubric coverage: `prometheus-2-8x7b` at .48, and both `claude-4-sonnet` and `gpt-4.1-mini` at .63. Averaged over our annotated data, experts and `gpt-4.1-mini` differ on Coverage by only 0.11. On the other hand, `gpt-4.1-mini` tends to make more extreme predictions of Coverage for individual samples. Experts rate a rubric items *Completely* covered 12.3% of the time while `gpt-4.1-mini` does so 26.0%. While a suitable judge on average, conclusions with `gpt-4.1-mini` may overestimate the frequency of some coverage values at the extremes.

**LLM-Human agreement on Pairwise Preference.** We present the preference ranking accuracies of evaluator LLMs with respect to majority labels in Figure 4. Each graph presents a base direct judge and the results of ensembling with the four types of rubrics we consider. In cases where judges predict ties, we assign partial credit.[6] Human-Human accuracy agreement indicates an upper bound of 84% for LLM-Human agreement. Ensemble judges ($J_{\text{ensemble}}$) are consistently better estimators of expert preferences than direct judges ($J_{\text{direct}}$), achieving up to $\sim$75% accuracy. Notably, rubrics can decrease the LLM-Human and Human-Human agreement gap from 12.7% to 9.6%, corresponding to a 24% relative reduction. In general, query-specific rubrics match or outperform generic rubrics. Hybrid rubrics demonstrate high performance gain across all systems, allowing `prometheus-2-8x7b` to match the direct judge performance of `claude-4-sonnet`. As visualized in Figure 11, hybrid rubrics enhance evaluation in 7 of 8 fields, where they are the best-performing rubric in 4 of 8. The performance of parametric and generic rubrics depend on the strength of the evaluator: in strong evaluators, they can match hybrid rubrics, but they yield little to no benefit in open evaluators.

# 4 LLM Systems Evaluation Setup

To benchmark top systems used for scholarly inquiry on ResearchQA, we evaluate 18 parametric, retrieval augmented, and deep research systems in both open-source and proprietary families (Table 3) on $\mathcal{D}_{\text{test}}$. We perform all analyses with the best-performing rubrics (hybrid rubrics), evaluator LLM (`gpt-4.1-mini`), and protocol ($J_{\text{ensemble}}$). We describe the task setup in §4.1 and tournament setup in §4.2.

## 4.1 Evaluation Task and Metrics

**Task.** We consider the following task: systems are input a query $Q$ and generate a citation-supported answer $A$, constrainting on date $D$ and a response length $L$ = 250 words. A sample answer is visualized in Figure 9. This task design conservatively guards against potential biases from information recency, multi-turn clarification, and length biases (Singhal et al., 2024), trading off task naturalness. We explore more natural setups in §5.2, finding that $D$ can be presently discarded with low likelihood of bias. Further, we remove $L$, allowing longer responses that may result from multi-turn interaction or no length specification.

[6] We have 3 cases of partial credit: (1) for the direct judge, if $\mathbb{I}_A \neq \mathbb{I}'_A$, 50%, (2) if $\mathbb{I}_A$ or $\mathbb{I}'_A$ is correct and the other reports tie, 75%, and (3) if $J_{\text{ensemble}}$ reports tie, 50%.

**Coverage %.** We compute average Coverage over answers and divide by 4, normalizing the resulting percentage from 0% to 100%.

**Leaderboard score.** Consistent with Chatbot Arena (Chiang et al., 2024), we use the Bradley-Terry model (Bradley and Terry, 1952) for pairwise battles judged by $J_{\text{ensemble}}$, which is equivalent to the Elo equation (Elo, 1966). In case of ties, each system wins half the match. We report the median score from a 1000-iteration bootstrap and its standard deviation.

## 4.2 Tournament Details

We explore systems at a gradation of engineering effort: parametric, naive retrieval, production retrieval, and deep research.

**Parametric systems.** We directly generate answers using default configs of API providers (OpenRouter[7] for open-source models).

**Naive retrieval systems.** These systems represent retrieval with minimal engineering effort: language models conditioned on the top-20 retrieved passages. Each language model is paired with an embedding model from the same provider. We retrieve papers from a date-constrained Google Scholar search. For each query, up to 50 papers are retrieved: the top-20 using the query as search field, and an additional 10 for each of 3 related keywords generated by `gpt-4.1-mini`, a search method used in Asai et al. (2024). Papers corresponding to distilled survey papers are removed to prevent unfair evaluation advantages due to data leakage. Each paper is chunked into 1000-token passages with 200 overlapping tokens of subsequent chunks, and the top-20 passages by embedding similarity score are used as context, optionally reranking when a cross-encoder reranker is available. We generate answers using the same decoding configurations as in the parametric setting and employ greedy decoding for OpenScholar.

**Production retrieval systems.** These systems represent retrieval with advanced engineering effort, e.g., tool use, feedback, or refinement. We use default implementations in their respective repositories or APIs without modification.

**Deep research systems.** At the time of writing, two commercial deep research systems have API

[7] https://openrouter.ai/

| System | Coverage % ↑ | | | | | | | | Leaderboard Score ↑ | Avg Length (Words) |
|---|---|---|---|---|---|---|---|---|---|---|
| | All domains | | | | | | | | | |
| Parametric | | | | | | | | | | |
| `llama-3.3-70b` | 53.42 ± 0.26 | 51.82 | 54.21 | 54.89 | 55.74 | 53.91 | 53.22 | 58.10 | 617 ± 13 | 167.4 |
| `claude-4-sonnet` | 64.31 ± 0.31 | 62.92 | 64.96 | 66.71 | 67.08 | 63.33 | 59.85 | 66.17 | 1099 ± 09 | 226.5 |
| `gpt-4.1` | 65.43 ± 0.25 | 63.98 | 66.84 | 66.66 | 67.38 | 65.45 | 63.48 | 68.46 | 1080 ± 09 | 241.5 |
| `qwen-3-32b` | 66.64 ± 0.26 | 65.13 | 67.76 | 68.25 | 69.24 | 65.96 | 64.32 | 69.62 | 1038 ± 09 | 219.3 |
| `gemini-2.5-pro` | **68.84 ± 0.25** | **67.42** | **70.20** | **69.82** | **71.86** | **68.28** | **65.83** | **72.06** | **1244 ± 10** | 267.1 |
| Retrieval (Naive) | | | | | | | | | | |
| `openscholar-8b`$^{i}$ | 54.71 ± 0.28 | 54.08 | 56.15 | 54.76 | 54.98 | 54.67 | 52.69 | 57.46 | 478 ± 17 | 499.9 |
| `gemini-2.5-pro`$^{ii}$ | 59.92 ± 0.30 | 58.73 | 61.46 | 61.13 | 61.72 | 57.79 | 56.71 | 63.96 | 945 ± 10 | 270.4 |
| `qwen-3-32b`$^{iii}$ | 60.90 ± 0.32 | 57.62 | 62.93 | 64.25 | 65.58 | 60.49 | 60.23 | 65.82 | 1011 ± 09 | 265.5 |
| `claude-4-sonnet`$^{iv}$ | 62.50 ± 0.32 | 61.94 | 63.48 | 62.97 | 64.01 | 61.67 | 58.05 | 65.74 | 972 ± 09 | 238.4 |
| `gpt-4.1`$^{v}$ | **64.80 ± 0.29** | **63.69** | **66.65** | **65.11** | **66.72** | **64.25** | **62.09** | **66.33** | **1020 ± 09** | 263.6 |
| Retrieval (Production) | | | | | | | | | | |
| `sonar` | 58.61 ± 0.29 | 56.61 | 60.55 | 59.43 | 61.62 | 59.97 | 57.48 | 62.80 | 862 ± 10 | 242.2 |
| `openscholar-8b+feedback` | 58.72 ± 0.32 | 57.77 | 59.96 | 58.62 | 61.48 | 57.27 | 57.29 | 62.48 | 769 ± 12 | 788.8 |
| `sonar-reasoning` | 64.33 ± 0.31 | 62.73 | 66.00 | 65.19 | 68.11 | 62.68 | 61.76 | 67.49 | 1115 ± 10 | 280.5 |
| `gpt-4o-search-preview` | 65.98 ± 0.27 | 65.52 | 68.21 | 65.01 | 66.60 | 66.07 | 62.62 | 65.63 | 992 ± 09 | 255.0 |
| `gemini-2.5-pro+grounding` | 68.51 ± 0.25 | 67.38 | 70.02 | **68.76** | **70.99** | 68.09 | **65.98** | **71.21** | 960 ± 09 | 278.5 |
| `claude-4-sonnet+ws` | **69.18 ± 0.26** | **69.54** | **70.49** | 67.59 | 70.28 | **68.14** | 64.70 | 67.13 | **1149 ± 10** | 327.8 |
| Deep Research | | | | | | | | | | |
| `o4-mini-deep-research`$^{\dagger}$ | 72.69 ± 0.54 | 74.02 | 73.58 | 70.57 | 74.04 | 73.25 | 68.99 | 74.54 | 1145 ± 10 | 271.6 |
| `sonar-deep-research` | **75.29 ± 0.26** | **75.01** | **76.31** | **74.48** | **76.77** | **75.34** | **72.47** | **78.01** | **1505 ± 17** | 267.3 |

Table 3: Performances of LLM systems in a pairwise tournament across domains ( Health Sciences & Medicine; Life & Earth Sciences; Engineering & CS; Physical Sciences; Social Sciences; Humanities; Economics.) Retrieval embedding models belong to corresponding system providers: $^{i}$`openscholar-retrieve` and `openscholar-reranker`; $^{ii}$`text-embedding-004`; $^{iii}$`gte-qwen-2-7b-instruct`; $^{iv}$`voyage-3-large`; $^{v}$`text-embedding-3-large`. $^{\dagger}$This system costs ∼\$1.15 per query, so statistics are only computed on answers sampled for tournament battles (∼20%).

access: OpenAI and Perplexity deep research. We use both APIs without modification.

