



# 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](https://huggingface.co/datasets/realliyifei/ResearchQA)  
 **Code:** [github.com/realliyifei/ResearchQA](https://github.com/realliyifei/ResearchQA)  
 **Website:** [cylumn.com/ResearchQA](https://cylumn.com/ResearchQA)

## Abstract

Evaluating long-form responses to research queries heavily relies on expert annotators, restricting attention to areas like AI where researchers can conveniently enlist colleagues. Yet, research expertise is widespread: survey articles synthesize knowledge distributed 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. Each rubric, derived jointly with queries from survey sections, lists query-specific answer evaluation criteria, i.e., citing papers, making explanations, and describing limitations. Assessments by 31 Ph.D. annotators in 8 fields indicate 96% of queries support Ph.D. information needs and 87% of rubric items should be addressed in system responses by a sentence or more. Using our rubrics, we are able to construct an automatic pairwise judge obtaining 74% agreement with expert judgments. We leverage RESEARCHQA to analyze competency gaps in 18 systems in over 7.6K pairwise evaluations. No parametric or retrieval-augmented system we evaluate exceeds 70% on covering rubric items, and the highest-ranking agentic 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

\*Equal contribution.

### Survey-Mined Query

How does the frequency of terms in pre-training data influence numerical reasoning performance in few-shot settings? ( Engineering)

**Research System** : The frequency of terms in pre-training data significantly influences a model's numerical reasoning performance, particularly in few-shot learning scenarios [1]. Models pre-trained [...]

[1] Scaling Laws and Data Frequency Effects in Large Language [...]

### Survey-Mined Evaluation Rubric

Does the response reference the “performance gap” concept from the Razeghi et al. (2022) paper [...]?

Does the response include **examples of studies or experiments** that investigate the impact of term frequency on numerical reasoning performance?

Does the response discuss the **correlation between the frequency of terms** in pre-training data and **numerical reasoning performance**?

Additional rubric items ...

### Judge

0/4 Not at all covered

4/4 Completely covered

1/4 Barely covered

Source Survey: The Mystery of In-Context Learning (Zhou et al., 2024)

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).

as deep research systems (DeepMind, 2025; OpenAI, 2025) and scientific AI assistants (Skarliniski 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 systems that synthesize long form answers to research queries is extremely challenging** (Xu et al., 2024). Several benchmarks have been proposed (i.e., Lee et al., 2023; Auer et al., 2023; Asai et al., 2024), 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 present **RESEARCHQA, a resource for evaluating model capabilities at answering research queries** that reduces the re-

quirement on large scale expert recruitment. RESEARCHQA is constructed using academic surveys, which are a unique resource presenting peer-reviewed summaries on a research topic (Kasani et al., 2023). We propose distilling queries and answer *evaluation rubrics* (Lin et al., 2025; Sawada et al., 2025) from surveys across diverse fields. Each rubric, derived jointly with queries by mining high quality survey sections, lists query-specific criteria that are important for evaluating answer quality, which may include citing papers, and making comparisons, or describing causal effects, among others (Figure 1). To demonstrate the utility and breadth of our method, we release RESEARCHQA, containing **21.4K queries with 160k rubric items, across 75 research fields**.

We validate the quality of RESEARCHQA queries and rubrics by recruiting 31 Ph.D. level annotators with expertise from 8 fields. 96% of queries are judged as reflecting information needs of Ph.D. students and 80% are stylistically similar to how a Ph.D. student would have expressed 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 allow us to better automate evaluation of system responses according to expert preferences. We collect pairwise preferences from annotators about retrieval augmented systems (Lewis et al., 2020) responding to queries in RESEARCHQA. Human-Human agreement is 84%, but GPT based judges that directly predict preference only agree with annotators 71% of the time, likely because they lack appropriate expert knowledge. Rubrics can be used to help reason about criteria derived from expert writing: as seen in Figure 1, LLMs can assess *rubric coverage*, the extent to which responses cover rubric items. Ensemble judges that select the answer maximizing the sum of rubric coverage and direct judgment scores reduce the LLM-Human and Human-Human agreement gap by 24% relative, improving automatic proxies to expert-labeled pairwise preferences.

Using RESEARCHQA, we evaluate 18 parametric, retrieval augmented, and agentic systems. We average rubric coverage % to measure answer quality on an absolute scale. No parametric or re-

trieval system we evaluate exceeds 70%. Among agentic systems we evaluate, the best is Perplexity’s deep research (Perplexity, 2025), exhibiting 75.29% coverage, showing room for improvement. Further, analysis grouping rubric items by criteria reveals that systems can improve across many skills: common error cases include citing key works (under-addressed in 89% of cases) and describing limitations (52% of cases).

With 75 research fields, RESEARCHQA allows for analysis of system competencies by field. Across all 18 systems, responses to Physical Sciences queries obtain higher coverage than those from other domains, exceeding Health Sciences by 3.2% and Humanities by 4.8% in rubric coverage on average. These differences underscore the need for multi-field evaluations.

Finally, using our ensemble evaluator, we rank all systems in a pairwise tournament on RESEARCHQA. Agentic systems designed for research synthesis far exceed other systems that are not specialized for the task, with the highest-ranking system showing 82% win rate over the next best system. These rankings demonstrate that, despite significant skill gaps demonstrated on absolute scales of rubric coverage, LLM tools specialized for research tasks are making incremental progress over off-the-shelf LLMs.

## 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.

To build RESEARCHQA, we create a multi-stage pipeline (Figure 2), which extracts top venues from research fields (§2.1); extracts survey articles from venues (§2.2); and generates queries (§2.3) and rubrics (§2.4) from survey contents. The pipeline uses a single LLM ( $\mathcal{M} \leftarrow \text{gpt-4.1-mini}$ ) throughout multiple stages for both data generation and filtering. Finally, we partition RESEARCHQA into train, validation, and
























| 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 <sup>†</sup> (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) (<sup>†</sup>Concurrent work). Scholarly fields are marked as blank “—” when field descriptions are incomplete or metadata are missing.

test splits (§2.5) to support model development and evaluation, with the test split designed for low-cost and reproducible benchmarking.

## 2.1 Extract Top Venues from Research Fields

Top publishing venues are likely to contain topics and comparisons broadly important to researchers in that field. To focus on high-quality survey articles, we identify the top-20 publishing venues ranked by h5-index for each field in Google Scholar.<sup>1</sup> Fields on Google Scholar can be overly specific (e.g., Wood Science & Technology) or overly general (e.g., Health & Medical Sciences (general)) for representing query topics, so we manually redistribute from 257 to 94 fields.<sup>2</sup> This process results in 660 unique venues that are used to retrieve survey articles.