**Battle construction.** We construct a total of 7.6K battles, where each domain consists of a minimum of 1K battles and up to as many queries represented in the domain (1.8K for Health Sciences & Medicine). Each battle is constructed such that models are sampled uniformly (∼800 battles per model) and all model matchups are uniformly represented in each domain, ensuring connectivity of the comparison network.

| Systems | Avg Length | | | Coverage % | | |
|---|---|---|---|---|---|---|
| | $L$=250 | $\neg L$ | Δ% | $L$=250 | $\neg L$ | Δ |
| Parametric | | | | | | |
| `qwen-3-32b` | 219.5 | 176.5 | −19% | 66.8 | 64.0 | −2.8 |
| `gemini-2.5-pro` | 268.5 | 241.4 | −10% | 69.9 | 67.5 | −2.5 |
| Retrieval (Naive) | | | | | | |
| `claude-4-sonnet` | 238.9 | 299.5 | +25% | 63.6 | 62.8 | −0.8 |
| `gpt-4.1` | 264.0 | 246.1 | −6% | 65.0 | 63.7 | −1.3 |
| Deep Research | | | | | | |
| `o4-mini-deep-research` | 268.0 | 595.0 | +122% | 72.2 | 78.9 | +6.7 |
| `sonar-deep-research` | 267.2 | 1431.0 | +435% | 76.2 | 85.3 | +9.1 |

Table 4: Removing the $L$=250 words length guidance prompt can affect average answer length (words). Longer answers tend to score higher coverage, because coverage is a recall-based measure. Analysis is performed on a 225 query subset of $\mathcal{D}_{\text{test}}$ (3 per field).

# 5 LLM Systems Evaluation Results

## 5.1 Response Results

**Results by systems.** All systems struggle to fully cover rubric items, with no parametric or retrieval augmented systems we evaluated exceeding 70% coverage (Table 3). The best-performing system `sonar-deep-research` reaches not much more, obtaining 75% coverage. These results demonstrate headroom for further improvement. Win-rates of different systems indicate large performance gaps: systems designed for research synthesis have large advantages over others. In fact, `sonar-deep-research` is estimated to have an 82% win rate over the next highest rated system, `gemini-2.5-pro`.

| Item Type | Description | Example | Frequency % | Error % |
|---|---|---|---|---|
| Citation | $X$ is cited | Does the response cite Kvaskoff et al. (2015) (title: [...]) that links endometriosis with elevated cardiovascular risk? | 8.3 | 89.3 |
| Limitation | Limitations of $X$ are mentioned | Does the response address limitations of CTC in detecting small polyps and flat adenomas? | 2.7 | 52.4 |
| Comparison | $X$ and $Y$ are compared | Does the response compare reaction rates before and after catalyst saturation occurs? | 14.2 | 51.9 |
| Example | Examples of $X$ are mentioned | Does the response include forage species (e.g., legumes, chicory) affecting lamb meat's fatty acid profile? | 11.2 | 46.8 |
| Impact | Cause or impact of $X$ is mentioned | Does the response mention the preservation of the anterior cruciate ligament (ACL) as a benefit of UKA? | 15.5 | 46.3 |
| Other | None of the above | Does the response discuss METEOR's fragmentation penalty and its role in evaluating word order? | 48.1 | 43.6 |

Table 5: Error rates for different rubric types, measured as the percentage of items not rated as *Completely* covered by the best-performing system (`sonar-deep-research`). Each rubric type provides a description (with $X$ representing a concept, method, or paper being evaluated) along with an illustrated example.

We visualize sample responses from each system category in Figure 9 and Figure 10, presenting a case study to show qualitative differences. Here, the question is: "What strategies are recent research efforts employing to reduce the wall-clock training time for BERT models? (Date: 2020-10-02)". The parametric response covers sensible methods for training optimization (*mixed-precision training*, *gradient accumulation*) addressing content mentioned in rubrics. However, the response lacks detail about each method. Naive retrieval locates and explains in detail Li et al. (2020), describing that *the most effective recent strategies involve training larger models for fewer steps*. The response hardly addresses prototypical explanations to the query, scoring lower coverage on the rubric. We further compare production retrieval and deep research for the query, "How does substrate temperature influence the crystal orientation and quality of AlN (002) thin films during sputtering? (Date: 2020-10-19)". While the production retrieval response appears sensible, the deep research answer mentions the same explanations (*atomic mobility*) while uniquely discussing further aspects of temperature influences (*stress evolution*) in detail (*<300°C [increases] residual stress up to 1.2GPa*). In general, we find that deep research systems cover prototypical explanations while also dense in breadth and depth, demonstrating several advantages over systems from other categories.

**Results by rubric type.** To identify areas for improvement, we break down coverage of rubric items addressing different types of evaluation criteria. Each rubric item is lemmatized, using NLTK (Bird et al., 2009), and categorized using rules to 6 rubric types (Macro F1=.92, without "Other"). In Table 5, we show examples and the distribution of `sonar-deep-research` errors. We report the proportion of rubric item types that are not *Completely* covered: citation-based items can be improved the most (89%), followed by describing important limitations (52%) and making comparisons (52%). Error breakdowns of all other systems are shown in Figure 12.

**Results by domains.** Differences between domains are within ±6 Coverage % (Table 3). While differences are small, Health Sciences and Humanities rank as the 2 lowest performing domains: in 10/18 systems for Health Sciences and 18/18 for Humanities. These trends are suggestive that evaluations focused on Engineering and Physical Sciences, which rank as top-performing domains and are also heavily represented in prior evaluations (Table 1), provide an incomplete analysis of performance across research domains.

### 5.2 Effects of Information Recency, Survey Leakage, and Answer Length

In this section, we explore biases on Coverage % due to citing recent information, survey retrieval leakage, and constrained answer lengths.

**Information recency analysis.** We examine whether relying on sources published after the distilled survey articles unfairly lowers Coverage %. For each answer, we count the % of cited sources that violate the instructed date cutoff (survey article publication year). Sources that violate the date

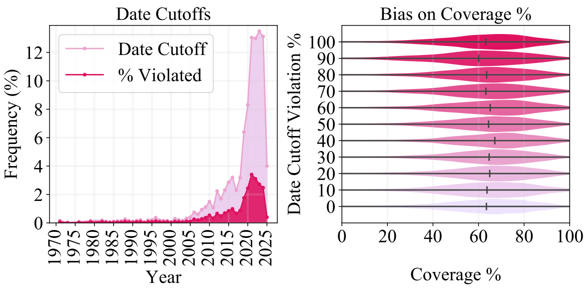


Figure 5: Rubrics are recent, mostly originating from surveys in the past decade. When cited sources violate date cutoff years, there is little bias on Coverage %.

cutoff make up ∼30% of the citations (Figure 5), suggesting systems do not adhere to date cutoffs stated in instruction text. Despite high frequency of date violations, we observe that recent sources do not positively or negatively affect Coverage % in aggregate: Coverage % is on average ±1.3% relative to mean coverage when there are no date violations. While current effects of date violations are small, we recommend monitoring possible information recency biases in future evaluations.

**Leakage analysis.** By distilling rubrics from downloadable survey papers, systems may gain evaluation advantages when retrieving the same survey papers to generate answers, i.e., *leakage*. To mitigate leakage from affecting evaluation, we restrict source papers from appearing in search retrievals in systems where appropriate control is possible. We additionally perform post-hoc leakage analysis, because many production systems do not have the option to restrict search retrieval for answer generation. Leakage is detected by checking whether the survey titles appears in the reference section of the answer.[8] We compare Coverage % in subsets of answers where leakage is and is not detected for top-performing production systems in Table 6 and all systems in Table 13.

Leakage advantages are small and inconsistent: in 13 systems exhibiting leakage, Coverage % increased for answers with leakage in 8/13 systems, where Coverage % actually decreased in the other 5/13 systems. In aggregate, we observe increases of ∼1.1% when distilled surveys are retrieved to generate answers. These small differences assuage our concerns of leakage advantages in current systems, but evaluations of future systems should continue verifying that leakage advantages are minimal and restrict source surveys from search results whenever possible.

[8]To address noise in paper titles, we lowercase text, remove special characters, and search for 6-gram matches.

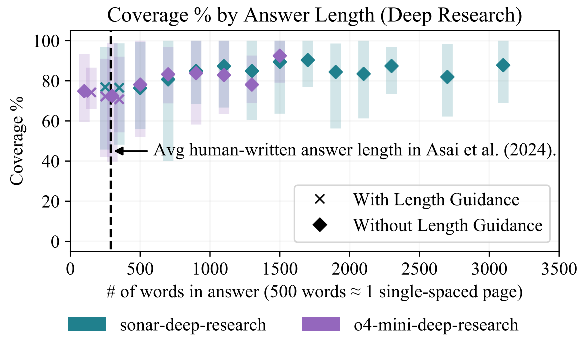


Figure 6: Coverage % increases with answer length, up until ∼2K words (about 4 pages of text).

| System | L% | Coverage % | | |
|---|---|---|---|---|
| | | ¬ Leaked | Leaked | Δ |
| `claude-4-sonnet+ws` | 30 | **71.27** | 68.30 | −3.0 |
| `gemini-2.5-pro+grounding` | 2 | 66.91 | **68.54** | +1.6 |
| `o4-mini-deep-research` | 28 | **74.82** | 71.85 | −3.0 |
| `sonar-deep-research` | 21 | 74.46 | **75.51** | +1.1 |

Table 6: Production systems often cite the related survey as a source, at 20-30% leakage (L%). However, Coverage % roughly stays the same with leakage (Leaked) and without leakage (¬ Leaked).

**Unconstrained answer length analysis.** We remove the 250 word constraint from the task instruction text to further explore how Coverage %, a recall-based metric, increases when we do not control for answer length during evaluations. We select from the top performers for parametric, naive retrieval, and deep research systems by Coverage % and additionally generate answers for 3 queries per field (225 total queries). Average statistics are shown in Table 4, and trends for deep research systems are visualized in Figure 6 and for the remaining systems in Figure 13.

Unconstrained length analyses reveal that many systems naturally generate answers of close to 250 words, with no substantial differences to answer length or Coverage % (Figure 13). However, deep research systems tend to generate longer answers, up to 3K words (about 6 pages of text) per answer. By contrast, expert-written answers lengths for similar topics average 289 words in Asai et al. (2024). Coverage % increases with answer length up to ∼2K words (about 4 pages of text), as shown in Table 4, plateauing at 85% for `sonar-deep-research` and 79% for `o4-mini-deep-research`. These trends suggest that deep research systems are able to convey more helpful information while trading off concise and focused answers.

## 6 Related Work

We fit into a broad body of work trying to improve benchmarking for LLMs. This includes multi-domain works (Hendrycks et al., 2021; Wang et al., 2025), challenge sets (Rein et al., 2024; Phan et al., 2025; Wolfson et al., 2025) and expert annotations (Malaviya et al., 2024, 2025). Model development for survey creation has used survey data material (Goldfarb-Tarrant et al., 2020; Kasanishi et al., 2023; Agarwal et al., 2024; Wang et al., 2024). Below we outline the key differences between RESEARCHQA and others.

**Manually crafted scholarly benchmarks.** Manual curation of benchmarks for scholarly QA has encountered practical challenges for creating large, diverse, and complex datasets. QASPER (Dasigi et al., 2021) and QASA (Lee et al., 2023) limit their focus on queries that can be answered within a single paper. KIWI (Xu et al., 2024) is built from questions derived by researchers on related work sections, and Scholar-QABench (Asai et al., 2024) recruited researchers to write questions from scratch. Both efforts have multi-document queries but are smaller than RESEARCHQA because of the challenges of recruiting researchers. Concurrently, SciArena (Zhao et al., 2025) leverages community contributions to collect queries and preferences; however, it remains smaller in size and coverage than RESEARCHQA and lacks evaluation rubrics.

**Auto-generated scholarly benchmarks.** Automatically generated scholarly QA benchmarks trade-off scale for complexity and naturalness. DeepScholar-Bench (Patel et al., 2025) uses 63 CS papers to compare machine-written and human-written related works. SciQA (Auer et al., 2023) is template generated and focuses on questions generated from knowledge graphs. PubMedQA (Jin et al., 2019) generates queries from the abstracts of PubMed articles, but limits to yes or no questions. SciDQA (Singh et al., 2024) extracts 188k queries asked during peer review on OpenReview but limits queries to extractions about the paper being reviewed. RESEARCHQA achieves more abstractive queries and evaluation materials by focusing on surveys but is smaller than SciDQA.

**Evaluating long-form answers and rubrics.** A central problem for long-form answer evaluation is a large space of possible correct answers. Reference answers can be difficult to use, and token based measures like ROUGE (Lin, 2004) are gameable (Krishna et al., 2021). Recent efforts rely on direct evaluator LLMs (Asai et al., 2024; Dubois et al., 2024) but may inherit self-preference (Panickssery et al., 2024; Wataoka et al., 2024), length biases (Dubois et al., 2024) and may be inaccurate for research queries. RESEARCHQA shows these judges can be effective. *Rubric-based evaluations* that decompose judgment into nuanced criteria (Sawada et al., 2025; Liu et al., 2023) are promising alternatives to direct evaluator LLMs. Manually curated rubrics are often task-specific and small scale (Asai et al., 2024; Starace et al., 2025; Qin et al., 2024). WildBench (Lin et al., 2025) creates query-specific checklist rubrics from parametric memory at scale. EvalAgent (Wadhwa et al., 2025) discovers query-specific rubrics with online search. Our rubrics leverage discovery from surveys, aligning them to expert-written material.

## 7 Conclusions

We introduce RESEARCHQA, a resource distilled from survey articles for large scale and multi-field evaluations of research synthesis. We leverage RESEARCHQA to benchmark 18 systems showing each has headroom for improvement. The highest-ranking deep research system, which achieves 82% win rate over the next system, only fully addresses fewer than 11% of items addressing citations, 48% of items describing limitations, and 49% of items asking about comparisons. RESEARCHQA can be expanded with newly written surveys as to increase coverage of new topics.

## Limitations

While RESEARCHQA reduces reliance on experts, expert involvement is not eliminated. We depend on experts and expert-written articles to build and validate RESEARCHQA. To the extent that we can, we recruit experts to validate data, but not all fields are equally validated. Additionally, open access to research articles is limited, hindering exhaustive coverage of the literature. Rubrics distilled from these articles may not fully cover important criteria; we leave synthesis of larger and comprehensive rubrics for future work. Many recently released deep research systems involve limited access; their omission can distort leaderboard rankings and obfuscate important failure modes.

## Acknowledgements

This research was developed with funding from the Defense Advanced Research Projects Agency's (DARPA) SciFy program (Agreement No. HR00112520300) and the National Science Foundation Graduate Research Fellowship under Grant No. DGE-2236662. The views expressed are those of the authors and do not reflect the official policy or position of the sponsors. We would like to thank the UPenn NLP group and Material Science and Engineering, Biomedical Engineering, Computer and Information Science, Genetics, Linguistics, Physics and Astronomy, Criminology, and Psychology departments for generously participating in our annotation. We thank Zack Ives for aiding university wide recruitment.

## References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. GPT-4 technical report. *arXiv preprint*, cs.CL/2303.08774.

Shubham Agarwal, Gaurav Sahu, Abhay Puri, Issam H. Laradji, Krishnamurthy DJ Dvijotham, Jason Stanley, Laurent Charlin, and Christopher Pal. 2024. LLMs for literature review: Are we there yet? *Transactions on Machine Learning Research*.

Anthropic. 2024. Introducing Claude 4.

Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph C. Chang, Kyle Lo, Luca Soldaini, Sergey Feldman, Mike D'arcy, David Wadden, Matt Latzke, Minyang Tian, Pan Ji, Shengyan Liu, Hao Tong, Bohao Wu, Yanyu Xiong, Luke Zettlemoyer, Graham Neubig, Dan Weld, Doug Downey, Wen tau Yih, Pang Wei Koh, and Hannaneh Hajishirzi. 2024. OpenScholar: Synthesizing scientific literature with retrieval-augmented LMs. *arXiv preprint*, cs.CL/arXiv.2411.14199.

Sören Auer, Dante A. C. Barone, Cassiano Bartz, Eduardo G. Cortes, Mohamad Yaser Jaradeh, Oliver Karras, Manolis Koubarakis, Dmitry Mouromtsev, Dmitrii Pliukhin, Daniil Radyush, Ivan Shilin, Markus Stocker, and Eleni Tsalapati. 2023. The SciQA scientific question answering benchmark for scholarly knowledge. *Scientific Reports*.

Anton Benz and Katja Jasinskaja. 2017. Questions under discussion: From sentence to discourse. *Discourse Processes*.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. " O'Reilly Media, Inc.".

Ralph A. Bradley and Milton E. Terry. 1952. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios N. Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. 2024. Chatbot Arena: An open platform for evaluating LLMs by human preference. In *International Conference on Machine Learning (ICML)*.

Eunsol Choi, Jennimaria Palomaki, Matthew Lamm, Tom Kwiatkowski, Dipanjan Das, and Michael Collins. 2021. Decontextualization: Making sentences stand-alone. *Transactions of the Association for Computational Linguistics*.

Janara Christensen, Stephen Soderland, Gagan Bansal, and Mausam. 2014. Hierarchical summarization: Scaling up multi-document summarization. In *Association for Computational Linguistics (ACL)*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint*, cs.CL/arXiv.2411.14199.

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 *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Google DeepMind. 2025. Gemini Deep Research.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2024. Length-controlled AlpacaEval: A simple way to debias automatic evaluators. *Conference on Language Modeling (CoLM)*.

Arpad E. Elo. 1966. *The USCF Rating System: Its Development, Theory, and Applications*. United States Chess Federation.

Seraphina Goldfarb-Tarrant, Alexander Robertson, Jasmina Lazic, Theodora Tsouloufi, Louise Donnison, and Karen Smyth. 2020. Scaling systematic literature reviews with machine learning pipelines. In *ACL Workshop on Scholarly Document Processing*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The Llama 3 herd of models. *arXiv preprint*, cs.AI/2407.21783.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *International Conference on Learning Representations (ICLR)*.

Peter Jansen, Samiah Hassan, and Ruoyao Wang. 2025. Matter-of-fact: A benchmark for verifying the feasibility of literature-supported claims in materials science. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Tetsu Kasanishi, Masaru Isonuma, Junichiro Mori, and Ichiro Sakata. 2023. SciReviewGen: A large-scale dataset for automatic literature review generation. In *Association for Computational Linguistics (ACL)*.

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Y. Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024. Prometheus 2: An open source language model specialized in evaluating other language models. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Kalpesh Krishna, Aurko Roy, and Mohit Iyyer. 2021. Hurdles to progress in long-form question answering. In *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Peder Larsen and Markus Von Ins. 2010. The rate of growth in scientific publication and the decline in coverage provided by science citation index. *Scientometrics*.

Yoonjoo Lee, Kyungjae Lee, Sunghyun Park, Dasol Hwang, Jaehyeon Kim, Hong-In Lee, and Moontae Lee. 2023. QASA: Advanced question answering on scientific articles. In *International Conference on Machine Learning (ICML)*.

Zhuohan Li, Eric Wallace, Sheng Shen, Kevin Lin, Kurt Keutzer, Dan Klein, and Joey Gonzalez. 2020. Train big, then compress: Rethinking model size for efficient training and inference of transformers. In *International Conference on Machine Learning (ICML)*.