## 2.2 Extract Survey Articles

We collect survey articles by performing keyword search from three sources: Crossref,<sup>3</sup> Semantic Scholar,<sup>4</sup> 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.** Survey articles are sourced from publications in one of 660 unique venues iden-

tified from Google Scholar, with titles that must include a specified keyword: *survey*, *literature review*, *a review*, *an overview*, and *meta-analysis*.

**Article filters.** Survey keywords can have multiple meanings (e.g., *survey* can be an analysis of focus groups or field observations) and thus do not guarantee the extracted articles are literature reviews. We prompt  $\mathcal{M}$  to classify and retain articles whose title corresponded to the description of literature reviews, i.e., works aimed at synthesizing and reviewing existing literature.

## 2.3 Generate Queries from Survey Content

We start by selecting content-rich survey sections with sufficient 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, without excessive variability in appropriate answers. In total, 319K of 886K sections are used, yielding 21K queries after filtering.

**Survey section filters.** Sections are removed if their titles suggest they are not part of the main body (e.g., *abstract*, *introduction*, *contribution*, *table*, *result*; see Appendix A for a complete list of keywords). Next, we exclude sections that are too short to be informative (< 3 sentences or < 800 characters), too long to process (> 300K characters), or lacking citations (< 3 inline citations).

**Query and reference answer generation.** We few-shot prompt  $\mathcal{M}$  to extract hierarchical summaries (Christensen et al., 2014) from section con-

<sup>1</sup><https://scholar.google.com>

<sup>2</sup>We merge 170 of 257 fields and introduce 7 more not originally present in Google Scholar.

<sup>3</sup><https://www.crossref.org/>

<sup>4</sup><https://www.semanticscholar.org>

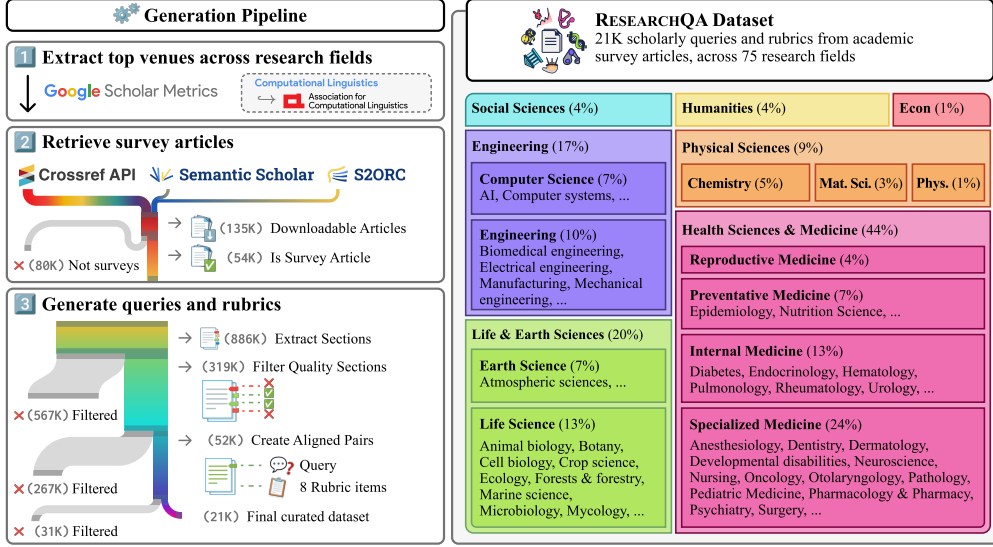


Figure 2: **(Left) RESEARCHQA generation stages:** We identify top-20 venues from each field in Google Scholar, retrieve survey articles from available databases, and generate queries and rubrics from survey sections. Throughout generation, we employ appropriate filtering mechanisms to ensure data quality. **(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.

tent. 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 10). We prompt  $\mathcal{M}$  with examples, generating a query and a reference answer using the summary and raw section content as input. Few-shot prompts ask that queries must be supported by at least 3 sentences and reference answers should 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 survey 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 standalone and answer variability of each query on a scale from 1 to 10. Queries scoring  $< 7$  for standalone or  $> 4$  for answer variability are removed.<sup>5</sup> 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 ( $< 800$  char-

acters). 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. We create rubrics from three sources: survey rubrics ( $\mathbf{R}_S$ ), which are conditioned on the query and reference answer; parametric rubrics ( $\mathbf{R}_P$ ), which are conditioned only on the query; and hybrid rubrics ( $\mathbf{R}_H$ ), which are the top- $n$  rubric items from a deduplicated and reranked union of  $\mathbf{R}_S$  and  $\mathbf{R}_P$ . Hybrid rubrics contain  $\sim 7.5$  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 items. 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.

<sup>5</sup>Thresholds validated by NLP experts on initial dev. data.



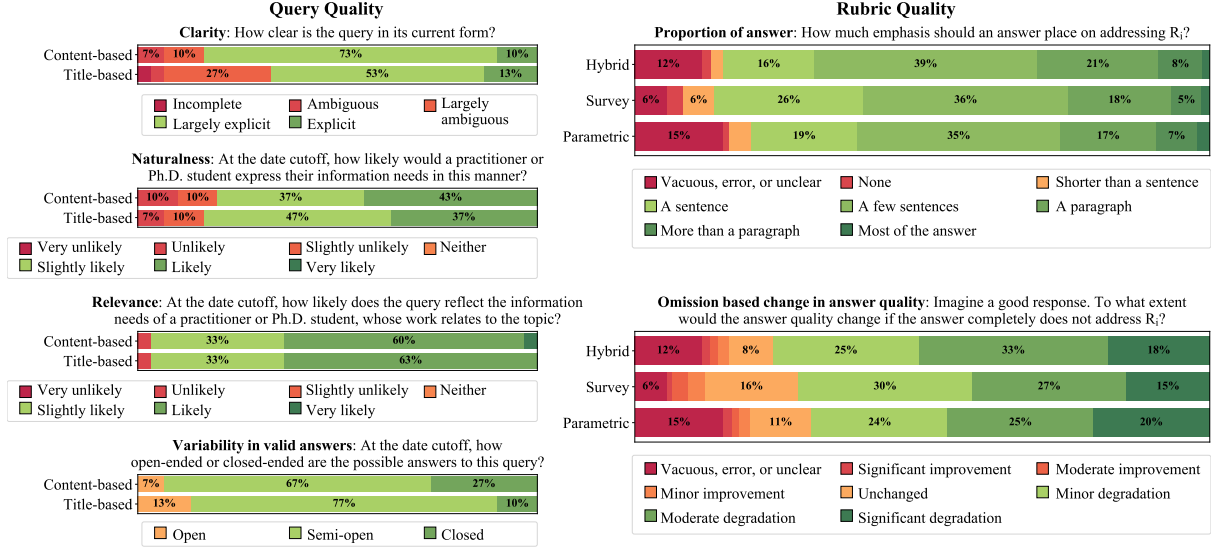


Figure 3: RESEARCHQA query and rubric quality ratings by 31 Ph.D. level experts.