Bill Y. Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. 2025. WildBench: Benchmarking LLMs with challenging tasks from real users in the wild. In *International Conference on Learning Representations (ICLR)*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG evaluation using GPT-4 with better human alignment. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Kyle Lo, Lucy L. Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The Semantic Scholar Open Research Corpus. In *Association for Computational Linguistics (ACL)*.

Chaitanya Malaviya, Priyanka Agrawal, Kuzman Ganchev, Pranesh Srinivasan, Fantine

Huot, Jonathan Berant, Mark Yatskar, Dipanjan Das, Mirella Lapata, and Chris Alberti. 2025. Dolomites: Domain-specific long-form methodical tasks. *Transactions of the Association for Computational Linguistics*.

Chaitanya Malaviya, Subin Lee, Sihao Chen, Elizabeth Sieber, Mark Yatskar, and Dan Roth. 2024. ExpertQA: Expert-curated questions and attributed answers. In *North American Chapter of the Association for Computational Linguistics (NAACL)*.

OpenAI. 2025. Introducing Deep Research.

Arjun Panickssery, Samuel Bowman, and Shi Feng. 2024. LLM evaluators recognize and favor their own generations. *Neural Information Processing Systems (NeurIPS)*.

Liana Patel, Negar Arabzadeh, Harshit Gupta, Ankita Sundar, Ion Stoica, Matei Zaharia, and Carlos Guestrin. 2025. DeepScholar-Bench: A live benchmark and automated evaluation for generative research synthesis. *NeurIPS 2025 Workshop on Evaluating the Evolving LLM Lifecycle: Benchmarks, Emergent Abilities, and Scaling*.

Perplexity. 2025. Introducing Perplexity Deep Research.

Long Phan, Alice Gatti, Ziwen Han, Nathaniel Li, Josephina Hu, Hugh Zhang, Chen Bo Calvin Zhang, Mohamed Shaaban, John Ling, Sean Shi, et al. 2025. Humanity's last exam. *arXiv preprint*, cs.LG/2501.14249.

Derek J. Price. 1963. *Little Science, Big Science*. Columbia University Press.

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 *Association for Computational Linguistics (ACL)*.

Yasaman Razeghi, Robert L. Logan IV, Matt Gardner, and Sameer Singh. 2022. Impact of pretraining term frequencies on few-shot numerical reasoning. In *Empirical Methods in Natural Language Processing (EMNLP)*.

David Rein, Betty L. Hou, Asa C. Stickland, Jackson Petty, Richard Y. Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. 2024. GPQA: A graduate-level Google-proof Q&A benchmark. In *Conference on Language Modeling (CoLM)*.

Tomohiro Sawada, Daniel Paleka, Alexander Havrilla, Pranav Tadepalli, Paula Vidas, Alexander Kranias, John J Nay, Kshitij Gupta, and Aran Komatsuzaki. 2025. ARB: Advanced reasoning benchmark for large language models. *NeurIPS MATH-AI Workshop*.

Lin Shi, Chiyu Ma, Wenhua Liang, Weicheng Ma, and Soroush Vosoughi. 2024. Judging the judges: A systematic investigation of position bias in pairwise comparative assessments by LLMs. *Asia-Pacific Chapter of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (AACL-IJCNLP)*.

Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. 2025. Can LLMs generate novel research ideas? A large-scale human study with 100+ NLP researchers. In *International Conference on Learning Representations (ICLR)*.

Amanpreet Singh, Joseph C. Chang, Chloe Anastasiades, Dany Haddad, Aakanksha Naik, Amber Tanaka, Angele Zamarron, Cecile Nguyen, Jena D. Hwang, Jason Dunkleberger, Matt Latzke, Smita Rao, Jaron Lochner, Rob Evans, Rodney Kinney, Daniel S. Weld, Doug Downey, and Sergey Feldman. 2025. AI2 Scholar QA: Organized literature synthesis with attribution. *Association for Computational Linguistics (ACL)*.

Shruti Singh, Nandan Sarkar, and Arman Cohan. 2024. SciDQA: A deep reading comprehension dataset over scientific papers. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Prasann Singhal, Tanya Goyal, Jiacheng Xu, and Greg Durrett. 2024. A long way to go: Investigating length correlations in RLHF. In *Conference on Language Modeling (CoLM)*.

Michael D. Skarlinski, Sam Cox, Jon M. Laurent, James D. Braza, Michaela Hinks, Michael J. Hammerling, Manvitha Ponnapati, Samuel G.

Rodriques, and Andrew D. White. 2024. Language agents achieve superhuman synthesis of scientific knowledge. *arXiv preprint*, cs.CL/2409.13740.

Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, Johannes Heidecke, Amelia Glaese, and Tejal Patwardhan. 2025. PaperBench: Evaluating AI's ability to replicate AI research. In *International Conference on Machine Learning (ICML)*.

Manya Wadhwa, Zayne Sprague, Chaitanya Malaviya, Philippe Laban, Junyi Jessy Li, and Greg Durrett. 2025. EvalAgent: Discovering implicit evaluation criteria from the web. *Conference on Language Modeling (CoLM)*.

Yuwei Wan, Yixuan Liu, Aswathy Ajith, Clara Grazian, Bram Hoex, Wenjie Zhang, Chunyu Kit, Tong Xie, and Ian Foster. 2024. SciQAG: A framework for auto-generated science question answering dataset with fine-grained evaluation. *arXiv preprint*, cs.CL/2405.09939.

Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Min zhang, Qingsong Wen, Wei Ye, Shikun Zhang, and Yue Zhang. 2024. AutoSurvey: Large language models can automatically write surveys. *Neural Information Processing Systems (NeurIPS)*.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. 2025. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. *Neural Information Processing Systems (NeurIPS)*.

Koki Wataoka, Tsubasa Takahashi, and Ryokan Ri. 2024. Self-preference bias in LLM-as-a-judge. In *NeurIPS Safe Generative AI Workshop*.

Tomer Wolfson, Harsh Trivedi, Mor Geva, Yoav Goldberg, Dan Roth, Tushar Khot, Ashish Sabharwal, and Reut Tsarfaty. 2025. MoNaCo: More natural and complex questions for reasoning across dozens of documents. *Transactions of the Association for Computational Linguistics*.

Yating Wu, Ritika Mangla, Greg Durrett, and Junyi Jessy Li. 2023. QUDeval: The evaluation of questions under discussion discourse parsing. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Fangyuan Xu, Kyle Lo, Luca Soldaini, Bailey Kuehl, Eunsol Choi, and David Wadden. 2024. KIWI: A dataset of knowledge-intensive writing instructions for answering research questions. In *Association for Computational Linguistics (ACL)*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. 2025. Qwen3 technical report. *arXiv preprint*, cs.CL/2505.09388.

Zonglin Yang, Xinya Du, Junxian Li, Jie Zheng, Soujanya Poria, and Erik Cambria. 2024. Large language models for automated open-domain scientific hypotheses discovery. In *Association for Computational Linguistics (ACL)*.

Yilun Zhao, Kaiyan Zhang, Tiansheng Hu, Sihong Wu, Ronan Le Bras, Charles McGrady, Taira Anderson, Jonathan Bragg, Joseph Chee Chang, Jesse Dodge, Matt Latzke, Yixin Liu, Xiangru Tang, Zihang Wang, Chen Zhao, Hannaneh Hajishirzi, Doug Downey, and Arman Cohan. 2025. SciArena: An open evaluation platform for foundation models in scientific literature tasks. *Neural Information Processing Systems (NeurIPS)*.

Yuxiang Zhou, Jiazheng Li, Yanzheng Xiang, Hanqi Yan, Lin Gui, and Yulan He. 2024. The mystery of in-context learning: A comprehensive survey on interpretation and analysis. In *Empirical Methods in Natural Language Processing (EMNLP)*.

## A Additional Details of Dataset Pipeline

Below, we describe the data pipeline (§2) details.

### A.1 Query generation from survey articles

**Curating a list of top publication venues for 257 research fields.** We identify the top-20 publishing venues for each of 257 research fields listed on Google Scholar Metrics (total venues = 4634). The aggregate counts are presented in Table 7.

| Field | Fields | Venues |
|---|---|---|
| Business, Economics, & Management | 16 | 285 |
| Chemical & Material Sciences | 17 | 318 |
| Physics & Mathematics | 21 | 370 |
| Humanities, Literature, & Arts | 26 | 481 |
| Life Sciences | 30 | 509 |
| Social Sciences | 31 | 526 |
| Engineering & Computer Science | 50 | 925 |
| Health & Medical Sciences | 66 | 1220 |
| Total | 257 | 4634 |

Table 7: Aggregated numbers for top-20 publishing venues in Google Scholar categories.

**Retrieving survey articles from each publication venue.** We retrieve articles from three datastores: Crossref, the Semantic Scholar API, and S2ORC (Lo et al., 2020). Articles are queried with the following keywords: *survey*, *literature review*, *a review*, *an overview*, and *meta-analysis*. The article retrieval yields 615K article weblinks, where 134K are downloadable full-text articles.

**Removing non-survey articles.** We use $\mathcal{M}$ to classify between actual literature reviews and articles that mention a keyword like *survey* in its title. Classifier metrics are F1=.80, Prec=.87, Rec=.75 on an author-annotated validation set. The final yield is 54K survey articles across 254 research fields (3 research fields do not yield survey articles using our method).

**Selecting sections from survey articles.** We apply a sequence of filters to select sections (total = 886K, yield = 319K) from survey articles:

- **Title passes keyword blacklist**: We ignore abstract, introduction, or other summary sections, removing any sections that contain the following words: *question*, *survey*, *abstract*, *introduction*, *contribution*, *related*, *result*, *discussion*, *conclusion*, *limitation*, *appendix*, *appendices*, *appendixes*, *supplementary*, *supplemental*, *supplement*, *material*, *acknowledgement*, *future*, *direction*, *summary*, *suggestion*, *table*, *tab.*, *tbl.*, *figure*, *fig.*, and *plot*.
- **Section length is not too short ($\geq$ 3 sentences, $\geq$ 800 characters) and not too long ($\leq$ 300K characters)**: We apply a basic length filter to ensure that queries are generated from substantial, but focused, sections.
- **Minimum number of in-text citations ($\geq$ 3)**: We select sections that are well grounded in the literature, where each section must have 3 or more citations.