**Hybrid rubric construction.** Inspired by prior work (Wadhwa et al., 2025), we create a hybrid rubric  $R_H$  that leverages the potential strengths of both  $R_P$  and  $R_S$  by filtering the reranked union of  $R_S$  and  $R_P$ . Using  $\mathcal{M}$ , we deduplicate repeating rubric items that represent identical criteria and rerank rubric items based on their importance for answer evaluation. Rubric items referencing papers that cannot be automatically matched to a Google Scholar article are removed. The top- $n$  reranked items form  $R_H$ , where the  $n$  is set to 8 to match the average size of the other rubrics;  $\sim 61\%$  of hybrid rubric items originate from  $R_S$ .

## 2.5 Dataset statistics and splits

RESEARCHQA is partitioned as train ( $\mathcal{D}_{\text{train}}$ ), validation ( $\mathcal{D}_{\text{validation}}$ ), and test ( $\mathcal{D}_{\text{test}}$ ) splits.  $\mathcal{D}_{\text{test}}$  (3.7K queries) ensures broad domain coverage by sampling 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 fields with left over queries (74 of 75 fields), 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.

## 3 Expert Validation

We conducted an IRB-approved user study to make Likert-scale quality assessments of RESEARCHQA queries (§3.1) and rubrics (§3.2). We also explored whether rubrics help approximate expert judgements when used in automatic eval-

uations, comparing judges using direct prediction and rubric-based prediction protocols as proxies to expert-labeled preference rankings (§3.3).

We recruited expert 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). Compensation was \$25 per hour of annotation.

### 3.1 Quality of Survey-Mined Queries

**Setup.** We ask experts to rate queries on clarity, naturalness, relevance, and variability in valid answers, as shown in Figure 3. Comparisons to parametric queries generated from scratch are difficult due to topic differences, so we compare queries generated from our pipeline (content-based) against an ablated pipeline that generates solely from survey titles (title-based). This analysis isolates the effect of pipeline components used to enhance query generation and filtering.

**Survey content yields unambiguous queries, eliciting answers without too much variability.** Experts rate  $\sim 83\%$  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 (93% versus 87%), 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. 96% are rated as *Slightly likely* or stronger to reflect researchers’ information needs, and 80-84% 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 Quality of Hybrid Rubrics

**Setup.** Hybrid rubrics are compared against survey rubrics and parametric rubrics. For each rubric item, experts judge the number of sentences that should be used to address the item in an answer and how its omission affects answer quality. 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 un-specific assessment (e.g., it is a mere rephrasing of the query);<sup>6</sup> (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. Annotation scales and labels are listed in Figure 3.

**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 (11%). Experts note

<sup>6</sup>A rule-based system is also employed to detect vacuous rubric items, where rubric items need to include at least one substantive word not present in the original query.

that parametric rubric items contain nonexistent paper titles, causing statements about their coverage in answers to be unanswerable or ambiguous.

### 3.3 Validating Evaluation Protocols Using Rubrics as Proxies to Expert Judgement

**Setup.** We collect expert judgements on answers responding to RESEARCHQA queries and assess whether rubrics help evaluator LLMs approximate expert judgments, comparing different evaluation protocols (interface visualized in Figure 11). We first generate answers using two retrieval-augmented systems: gpt-4.1-mini and gemini-2.5-flash, employing their providers’ embedding models for retrieval.<sup>7</sup> Each query is used to search the top-20 arXiv papers via Google search constrained 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.

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*). These ratings validate how closely evaluator LLMs align with human assessments of content coverage. (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*. Plurality voting determines the final label; if votes split evenly between *Left* and *Right* with no *Ties* or *Both-bad* labels, the result is marked as *Disagreement*. These annotations support evaluating whether rubrics enhance the agreement between evaluator LLMs and human judgments.

**Evaluator LLMs.** Open-source LLMs (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, 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*).

<sup>7</sup>text-embedding-3-large, text-embedding-004

We compare two types of automatic judges to predict pairwise preference. The **Direct Judge** ( $J_{\text{direct}}$ ) 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** ( $J_{\text{ensemble}}$ ), which uses both  $J_{\text{direct}}$  and Coverage to predict preferred answers: the answer with a larger sum of Coverage (0-4 scale) and a full 4 points for each  $J_{\text{direct}}$  comparison, is marked as the preferred answer. Formally, this sum can be expressed as:

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

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

**LLM-Human agreement on 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.

**Ensemble judges increase LLM-Human agreement on preference rankings.** We present the preference ranking accuracies of evaluator LLMs with respect to binary plurality preference labels in Figure 4. Each graph presents a base direct judge and the results of ensembling with the three types of rubrics we consider. In cases where judges predict ties, we assign partial credit.<sup>8</sup> Human-Human agreement indicates an upper bound of 84% for LLM-Human agreement. Ensemble judges ( $J_{\text{ensemble}}$ ) are consistently better estimators of expert preferences than

<sup>8</sup>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%.

Rubrics Improve LLM-Human Agreement on Preference Rankings

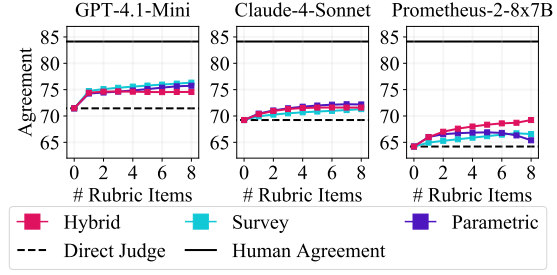


Figure 4: A comparison of how much rubrics can aid different evaluators in making predictions that agree with plurality human labels (y-axis) as a function of rubric size (x-axis). All direct judges benefit from integration of rubrics through the hybrid judge, substantially reducing their disagreement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

direct judges ( $J_{\text{direct}}$ ). In the best-performing model, gpt-4.1-mini, the LLM-Human and Human-Human agreement gap decreases from 12.7% to 9.6%, corresponding to a 24% relative reduction. Hybrid rubrics demonstrate highest performance gain across most systems, allowing prometheus-2-8x7b to match the direct judge performance of claude-4-sonnet. As visualized in Figure 14, hybrid rubrics enhance evaluation in 7 of 8 fields, where they are the best-performing rubric in 5 of 8.