**Generating queries from survey sections.** We extract a date cutoff $D$ from each section's article metadata, and we use $\mathcal{M}$ to parse the section content into a hierarchical summary to generate an initial query and initial reference answer $(Q_{\text{initial}}, \hat{A}_{\text{initial}})$. To select high quality queries, all queries (total = 319K, yield = 21K) are filtered:

- **Initial query is self-contained**: $\mathcal{M}$ assigns $Q_{\text{initial}}$ a self-containment score from 1 to 10, where higher scores indicate more self-containment. We remove queries with $< 7$.
- **Initial query has low answer variability**: $\mathcal{M}$ assigns $Q_{\text{initial}}$ an answer variability score from 1 to 10, where higher scores indicate likely answer variability (e.g., expert disagreement or subjectivity). We remove queries with $> 4$.
- **(Rephrasing Step) Query and reference answer cohesion**: $\mathcal{M}$ rephrases $(Q_{\text{initial}}, \hat{A}_{\text{initial}})$ to queries $Q$ and reference answers $\hat{A}$ to improve their cohesion. We do not remove any queries at this step.
- **Query does not contain a citation**: $\mathcal{M}$ detects whether a citation is in the query $Q$.
- **Reference answer length is long enough ($\geq$ 800 characters)**: Reference answers $\hat{A}$ substantiate a semi open-ended query.
- **Final query is self-contained**: After rephrasing, some queries are still not standalone, so we apply a keyword-based method to remove non-standalone queries. Keywords include past tense auxiliary verbs (*did*, *was*, *were*) or words in referring expressions (*questionnaire*, *literature*).

| Research Domain and Field | # Queries |
|---|---|
| **Health Sciences & Medicine** | **7454** |
| ***Specialized Medicine*** | ***3968*** |
| Surgery | 673 |
| Pharmacology & Pharmacy | 630 |
| Dentistry | 380 |
| Oncology | 369 |
| Veterinary Medicine | 332 |
| Pediatric Medicine | 294 |
| Emergency Medicine | 232 |
| Psychiatry | 208 |
| Dermatology | 189 |
| Ophthalmology & Optometry | 144 |
| Developmental Disabilities | 138 |
| Anesthesiology | 137 |
| Otolaryngology | 101 |
| Nuclear Medicine, Radiotherapy & Molecular Imaging | 89 |
| Neuroscience | 52 |
| ***Preventive Medicine*** | ***2151*** |
| Nutrition Science | 1410 |
| Physical Education & Sports Medicine | 327 |
| Epidemiology | 156 |
| Alternative & Traditional Medicine | 140 |
| Tropical Medicine & Parasitology | 118 |
| ***Internal Medicine & Chronic Diseases*** | ***1070*** |
| Toxicology | 196 |
| Communicable Diseases | 182 |
| Endocrinology | 124 |
| Diabetes | 118 |
| Hematology | 110 |
| Urology & Nephrology | 84 |
| Pulmonology | 76 |
| Rheumatology | 76 |
| Gastroenterology & Hepatology | 54 |
| Vascular Medicine | 50 |
| ***Reproductive Medicine*** | ***265*** |
| Reproductive Health | 95 |
| Gynecology & Obstetrics | 91 |
| Pregnancy & Childbirth | 79 |
| **Life & Earth Sciences** | **4928** |
| ***Life Sciences*** | ***3528*** |
| Agronomy & Crop Science | 780 |
| Forests & Forestry | 674 |
| Biochemistry | 588 |
| Microbiology | 413 |
| Botany | 294 |
| Animal Biology & Behavior | 224 |
| Cell Biology | 190 |
| Marine Sciences & Fisheries | 188 |
| Mycology | 94 |
| Ecology | 83 |

| Research Domain and Field (continued) | # Queries |
|---|---|
| ***Earth Sciences*** | ***1400*** |
| Sustainable Energy | 466 |
| Hydrology & Water Resources | 400 |
| Geology | 358 |
| Geochemistry & Mineralogy | 107 |
| Atmospheric Sciences | 69 |
| **Engineering & Computer Science** | **4676** |
| ***Engineering*** | ***3417*** |
| Materials Engineering | 1944 |
| Manufacturing & Machinery | 329 |
| Electrical Engineering | 274 |
| Biomedical Engineering | 266 |
| Architecture | 240 |
| Environmental & Geological Engineering | 194 |
| Ocean & Marine Engineering | 88 |
| Mechanical Engineering | 82 |
| ***Computer Science*** | ***1259*** |
| Artificial Intelligence | 509 |
| Signal Processing | 274 |
| Natural Language Processing | 230 |
| Computer Vision & Pattern Recognition | 193 |
| Computing Systems | 53 |
| **Physical Sciences** | **2544** |
| ***Material Sciences*** | ***1320*** |
| Polymers & Plastics | 1094 |
| Composite Materials | 226 |
| ***Chemistry*** | ***1153*** |
| Analytical Chemistry | 749 |
| Chemical Kinetics & Catalysis | 212 |
| Dispersion Chemistry | 134 |
| Crystallography & Structural Chemistry | 58 |
| ***Physics*** | ***71*** |
| Physics | 71 |
| **Social Sciences** | **1395** |
| Sustainable Development | 600 |
| Public Health | 566 |
| Education | 229 |
| **Humanities** | **362** |
| ***Cognitive Studies*** | ***301*** |
| Psychology | 250 |
| Language & Linguistics | 51 |
| ***Anthropology*** | ***61*** |
| Anthropology | 61 |
| **Economics** | **55** |
| Economics | 55 |

Table 8: The number of queries for each of 75 fields, by research domain, in RESEARCHQA.

- **Final query is aligned with field**: Venues can be multidisciplinary, and therefore some queries are unrelated to the field those venues are mapped to. To ensure field alignment, $\mathcal{M}$ clasifies and removes queries that are unrelated to the field.
- **Queries are from a field with $\geq$ 50 queries**: Queries should cover important aspects of a field, which is difficult to accomplish if there are not enough queries. We set 50 to be the minimum query count to call a field sufficiently covered by queries, which leaves 21K queries $Q$ that cover 75 research fields.

| Type | gpt-4.1-mini | gpt-4.1 |
|---|---|---|
| $Q$ | What is the rule-based approach in machine translation, and what are its benefits and limitations? | How do rule-based machine translation systems use linguistic analysis and dictionaries to carry out translations? |
| $R_1$ | Does the response provide a detailed explanation of the linguistic analysis process involved in the Rule-based Approach, including morphology, syntax, and lexical semantics? | Does the response explain how linguistic analysis at multiple levels (morphology, syntax, lexical semantics) is used in rule-based machine translation? |
| $R_2$ | Does the response discuss the flexibility of the Rule-based Approach in adapting to new language constructs by updating the dictionary? | Does the response describe the use of bilingual dictionaries for mapping source language words to target language equivalents? |
| $R_3$ | Does the response highlight the advantage of the Rule-based Approach in not requiring as many parallel sentence pairs as Neural Machine Translation? | Does the response discuss how RBMT systems handle word ambiguities and parts of speech during translation? |
| $R_4$ | Does the response mention the use of dictionaries and expert knowledge in establishing grammar rules for the Rule-based Approach? | Does the response explore in depth how linguistic analysis at multiple levels (morphology, syntax, lexical semantics) is performed and integrated in rule-based machine translation systems? |
| $R_5$ | Does the response include a definition or explanation of what a Rule-based Approach in machine translation is? | Does the response provide a detailed explanation of how bilingual dictionaries are structured and utilized within the translation process? |
| $R_6$ | Does the response mention specific benefits of using a Rule-based Approach in machine translation? | Does the response provide a detailed explanation of how linguistic analysis components are integrated into the translation process? |
| $R_7$ | Does the response highlight any limitations or challenges associated with the Rule-based Approach in machine translation? | Does the response explore in depth the role and structure of dictionaries used in rule-based machine translation? |
| $R_8$ | Does the response cite the paper by Scott and Barreiro (2009) (title: OpenLogos MT and the SAL representation language) that provides insights into the linguistic analysis and dictionary mapping process in Rule-based Machine Translation? | Does the response cite papers like Kay (1980) (title: The Proper Place of Men and Machines in Language Translation) that discuss the integration of dictionaries with linguistic rules for accurate translation? |

Table 9: Comparisons between when pipeline models are used as generators. Generally, queries and rubrics are comparable in quality, where those from `gpt-4.1` are slightly more specific than those from `gpt-4.1-mini`.

### A.2 Rubric generation from survey articles

We create rubrics with $\mathcal{M}$. To diversify rubric item types, three generation prompts are used—information, depth, and citation. These rubric types are intended to diversify the types of rubric items that are generated, whereas rubric items have been categorized post-hoc to more granular labels for analysis.

- **Information-based rubric item**: A binary yes/no question asking whether an answer addresses a specific statement, finding, opinion, or comparison. Rubric items of this category allow for broad analysis of quality.

  *Example: Does the response address the specific benchmarks where auto-regressive LMs outperform bi-directional encoders?*

- **Depth-based rubric item**: A binary yes/no question asking whether an answer elaborates or explains a topic. Rubric items of this category allow measuring analysis, discussion, and explanation in answers.

  *Example: Does the response elaborate on the hybrid approach of GRIT in unifying auto-regressive and bi-directional features?*

- **Citation-based rubric item**: A binary yes/no question asking whether or not an answer cites a specific study. Rubric items of this category evaluate answer grounding in the literature.