## 4 LLM Systems Evaluation Setup

To benchmark top systems used for scholarly inquiry on RESEARCHQA, we evaluate 18 parametric, retrieval augmented, and agentic systems in both open-source and proprietary families (Table 2) 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  $A$ , constraining on date  $D$  and a response length  $L = 250$  words, limited to control for length bias (Singhal et al., 2024). A sample answer is visualized in Figure 7.

**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-


































| System   | Avg Length<br>(Words) | Leaderboard<br>Score $\uparrow$ | Coverage % $\uparrow$ |  |   |   |   |   |   |   |   |  |
|--|-----------------------|---------------------------------|-----------------------|--|---|---|---|---|---|---|---|--|
|  |                       |                                 | All domains           |  |  |  |  |  |  |  |  |  |
| Parametric   |                       |                                 |                       |  |   |   |   |   |   |   |   |  |
|  llama-3.3-70b                      | 167.4                 | 617 $\pm$ 13                    | 53.42 $\pm$ 0.26      | 51.82  | 54.21   | 54.89   | 55.74   | 53.91   | 53.22   | 58.10   |   |  |
|  claude-4-sonnet                    | 226.5                 | 1099 $\pm$ 09                   | 64.31 $\pm$ 0.31      | 62.92  | 64.96   | 66.71   | 67.08   | 63.33   | 59.85   | 66.17   |   |  |
|  gpt-4.1                            | 241.5                 | 1080 $\pm$ 09                   | 65.43 $\pm$ 0.26      | 63.98  | 66.84   | 66.66   | 67.38   | 65.45   | 63.48   | 68.46   |   |  |
|  qwen-3-32b                         | 219.3                 | 1038 $\pm$ 09                   | 66.64 $\pm$ 0.24      | 65.13  | 67.76   | 68.25   | 69.24   | 65.96   | 64.32   | 69.62   |   |  |
|  gemini-2.5-pro                     | 267.1                 | 1244 $\pm$ 10                   | 68.84 $\pm$ 0.25      | 67.42  | 70.20   | 69.82   | 71.86   | 68.28   | 65.83   | 72.06   |   |  |
| Retrieval  |                       |                                 |                       |  |   |   |   |   |   |   |   |  |
|  openscholar-8b <sup>i</sup>        | 499.9                 | 478 $\pm$ 17                    | 54.71 $\pm$ 0.30      | 54.08  | 56.15   | 54.76   | 54.98   | 54.67   | 52.69   | 57.46   |   |  |
|  gemini-2.5-pro <sup>ii</sup>       | 270.4                 | 945 $\pm$ 10                    | 59.92 $\pm$ 0.31      | 58.73  | 61.46   | 61.13   | 61.72   | 57.79   | 56.71   | 63.96   |   |  |
|  qwen-3-32b <sup>iii</sup>          | 265.5                 | 1011 $\pm$ 09                   | 60.90 $\pm$ 0.30      | 57.62  | 62.93   | 64.25   | 65.58   | 60.49   | 60.23   | 65.82   |   |  |
|  claude-4-sonnet <sup>iv</sup>      | 238.4                 | 972 $\pm$ 09                    | 62.50 $\pm$ 0.31      | 61.94  | 63.48   | 62.97   | 64.01   | 61.67   | 58.05   | 65.74   |   |  |
|  gpt-4.1 <sup>v</sup>               | 263.6                 | 1020 $\pm$ 09                   | 64.80 $\pm$ 0.27      | 63.69  | 66.65   | 65.11   | 66.72   | 64.25   | 62.09   | 66.33   |   |  |
| Agentic  |                       |                                 |                       |  |   |   |   |   |   |   |   |  |
|  sonar                              | 242.2                 | 862 $\pm$ 10                    | 58.61 $\pm$ 0.30      | 56.61  | 60.55   | 59.43   | 61.62   | 59.97   | 57.48   | 62.80   |   |  |
|  openscholar-8b+feedback            | 788.8                 | 769 $\pm$ 12                    | 58.72 $\pm$ 0.30      | 57.77  | 59.96   | 58.62   | 61.48   | 57.27   | 57.29   | 62.48   |   |  |
|  sonar-reasoning                    | 280.5                 | 1115 $\pm$ 10                   | 64.33 $\pm$ 0.29      | 62.73  | 66.00   | 65.19   | 68.11   | 62.68   | 61.76   | 67.49   |   |  |
|  gpt-4o-search-preview              | 255.0                 | 992 $\pm$ 09                    | 65.98 $\pm$ 0.28      | 65.52  | 68.21   | 65.01   | 66.60   | 66.07   | 62.62   | 65.63   |   |  |
|  gemini-2.5-pro+grounding           | 278.5                 | 960 $\pm$ 09                    | 68.51 $\pm$ 0.25      | 67.38  | 70.02   | 68.76   | 70.99   | 68.09   | 65.98   | 71.21   |   |  |
|  claude-4-sonnet+ws                 | 327.8                 | 1149 $\pm$ 10                   | 69.18 $\pm$ 0.25      | 69.54  | 70.49   | 67.59   | 70.28   | 68.14   | 64.70   | 67.13   |   |  |
|  o4-mini-deep-research <sup>†</sup> | 271.6                 | 1145 $\pm$ 10                   | 72.69 $\pm$ 0.53      | 74.02  | 73.58   | 70.57   | 74.04   | 73.25   | 68.99   | 74.54   |   |  |
|  sonar-deep-research                | 267.3                 | 1505 $\pm$ 17                   | 75.29 $\pm$ 0.25      | 75.01  | 76.31   | 74.48   | 76.77   | 75.34   | 72.47   | 78.01   |   |  |

Table 2: 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: <sup>i</sup>openscholar-retrieve and openscholar-reranker; <sup>ii</sup>text-embedding-004; <sup>iii</sup>gte-qwen-2-7b-instruct; <sup>iv</sup>voyage-3-large; <sup>v</sup>text-embedding-3-large. <sup>†</sup>This system costs  $\sim$ \\$1.15 per query, so statistics are only computed on answers sampled for tournament battles ( $\sim$ 20%).