  *Example: Does the response cite papers such as Wang et al. (2023) (title: Improving text embeddings with large language models) and the MTEB paper (title: Massive Text Embedding Benchmark) that show the performance of auto-regressive LMs surpassing bi-directional encoders in retrieval tasks?*

In addition to automatically generated rubrics, we explore generic rubrics using evaluation criteria from related work (Asai et al., 2024; Zhao et al., 2025). Generic rubric items are in Table 11.

**Remapping research fields and domains.** For the purposes of evaluation, some fields are overly specific (e.g., Wood Science & Technology) or overly general (e.g., Health & Medical Sciences

| Model | Queries | Rubrics | Total |
|---|---|---|---|
| gpt-4.1-mini | $ 349.65 | $ 133.65 | $ 483.30 |
| gpt-4.1 | $ 1748.25 | $ 668.25 | $ 2416.50 |
| gpt-4o | $ 2185.31 | $ 835.31 | $ 3020.62 |

Table 10: Cost comparison for different pipeline models for the full generation of RESEARCHQA. All numbers are in batched inference pricing (2× cheaper).

(general)). Additionally, the Physics & Mathematics branch does not yield enough queries to sufficiently represent a domain. For these reasons, we redistribute data to match a more intuitive hierarchical structure. In total, we merge 170 of 257 fields and introduce 7 more: Animal Biology & Behavior, Biomedical Engineering, Electrical Engineering, Management, Mathematics, Physics, and Neuroscience. These changes result in a hierarchy covering 94 fields and 7 domains. Our pipeline is able to sufficiently cover 75 of 94 fields (where each of the 75 fields has $\geq 50$ queries) using the available datastores. A full list of fields is in tabulated in Table 8.

## B Pipeline Model Analysis

In this section, we discuss the performances and potential costs of different models $\mathcal{M}$ used in the pipeline to create RESEARCHQA. We consider three state-of-the-art models at the time of consideration in Table 10.

**$\mathcal{M}$ as a classifier.** We use $\mathcal{M}$ as a classifier for quality filtering, such as removing non-literature reviews, ambiguous queries, open-ended queries, and removing queries with citations. For each filtering task, we find that the smallest model, gpt-4.1-mini, obtains at least .80 F1 compared against author annotations.

**$\mathcal{M}$ as a generator.** We show sample outputs from using gpt-4.1-mini and gpt-4.1 in Table 9. We find that queries and rubrics generated by the two models are comparable in quality and content. Queries generated by gpt-4.1 are more specific, such as asking explicitly how linguistics and dictionaries are applied in rule-based machine translation; by contrast, gpt-4.1-mini asks for an open-ended explanation of rule-based machine translation with benefits and limitations. These differences propagate to the rubrics, with those from gpt-4.1 judging by discussion of word ambiguities and parts-of-speech whereas those from gpt-4.1-mini judge by data efficiency in rule-based methods ($R_3$). Overall, we find content similarities between model outputs with small differences in the fine-grained evaluation criteria.

**Costs.** Costs for batched inference of $\mathcal{M}$ are in Table 10. Notably, gpt-4.1 is ∼$2K more expensive than gpt-4.1-mini. Given performance similarities, we choose gpt-4.1-mini for scaling out RESEARCHQA.

| Type | Generic Rubric Item |
|---|---|
| Correctness | Does the response make factually correct statements? |
| Citations | Does the response cite relevant papers? |
| Coverage | Does the response provide sufficient coverage and amount of information? |
| Relevance | Does the response stay on topic and maintain a clear focus? |
| Organization | Does the response have an organized structure? |
| Usefulness | Does the response fulfill information needs? |
| Attribution | Does the response make correct citation-to-claim attributions? |
| Examples | Does the response include relevant examples? |

Table 11: The generic rubric baseline, sampled from ScholarQABench (Asai et al., 2024) and SciArena (Zhao et al., 2025) criteria. "Examples" is manually written to fill the rubric to 8 items.

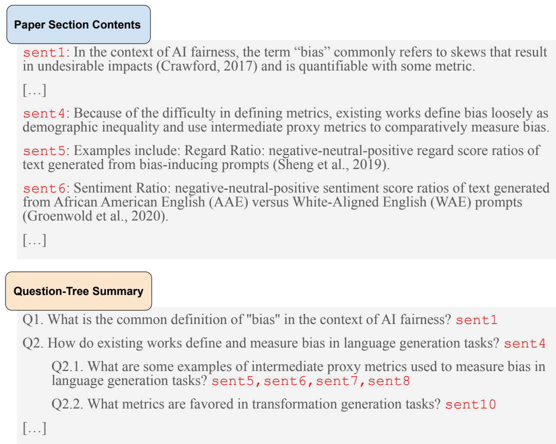


Figure 7: Hierarchical summary of section content, motivated by hierarchical summarization (Christensen et al., 2014) and question-under-discussion parsing (Benz and Jasinskaja, 2017; Wu et al., 2023).

| System | Model Card | Additional settings | Cost |
|---|---|---|---|
| **Parametric** | | | |
| llama-3.3-70b | llama-3.3-70b-instruct | | $ 5 |
| claude-4-sonnet | claude-sonnet-4-20250514 | | $ 24 |
| gpt-4.1 | gpt-4.1 | | $ 15 |
| qwen-3-32b | qwen3-32b | | $ 5 |
| gemini-2.5-pro | gemini-2.5-pro-preview-06-05 | | $ 15 |
| **Retrieval (Naive)** | | | |
| openscholar-8b | llama-3.1_openscholar-8b | [Embed] openscholar-retriever [Reranker] openscholar-reranker | $ 0 |
| qwen-3-32b | qwen3-32b | [Embed] gte-qwen-2-7b-instruct | $ 10 |
| gemini-2.5-pro | gemini-2.5-pro-preview-06-05 | [Embed] text-embedding-004 | $ 35 |
| claude-4-sonnet | claude-sonnet-4-20250514 | [Embed] voyage-3-large | $ 218 |
| gpt-4.1 | gpt-4.1 | [Embed] text-embedding-3-large | $ 151 |
| **Retrieval (Production)** | | | |
| sonar | sonar | | $ 19 |
| openscholar-8b+feedback | llama-3.1_openscholar-8b | | $ 100 |
| sonar-reasoning | sonar-reasoning | | $ 38 |
| gpt-4o-search-preview | gpt-4o-search-preview | search_context_size=medium | $ 150 |
| gemini-2.5-pro+grounding | gemini-2.5-pro-preview-06-05 | dynamic_threshold=0.3 | $ 30 |
| claude-4-sonnet+ws | claude-sonnet-4-20250514 | tools.type=web_search_20250305 max_uses=3 | $ 450 |
| **Deep Research** | | | |
| o4-mini-deep-research | o4-mini-deep-research | tools.type=web_search_preview reasoning.summary=auto | $4200 |
| sonar-deep-research | sonar-deep-research | | $1500 |

Table 12: **LLM System Configuration and Cost.** Default setting: `temperature=0`. Cost is for running on the full test set of RESEARCHQA (3750 queries). Organizations and related works: Meta (Grattafiori et al., 2024), Anthropic (Anthropic, 2024), OpenAI (Achiam et al., 2023; OpenAI, 2025), Alibaba (Yang et al., 2025), Google (Comanici et al., 2025; DeepMind, 2025), AI2 (Asai et al., 2024), Perplexity (Perplexity, 2025).

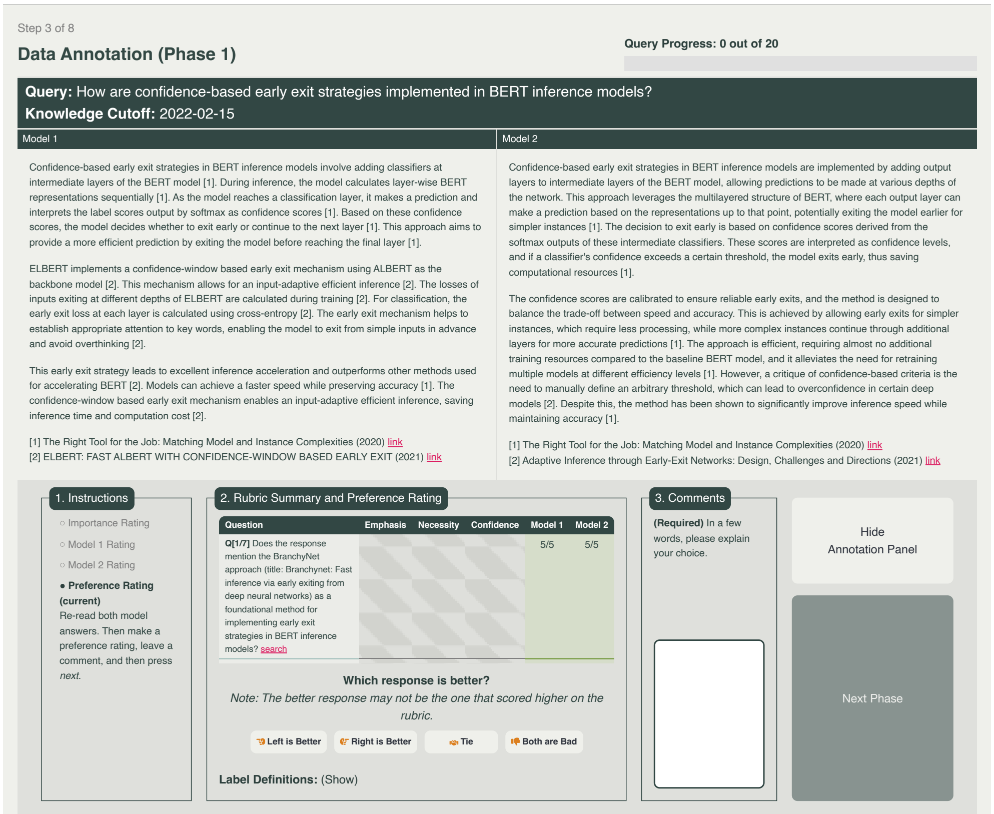


Figure 8: RESEARCHQA human evaluation interface.

**Q:** What strategies are recent research efforts employing to reduce the wall-clock training time for BERT models? (Date: 2020-10-02)

`[Parametric]` **gpt-4.1:**

Recent research efforts have focused on several strategies to reduce the wall-clock training time for BERT models, primarily **by optimizing both the model architecture and the training process**. One prominent approach is the use of **mixed-precision training**, which leverages lower-precision arithmetic (such as FP16) to accelerate computation and reduce memory usage without significantly impacting model accuracy. This technique enables **larger batch sizes** and faster throughput on modern GPUs, leading to substantial reductions in training time [1][2]. Another effective strategy is **distributed training**, where the workload is split across multiple GPUs or even multiple nodes. Techniques such as data parallelism and model parallelism allow for efficient scaling, and recent advancements in communication algorithms, such as **gradient accumulation and all-reduce optimizations**, have further minimized synchronization overhead [1][3].

Additionally, researchers have explored architectural modifications to BERT itself. For example, models like ALBERT employ **parameter sharing and factorized embedding parameterization**, which reduce the number of parameters and computational requirements, thereby speeding up training [2]. **Knowledge distillation** is another promising avenue, where a smaller "student" model is trained to mimic the behavior of a larger "teacher" BERT model, resulting in faster training and inference while retaining much of the original performance [3]. Furthermore, techniques such as **dynamic masking** and more efficient data pipelines have been introduced to streamline the pre-training process, reducing redundant computation and improving hardware utilization [1][2]. Collectively, these strategies have enabled significant reductions in wall-clock training time for BERT models, making large-scale pre-training more accessible and cost-effective.

**[1] Large Batch Optimization for Deep Learning: Training BERT in 76 minutes (2019)**
[2] ALBERT: A Lite BERT for Self-supervised Learning of Language Representations (2019)
[3] DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter (2019)

`[Retrieval (Naive)]` **gpt-4.1:**

Recent research in reducing the wall-clock training time for BERT and Transformer-based models has focused on several key strategies that balance computational efficiency with model performance. **One prominent approach is to leverage larger models—wider and deeper architectures—which, despite having higher per-iteration computational costs, converge in significantly fewer training steps.** This acceleration in convergence often outweighs the additional overhead, making it more compute-efficient to train very large models for a smaller number of iterations rather than training smaller models for longer periods [1].

Another effective strategy is to combine this "train big" approach with aggressive model compression techniques such as **quantization and pruning**. Large models, once trained, are shown to be more robust to these compression methods than their smaller counterparts. As a result, heavily compressed large models can achieve higher accuracy and inference efficiency than lightly compressed small models, effectively providing the benefits of both rapid training and efficient deployment [1].

Additionally, research has highlighted the importance of **optimizing hyperparameters and training regimes**. For example, studies have found that BERT was originally undertrained and that careful tuning of training duration, batch size, and learning rates can yield substantial improvements in both training speed and final model quality [2]. These insights suggest that, beyond architectural changes, meticulous attention to training procedures can further reduce wall-clock time without sacrificing performance.

In summary, **the most effective recent strategies involve training larger models for fewer steps**, followed by robust compression, and optimizing training protocols—collectively enabling faster, more efficient BERT training and deployment [1], [2].

[1] Train big, then compress: Rethinking model size for efficient training and inference of transformers (2020)
[2] Roberta: A robustly optimized bert pretraining approach (2019)

| **Rubric Item** | **Left** | **Right** |
|---|---|---|
| Does the response cite the paper by You et al. (2019) (title: **Large batch optimization for deep learning: Training bert in 76 minutes**) that introduces the LAMB optimizer to significantly reduce BERT training time? | 3/4 | 0/4 |
| Does the response provide a detailed explanation of the LAMB optimizer and its impact on reducing BERT training time? | 1/4 | 0/4 |
| Does the response mention the use of **larger batch sizes** as a method to reduce training time? | 4/4 | 1/4 |
| Does the response include **strategies that focus on increasing computational efficiency** for reducing BERT training time? | 4/4 | 4/4 |
| Does the response include examples of **specific optimization techniques** used to reduce training time for BERT models? | 4/4 | 3/4 |
| Does the response highlight the use of hardware accelerators, such as GPUs or TPUs, in reducing training time? | 3/4 | 0/4 |
| Does the response mention the role of **distributed training strategies** in reducing wall-clock time? | 4/4 | 0/4 |
| Does the response explore in detail the various optimization techniques used to reduce wall-clock training time for BERT models? | 3/4 | 3/4 |

Figure 9: Retrieval-augmented responses can overly commit to orthogonal context, losing to parametric responses. Green text is content mentioned in the rubrics; Orange text is content generally not in the rubrics.

**Q:** How does substrate temperature influence the crystal orientation and quality of AlN (002) thin films during sputtering? (Date: 2020-10-19)

`[Retrieval (Production)]` **sonar-reasoning:**

Substrate temperature significantly influences the crystallographic orientation and structural quality of AlN (002) thin films during sputtering, primarily through its impact on **atomic mobility and growth kinetics**. Higher temperatures (typically above 500-700°C) enable epitaxial growth by providing sufficient energy for atoms to arrange into ordered crystal structures[2], [3]. At 700°C, AlN films grown on Si(111) substrates exhibit epitaxial alignment, while lower temperatures (300–600°C) result in **fiber-textured morphology due to incomplete atomic ordering**[2]. This orientation transition correlates with **improved crystallinity**, as higher temperatures reduce defects like oxygen impurities and sub-bandgap absorption[2], [4].

**Crystal Quality**
- **Defect Density**: Elevated temperatures decrease oxygen incorporation and defect states, **enhancing thermal conductivity and optical transparency**[2], [4].
- **Surface Morphology**: Atomic force microscopy (AFM) reveals smoother surfaces at 700°C, suggesting better atomic layer-by-layer growth[2].
- **Interfacial Control**: Higher temperatures strengthen interface quality, critical for piezoelectric applications, though excessive oxygen can degrade performance[2], [3].

**Practical Constraints**
While higher temperatures optimize (002) orientation and quality, process limitations (e.g., substrate material stability) often necessitate balancing growth conditions. **Low-temperature sputtering (<100°C) achieves functional AlN films but with smaller grain sizes and higher defect densities**[4].

[1] Impacts of substrate and growth temperature on the deposition and ... (2025)
[2] Influence of growth temperature on the properties of aluminum ... (2023)
[3] Reactive Sputtering of Aluminum Nitride (002) Thin Films for ... - MDPI
[4] [PDF] High Thermal Conductivity of Submicrometer Aluminum Nitride Thin ... (2024)

`[Deep Research]` **sonar-deep-research:**

Substrate temperature critically governs the crystalline orientation and quality of AlN (002) thin films during sputtering by modulating **adatom mobility**, **stress evolution**, and **defect dynamics**. At elevated temperatures (500-800°C), enhanced surface diffusion **enables Al and N atoms to preferentially align along the (002) c-axis orientation**, as the wurtzite structure's lowest surface energy plane minimizes strain[2], [4], [9]. For example, at 823K (550°C), optimized adatom rearrangement yields AlN (002) films with full-width half-maximum (FWHM) X-ray diffraction (XRD) rocking curves of 61 arcsec for (0002) reflections, indicating **high crystallinity**[2]. This occurs because thermal energy facilitates **dislocation annihilation and reduces point defect densities**, lowering oxygen incorporation to ~$7\times10^{20}$ $cm^{-3}$[4].

However, excessive temperatures (>900°C) induce compressive stress from **thermal expansion mismatches** with substrates like sapphire, causing **lattice elongation** (c-axis expansion up to 4.98 Å)[2], [5]. Such stress promotes polycrystalline growth or **film delamination, degrading (002) texture**. Intermediate temperatures (300–500°C) balance stress mitigation and crystallinity: at 400°C, AlN on Si(111) achieves epitaxial (002) growth with 0.346° FWHM, while 700°C yields fiber-textured films due to increased in-plane disorder[4], [10]. Lower temperatures (<300°C) limit adatom mobility, increasing **defect densities and residual stress (up to 1.2 GPa), which broadens XRD peaks** (FWHM >0.62°)[3], [7], [12].

**Optimal (002) alignment requires substrate-specific thermal regimes. For polymer substrates, temperatures ≤300°C** prevent deformation while maintaining (002) preference through kinetic control via higher sputtering power (414 W)[3]. Conversely, **rigid substrates (sapphire, Si) tolerate higher temperatures (600–800°C)** that enhance grain coalescence, reducing tilt/twist dislocations and improving piezoelectric response[9], [11]. Thus, substrate temperature acts synergistically with sputtering power and gas chemistry to achieve high-quality (002)-oriented AlN, balancing crystallinity and stress.

**(... citations truncated ...)**

| **Rubric Item** | **Left** | **Right** |
|---|---|---|
| Does the response explain the effect of increasing substrate temperature on the **atomic mobility** and its role in promoting (002) crystal orientation? | 4/4 | 4/4 |
| Does the response identify an **optimal substrate temperature range** (around 350 to 430 °C) for achieving the best (002) orientation and film quality? | 0/4 | 3/4 |
| Does the response discuss **the decline in (002) peak intensity and film quality** when the substrate temperature exceeds this optimal range? | 0/4 | 3/4 |
| Does the response mention **thermal stresses from mismatched thermal expansion coefficients** as a factor affecting film quality at high substrate temperatures? | 0/4 | 4/4 |
| Does the response explore the relationship between substrate temperature and crystal orientation of AlN (002) films in detail **to explain the underlying physical mechanisms**? | 4/4 | 4/4 |
| Does the response provide an in-depth discussion on the degradation of film quality beyond the optimal substrate temperature range, including factors like thermal stress and impurity incorporation? | 0/4 | 3/4 |

Figure 10: Deep research responses cover more concepts in greater detail. Green text is content mentioned explicitly in the rubrics; Orange text is content generally not explicit in the rubrics.