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

**Parametric systems.** We directly generate answers using default configs of API providers (OpenRouter<sup>9</sup> for open-source models).

**Retrieval-augmented systems.** We implement retrieval systems directly and pair embedding models with language models from the same providers. 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 pa-

| Systems               | Avg Length |          |            | Coverage % |          |          |
|-----------------------|------------|----------|------------|------------|----------|----------|
|                       | $L=250$    | $\neg L$ | $\Delta\%$ | $L=250$    | $\neg L$ | $\Delta$ |
| <b>Parametric</b>     |            |          |            |            |          |          |
| 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     |
| <b>Retrieval</b>      |            |          |            |            |          |          |
| 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     |
| <b>Agentic</b>        |            |          |            |            |          |          |
| claude-4-sonnet+ws    | 329.5      | 483.3    | +46%       | 70.2       | 71.4     | +1.2     |
| 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 3: 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).

per 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

<sup>9</sup><https://openrouter.ai/>



| 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 4: 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.

as in the parametric setting and employ greedy decoding for OpenScholar.

**Agentic systems.** We label systems involving advanced engineering, e.g., tool use, feedback, or refinement, in the agentic category. We generate answers directly in agentic systems using default implementations in their respective repositories or APIs without modification.

**Battle construction.** We construct a total of 7.7K 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 ( $\sim 800$  battles per model) and all model matchups are uniformly represented in each domain, ensuring connectivity of the comparison network.

## 5 LLM Systems Evaluation Results

### 5.1 Response Results

**Result by systems.** All systems struggle to fully cover rubric items, with no parametric or retrieval augmented systems we evaluated exceeding 70% coverage (Table 2). 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 a 82% win rate over the next highest rated system, *gemini-2.5-pro*.

**Result by rubric type.** We analyze coverage of rubric items belonging to different semantic categories.<sup>10</sup> In Table 4, we show examples and distribution of *sonar-deep-research* errors, but all systems follow similar distributions. We perform an analysis on 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 other systems are shown in Figure 12 and Figure 13.

**Results by domains.** Performance varies between research domains (Table 2). Queries from Health Sciences & Medicine and Humanities consistently rank as the bottom 2 domains by Coverage %: in 10/18 systems for Health Sciences and 18/18 for Humanities. These trends identify the limitations of focusing evaluation on Engineering and Physical Sciences, which consistently rank as top-performing domains and are also heavily represented in prior evaluations (Table 1).

### 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 %.

<sup>10</sup>Each rubric item is lemmatized, using NLTK, and matched to words expressing 6 possible rubric types.

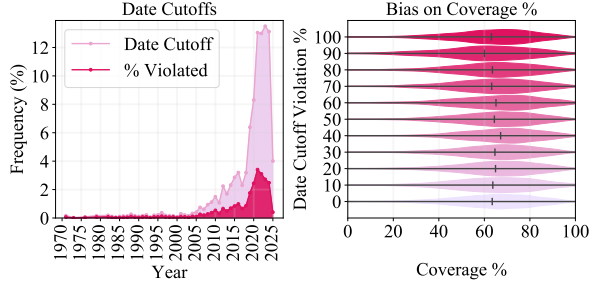


Figure 5: Rubrics are recent, mostly originating from surveys in the past decade. When cited sources violate date cutoff years, there is little bias on Coverage %.

For each answer, we count the % of cited sources that violate the instructed date cutoff (survey article publication year). Sources that violate the date cutoff make up  $\sim 30\%$  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  $\pm 1.3\%$  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 agentic systems do not have the option to restrict search retrieval for answer generation. Leakage is detected by checking whether the title of a survey paper appears in the reference section of the answer.<sup>11</sup> We compare Coverage % in subsets of answers where leakage is and is not detected for top-performing agentic systems in Table 5 and all systems in Table 7.

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  $\sim 1.1\%$  when distilled surveys are retrieved to generate answers. These small differences assuage our concerns of leakage advantages

<sup>11</sup>To address noise in paper titles, we lowercase text, remove special characters, and search for 6-gram matches.

| Agentic System           | L% | Coverage %    |              |          |
|--------------------------|----|---------------|--------------|----------|
|                          |    | $\neg$ Leaked | Leaked       | $\Delta$ |
| claude-4-sonnet+ws       | 30 | <b>71.27</b>  | 68.30        | -3.0     |
| gemini-2.5-pro+grounding | 2  | 66.91         | <b>68.54</b> | +1.6     |
| o4-mini-deep-research    | 28 | <b>74.82</b>  | 71.85        | -3.0     |
| sonar-deep-research      | 21 | 74.46         | <b>75.51</b> | +1.1     |

Table 5: Agentic 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 ( $\neg$  Leaked).

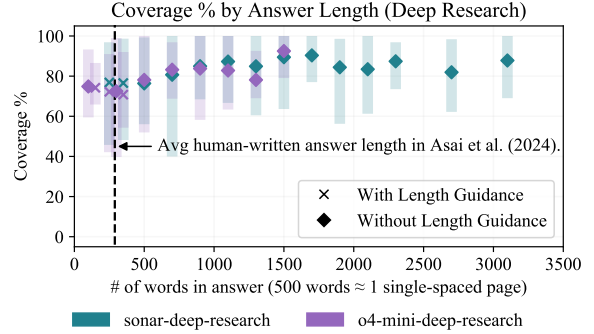


Figure 6: Coverage % increases with answer length, up until  $\sim 2K$  words (about 4 pages of text).

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.

**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, retrieval, and agentic systems by Coverage % and additionally generate answers for 3 queries per field (225 total queries). Average statistics are shown in Table 3, and trends for agentic systems are visualized in Figure 6 and for the remaining systems in Figure 16.