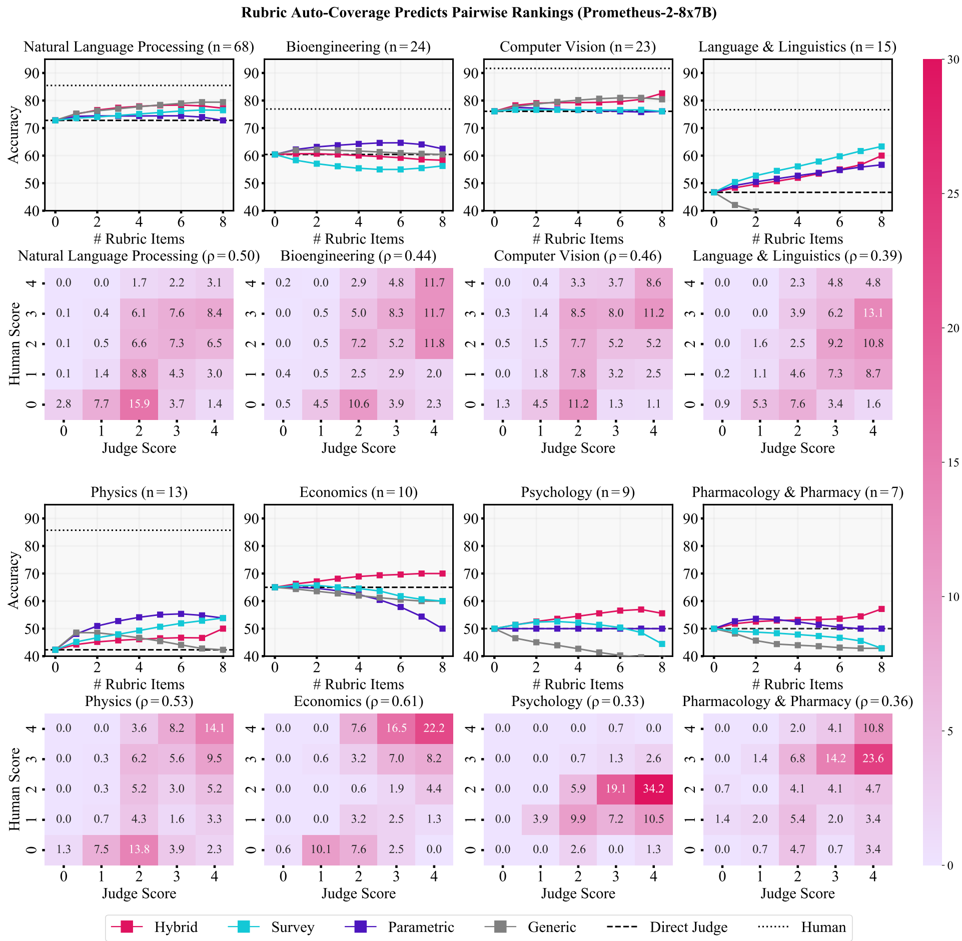


Figure 11: **Rubrics used in ensemble judges can help to predict expert-labeled pairwise preference**. Each graph represents fields covered by expert annotators. Heatmaps show confusion matrices of LLM and human rubric coverage scores. $n$ indicates the count of binary majority labels. In 7 out of 8 fields, hybrid rubrics can improve accuracy to majority labels, with hybrid rubrics obtaining highest performance in 4 out of 8.

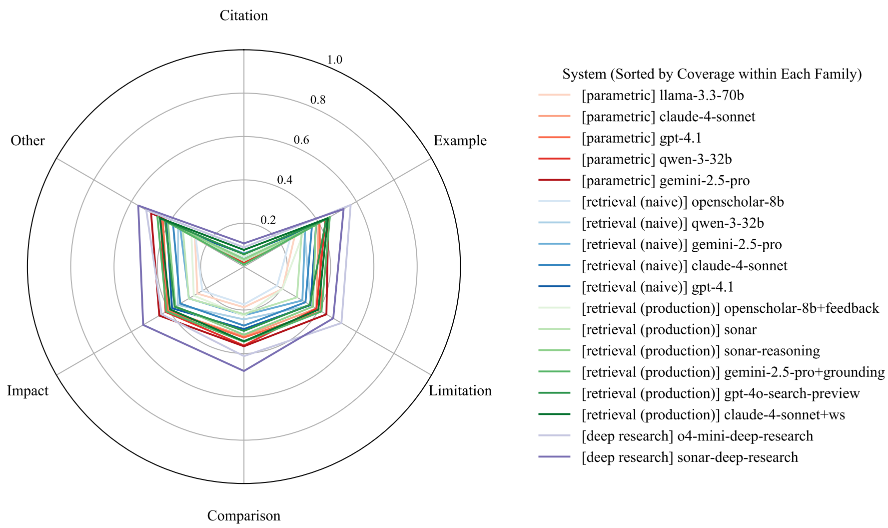


Figure 12: **Comparison of LLM System Performance by Rubric Type (Radar).** Performance is measured as the percentage of fully covered rubrics. Each rubric type is represented as an objective, with `sonar-deep-research` forming the Pareto frontier. “DR” denotes “Deep Research.”

| System | Leakage % | Coverage % | | |
|---|---|---|---|---|
| | | ¬ Leaked | Leaked | Δ |
| Parametric | | | | |
| `llama-3.3-70b` | 3 | **53.49** | 51.11 | −2.4 |
| `claude-4-sonnet` | 4 | **64.44** | 61.21 | −3.2 |
| `gpt-4.1` | 8 | **65.91** | 60.18 | −5.7 |
| `qwen-3-32b` | 3 | **66.69** | 65.08 | −1.6 |
| `gemini-2.5-pro` | 5 | **69.18** | 62.04 | −7.1 |
| Retrieval (Naive) | | | | |
| `openscholar-8b` | 0 | 54.71 | — | — |
| `gemini-2.5-pro` | 0 | 59.92 | — | — |
| `qwen-3-32b` | 0 | 60.90 | — | — |
| `claude-4-sonnet` | 0 | 62.50 | — | — |
| `gpt-4.1` | 0 | 64.80 | — | — |
| Retrieval (Production) | | | | |
| `sonar` | 18 | **58.84** | 57.50 | −1.3 |
| `openscholar-8b+feedback` | 29 | 57.57 | **61.54** | +4.0 |
| `sonar-reasoning` | 22 | 64.21 | **64.77** | +0.6 |
| `gpt-4o-search-preview` | 27 | 65.30 | **67.81** | +2.5 |
| `gemini-2.5-pro+grounding` | 2 | **68.54** | 66.91 | −1.6 |
| `claude-4-sonnet+ws` | 30 | 68.30 | **71.27** | +3.0 |
| Deep Research | | | | |
| `o4-mini-deep-research` | 28 | 71.85 | **74.82** | +3.0 |
| `sonar-deep-research` | 21 | **75.51** | 74.46 | −1.1 |

Table 13: Retrieving the distilled survey does not provide large advantages toward higher Coverage %. Parametric systems have 3-8% leakage (L%); Retrieval (Naive) do not produce answers with leakage (likely because surveys are intentionally removed from their retrieval); Retrieval (Production) and Deep Research result in 20-30% leakage. Coverage % roughly stays the same with leakage (Leaked) and without leakage (¬ Leaked).

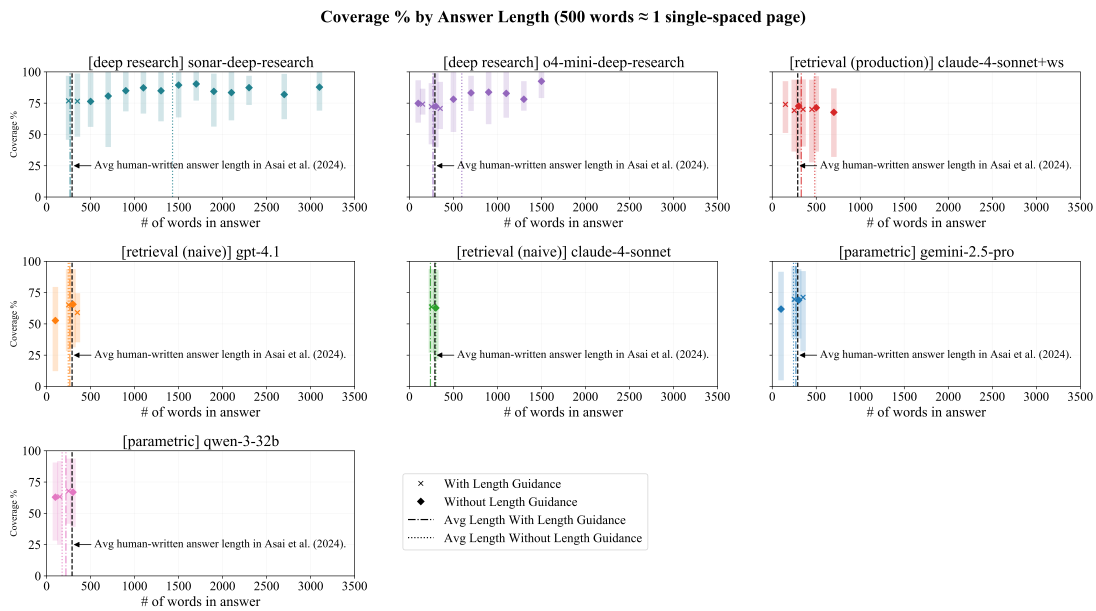


Figure 13: **Coverage % vs. Length.** The relation between Coverage % and answer length for the top performers in parametric, naive retrieval, and deep research categories selected by Coverage %, generating answers for 3 queries per field (225 total). Each subplot shows binned coverage across answer lengths, with markers distinguishing answers with vs. without length guidance. Vertical dashed lines indicate the average human-written answer length reported by Asai et al. (2024); and Error bars represent 95% confidence intervals from bootstrap resampling.