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 16). 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  $\sim 2K$  words (about 4 pages of text), as shown in Table 3, 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 article, 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. 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 Altschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. [Gpt-4 technical report](#). *arXiv preprint*.
- 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?](#) *arXiv preprint*.
- Anthropic. 2024. [Introducing claude 4](#).
- Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee 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*.
- Sören Auer, Dante A. C. Barone, Cassiano Bartz, Eduardo G. Cortes, Mohamad Yaser Jaradeh, Oliver Karras, Manolis Koubarakis, Dmitry Mourmoutsev, 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*.
- Ralph Allan 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 Nikolas 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, et al. 2014. [Hierarchical summarization: Scaling up multi-document summarization](#). In *Association for Computational Linguistics (ACL)*.
- 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](#). *arXiv preprint*.
- A.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*.
- 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](#). *arXiv preprint*.
- 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 Yuchen 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)*.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. [Retrieval-augmented generation for knowledge-intensive nlp tasks](#). *Neural Information Processing Systems (NeurIPS)*.
- Bill Yuchen 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 Lu 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](#). *arXiv preprint*.
- 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*.
- Derek J De Solla Price. 1963. [Little science, big science](#). Columbia University Press.
- Yiwei Qin, Kaiqiang Song, Yebowen Hu, Wenlin Yao, Sangwoo Cho, Xiaoyang Wang, Xuan-sheng 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 pre-training term frequencies on few-shot numerical reasoning](#). In *Empirical Methods in Natural Language Processing (EMNLP)*.
- David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe 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 Krnias, 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](#). *arXiv preprint*.
- 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 Chee Chang, Chloe Anastasiades, Dany Haddad, Aakanksha Naik, Amber Tanaka, Angele Zamarron, Cecile Nguyen, Jena D Hwang, Jason Dunkleberger, et al. 2025. [Ai2 scholar qa: Organized literature synthesis with attribution](#). *arXiv preprint*.
- 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 Ponnampati, Samuel G Rodriques, and Andrew D White. 2024. [Language agents achieve superhuman synthesis of scientific knowledge](#). *arXiv preprint*.
- 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)*.
- Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. 2024. [Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context](#). *arXiv preprint*.
- 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*.
- Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Qingsong Wen, Wei Ye, et al. 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, Ziyang Jiang, et al. 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, et al. 2025. [Qwen3 technical report](#). *arXiv preprint*.
- 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, Taira Anderson, Jonathan Bragg, Joseph Chee Chang, Jesse Dodge, Matt Latzke, et al. 2025. [Sciarena: An open evaluation platform for foundation models in scientific literature tasks](#). *arXiv preprint*.
- 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 dataset curation in detail.

### 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):

1. Business, Economics, & Management: 16 subfields, 285 unique venues
2. Chemical & Material Sciences: 17 subfields, 318 unique venues
3. Physics & Mathematics: 21 subfields, 370 unique venues
4. Humanities, Literature, & Arts: 26 subfields, 481 unique venues
5. Life Sciences: 30 subfields, 509 unique venues
6. Social Sciences: 31 subfields, 526 unique venues
7. Engineering & Computer Science: 50 subfields, 925 unique venues
8. Health & Medical Sciences: 66 subfields, 1220 unique venues

**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. Next, for each article, we use gpt-4.1-mini to discriminate between actual literature review articles (i.e., articles whose primary purpose is to analyze, synthesize, and evaluate research on a topic) and articles that mention a keyword like *survey* in its title. 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:** Sections that address specific issues typically are not the abstract, introduction, or other summary sections. Therefore, we remove 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*, *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 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 gpt-4.1-mini to parse the section content into a discourse tree to generate an initial query and initial reference answer ( $Q_{\text{initial}}$ ,  $\hat{A}_{\text{initial}}$ ) from the parsed representation and the date cutoff  $D$ .

To select high quality queries, all queries (total = 319K, yield = 21K) are subject to the filters:

- **Initial query is self-contained:** We apply gpt-4.1-mini to assign each initial query  $Q_{\text{initial}}$  a self-containment score from 1 to 10, where higher scores indicate more self-containment. To pass this filter, the self-containment score must be  $\geq 7$ .
- **Initial query has low answer variability:** We apply gpt-4.1-mini to assign each initial query  $Q_{\text{initial}}$  an answer variability score from 1 to 10, where higher scores indicate likely answer variability (e.g., expert disagreement or subjectivity). To pass this filter, the answer variability score must be  $\leq 4$ .



- **(Rephrasing Step) Query and reference answer cohesion:** Initial queries and initial reference answers ( $Q_{\text{initial}}, \hat{A}_{\text{initial}}$ ) are rephrased to queries  $Q$  and reference answers  $\hat{A}$  to improve their cohesion, and we do not eliminate any queries at this step.
- **Query does not contain a citation:** We apply `gpt-4.1-mini` to detect whether a citation is included 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 self-contained, 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*).
- **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 alignment with the field, we apply `gpt-4.1-mini` to remove 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.

## A.2 Rubric generation from survey articles

We create rubrics with `gpt-4.1-mini`. 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?*

**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 (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.

|   |
|---|
| <p># Query: What strategies are recent research efforts employing to reduce the wall-clock training time for BERT models?</p> <p># Date: 2020-10-02</p> <p># Field: Natural Language Processing</p> <p># Answer:</p> <p>Recent research efforts have focused on various strategies to reduce the wall-clock training time for BERT models, aiming to make them more efficient without compromising performance. Key approaches include:</p> <p><b>**1. Multi-Stage Layerwise Training (MSLT):**</b> This method decomposes the training process into stages, starting with a smaller model and progressively adding layers. At each stage, only the newly added layers are trained, while previously trained layers remain fixed. This approach reduces computational overhead and communication time, achieving over 110% training speedup without significant performance degradation. [1]</p> <p><b>**2. Early-Bird Lottery Tickets (EarlyBERT):**</b> Inspired by the lottery ticket hypothesis, EarlyBERT identifies structured winning tickets early in training by slimming self-attention and fully-connected sub-layers within the transformer. This technique reduces training time by 35-45% while maintaining performance comparable to standard BERT models. [2]</p> <p><b>**3. Knowledge Distillation (DistilBERT):**</b> DistilBERT leverages knowledge distillation during pre-training to create a smaller, faster model that retains 97% of BERT's language understanding capabilities. This results in a model that is 40% smaller and 60% faster, making it more efficient for both training and inference. [3]</p> <p><b>**4. Coarse-Refined Training Framework (CoRe):**</b> CoRe introduces a two-phase training process. Initially, a relaxed BERT model with fewer parameters is trained. In the second phase, this model is transformed into the original BERT and retrained. This framework significantly reduces training time without sacrificing performance. [4]</p> <p>[...]</p> <p>These strategies collectively contribute to more efficient BERT training, making it feasible to deploy these models in resource-constrained environments.</p> <p>[1] Progressively Stacking 2.0: A Multi-stage Layerwise Training Method for BERT Training Speedup (2011)</p> <p>[2] EarlyBERT: Efficient BERT Training via Early-bird Lottery Tickets (2020)</p> <p>[3] DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter (1910)</p> <p>[4] CoRe: An Efficient Coarse-refined Training Framework for BERT (2011)</p> <p>[5] NVIDIA Closes World's Fastest BERT Training Time and Largest Transformer Based Model, Paving Path For Advanced Conversational AI   NVIDIA Technical Blog (2025)</p> <p>[6] Amazon Web Services achieves fastest training times for BERT and Mask R-CNN   AWS Machine Learning Blog (2019)</p> <p>[7] Smart Batching Tutorial - Speed Up BERT Training   Chris McCormick (2020)</p> <p># Rubric Item with Coverage Judgement</p> <ol style="list-style-type: none"> <li>1. [Not at all] 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?</li> <li>2. [Barely] Does the response provide a detailed explanation of the LAMB optimizer and its impact on reducing BERT training time?</li> <li>3. [Completely] Does the response mention the use of larger batch sizes as a method to reduce training time?</li> <li>4. [Completely] Does the response include strategies that focus on increasing computational efficiency for reducing BERT training time?</li> <li>5. [Completely] Does the response include examples of specific optimization techniques used to reduce training time for BERT models?</li> </ol> <p>[...]</p> |
|---|

Figure 7: A sample from gpt-4o-search-preview in natural language processing field.

|   |
|---|
| <p># Query: How do beta-blockers and diuretics affect blood pressure and obstructive sleep apnea severity in hypertensive patients with OSA?</p> <p># Date: 2024-8-29</p> <p># Field: Vascular Medicine</p> <p># Answer:</p> <p>The interplay between beta-blockers, diuretics, blood pressure (BP), and obstructive sleep apnea (OSA) severity in hypertensive patients reveals complex pharmacological dynamics. <b>Beta-blockers</b> demonstrate dualistic effects: while they effectively reduce nighttime BP in OSA patients by attenuating sympathetic overactivity [1], [5], emerging evidence associates their use with elevated cardiovascular risks. A 2023 cohort study of 37,000 OSA patients found beta-blocker users faced higher 5-year mortality (4.9% vs 4.0%) and cardiovascular disease rates (13.0% vs 9.4%) compared to non-users [2]. This paradox may stem from beta-blockers' potential to exacerbate hypoxemia-induced bradyarrhythmias during apneic events [3], though some studies contradict this by showing no increased nocturnal arrhythmia risk [4]. Additionally, lipophilic beta-blockers like metoprolol may worsen sleep architecture through central nervous system penetration, with case reports linking them to non-REM parasomnias [6].</p> <p>Conversely, <b>diuretics</b> exhibit more favorable OSA-modifying properties, particularly in patients with fluid retention. Thiazides and aldosterone antagonists like spironolactone reduce OSA severity by mitigating nocturnal rostral fluid shifts that contribute to pharyngeal crowding. A propensity-matched analysis of 69,564 OSA patients revealed diuretics decreased apnea-hypopnea index (AHI) by 22% in overweight/moderately obese individuals compared to non-users [7]. Mineralocorticoid receptor antagonists demonstrate particularly robust effects-randomized trials show spironolactone reduces AHI by 45% (39.8 to 22.0 events/hour) and improves BP control in resistant hypertension with OSA [8], [9]. This dual benefit arises from both diuresis and aldosterone inhibition, which diminishes fluid retention and potentially stabilizes upper airway musculature [10], [12].</p> <p>The therapeutic dichotomy suggests beta-blockers may be suboptimal first-line agents for hypertensive OSA patients despite their antihypertensive efficacy, whereas diuretics offer synergistic OSA and BP management. Current guidelines should be reappraised considering these differential effects on cardiopulmonary outcomes.</p> <p>[1] safety and efficacy of beta-blocker therapy in obstructive sleep ... (2010)<br/> [2] New study finds that beta-blockers increase cardiovascular risk for sleep apnoea patients   UCL School of Pharmacy – UCL – University College London (2023)<br/> [3] Nocturnal Arrhythmias and Heart-Rate Swings in Patients With ...<br/> [4] Are beta blockers beneficial or harmful in obstructive sleep apnoea?   CPRD (2019)<br/> [...]</p> <p># Rubric Item with Coverage Judgement</p> <ol style="list-style-type: none"> <li>1. [Barely] Does the response mention effects of beta-blockers on both systolic and diastolic blood pressure in these patients?</li> <li>2. [Not at all] Does the response address how combination therapy involving beta-blockers and diuretics affects blood pressure in OSA patients?</li> <li>3. [Mostly] Does the response explore in depth the distinct mechanisms by which beta-blockers and diuretics influence both blood pressure and OSA severity?</li> <li>4. [Mostly] Does the response provide a detailed explanation of how fluid retention and its redistribution contribute to OSA pathophysiology and how diuretics mitigate this?</li> <li>5. [Completely] Does the response describe the effects of diuretics on blood pressure control in this patient population?</li> <li>6. [Mostly] Does the response address whether beta-blockers impact the severity or frequency of obstructive sleep apnea symptoms?</li> <li>7. [Completely] Does the response discuss the influence of diuretics on obstructive sleep apnea severity or related respiratory parameters?</li> <li>8. [Not at all] Does the response cite KRAICZI et al. (2000) (title: Comparison of atenolol, amlodipine, enalapril, hydrochlorothiazide, and losartan for antihypertensive treatment in patients with obstructive sleep apnea) to support the statement that atenolol lowers blood pressure more effectively than other antihypertensive drugs in OSA-related hypertension?</li> </ol> |
|---|

Figure 8: A sample from sonar-deep-research in vascular medicine field.

|  |
|--|
| <p># Query: How does heat treatment affect the wear resistance and the underlying mechanisms in FeCoNiCrMn and AlSiTiCrFeCoNi coatings?</p> <p># Date: 2022-5-21</p> <p># Field: Materials Engineering</p> <p># Answer:</p> <p>Heat treatment significantly influences the wear resistance of both FeCoNiCrMn and AlSiTiCrFeCoNi coatings through distinct microstructural modifications and phase transformations. For FeCoNiCrMn coatings, heat treatment shows minimal impact on wear performance, with polished as-sprayed and heat-treated coatings exhibiting nearly equivalent wear rates [2]. The average microhardness increases from 423 <math>\pm</math> 35 HV0.3 to 492 <math>\pm</math> 20 HV0.3 after heat treatment due to enhanced cohesive strength among splats and increased oxide content [2]. However, this hardness improvement doesn't translate to significantly better wear resistance, as both conditions maintain similar friction coefficients and wear mechanisms [2].</p> <p>The underlying mechanisms involve precipitation hardening through intermetallic compound formation, which increases coating hardness and reduces abrasive wear effects [1]. Heat treatment enables controlled precipitation of strengthening phases like Cr3Si, fundamentally altering the wear behavior from predominantly adhesive mechanisms to more resistant tribological responses [1].</p> <p>The underlying mechanisms involve precipitation hardening through intermetallic compound formation, which increases coating hardness and reduces abrasive wear effects [1]. Heat treatment enables controlled precipitation of strengthening phases like Cr3Si, fundamentally altering the wear behavior from predominantly adhesive mechanisms to more resistant tribological responses [1].</p> <p>[1] Thermal spray high-entropy alloy coatings : a review (2020)<br/> [2] Microstructural and tribological behavior of thermal spray CrMnFeCoNi high entropy alloy coatings (2022)<br/> [3] Comparison of micro-nano FeCoNiCrAl and FeCoNiCrMn coatings prepared from mechanical alloyed high-entropy alloy powders (2021)</p> <p># Rubric Item with Coverage Judgement</p> <ol style="list-style-type: none"> <li>1. [Mostly] Does the response state that heat treatment improves wear resistance by strengthening the microstructure in both FeCoNiCrMn and AlSiTiCrFeCoNi coatings?</li> <li>2. [Mostly] Does the response explore in detail how heat treatment influences the microstructural changes that lead to improved wear resistance in both FeCoNiCrMn and AlSiTiCrFeCoNi coatings?</li> <li>3. [Completely] Does the response provide an in-depth explanation of the specific mechanisms, such as phase precipitation or oxide formation, responsible for the wear resistance enhancement after heat treatment?</li> <li>4. [Mostly] Does the response describe the microstructural changes induced by heat treatment in these coatings?</li> <li>5. [Completely] Does the response explain the underlying mechanisms responsible for changes in wear resistance due to heat treatment?</li> <li>6. [Completely] Does the response differentiate the effects of heat treatment on FeCoNiCrMn coatings versus AlSiTiCrFeCoNi coatings?</li> <li>7. [Completely] Does the response include a comparison of wear resistance before and after heat treatment for both FeCoNiCrMn and AlSiTiCrFeCoNi coatings?</li> <li>8. [Mostly] Does the response provide a detailed explanation of the wear mechanisms before and after heat treatment for both FeCoNiCrMn and AlSiTiCrFeCoNi coatings?</li> </ol> |
|--|

Figure 9: A sample from retrieval claude-4-sonnet in materials engineering field.



### Paper Section Contents

**sent1:** In the context of AI fairness, the term “bias” commonly refers to skews that result in undesirable impacts (Crawford, 2017) and is quantifiable with some metric.

[...]

**sent4:** Because of the difficulty in defining metrics, existing works define bias loosely as demographic inequality and use intermediate proxy metrics to comparatively measure bias.

**sent5:** Examples include: Regard Ratio: negative-neutral-positive regard score ratios of text generated from bias-inducing prompts (Sheng et al., 2019).

**sent6:** Sentiment Ratio: negative-neutral-positive sentiment score ratios of text generated from African American English (AAE) versus White-Aligned English (WAE) prompts (Groenwold et al., 2020).

[...]

### Question-Tree Summary

Q1. What is the common definition of "bias" in the context of AI fairness? **sent1**

Q2. How do existing works define and measure bias in language generation tasks? **sent4**

Q2.1. What are some examples of intermediate proxy metrics used to measure bias in language generation tasks? **sent5, sent6, sent7, sent8**

Q2.2. What metrics are favored in transformation generation tasks? **sent10**

[...]

Figure 10: **A sample of hierarchical summary of the section contents.** The contents of a paper section are represented as a list of indexed sentences (named *Paper Section Contents*), then summarized into a tree-based structure of questions followed by their supporting sentence indexes (named *Question-Tree Summary*) by prompting LLM. Both will be fed altogether into the next-stage query generator in the RESEARCHQA synthesis pipeline. This implementation is inspired by hierarchical summarization (Christensen et al., 2014) and question-under-discussion parsing (Benz and Jasinskaja, 2017; Wu et al., 2023).

Step 3 of 8

Query Progress: 0 out of 20

Data Annotation (Phase 1)

Query: How are confidence-based early exit strategies implemented in BERT inference models?  
Knowledge Cutoff: 2022-02-15

| Model 1   | Model 2  |
|---|--|
| <p>Confidence-based early exit strategies in BERT inference models involve adding classifiers at intermediate layers of the BERT model [1]. During inference, the model calculates layer-wise BERT representations sequentially [1]. As the model reaches a classification layer, it makes a prediction and interprets the label scores output by softmax as confidence scores [1]. Based on these confidence scores, the model decides whether to exit early or continue to the next layer [1]. This approach aims to provide a more efficient prediction by exiting the model before reaching the final layer [1].</p> <p>ELBERT implements a confidence-window based early exit mechanism using ALBERT as the backbone model [2]. This mechanism allows for an input-adaptive efficient inference [2]. The losses of inputs exiting at different depths of ELBERT are calculated during training [2]. For classification, the early exit loss at each layer is calculated using cross-entropy [2]. The early exit mechanism helps to establish appropriate attention to key words, enabling the model to exit from simple inputs in advance and avoid overthinking [2].</p> <p>This early exit strategy leads to excellent inference acceleration and outperforms other methods used for accelerating BERT [2]. Models can achieve a faster speed while preserving accuracy [1]. The confidence-window based early exit mechanism enables an input-adaptive efficient inference, saving inference time and computation cost [2].</p> <p>[1] The Right Tool for the Job: Matching Model and Instance Complexities (2020) <a href="#">link</a><br/>[2] ELBERT: FAST ALBERT WITH CONFIDENCE-WINDOW BASED EARLY EXIT (2021) <a href="#">link</a></p> | <p>Confidence-based early exit strategies in BERT inference models are implemented by adding output layers to intermediate layers of the BERT model, allowing predictions to be made at various depths of the network. This approach leverages the multilayered structure of BERT, where each output layer can make a prediction based on the representations up to that point, potentially exiting the model earlier for simpler instances [1]. The decision to exit early is based on confidence scores derived from the softmax outputs of these intermediate classifiers. These scores are interpreted as confidence levels, and if a classifier's confidence exceeds a certain threshold, the model exits early, thus saving computational resources [1].</p> <p>The confidence scores are calibrated to ensure reliable early exits, and the method is designed to balance the trade-off between speed and accuracy. This is achieved by allowing early exits for simpler instances, which require less processing, while more complex instances continue through additional layers for more accurate predictions [1]. The approach is efficient, requiring almost no additional training resources compared to the baseline BERT model, and it alleviates the need for retraining multiple models at different efficiency levels [1]. However, a critique of confidence-based criteria is the need to manually define an arbitrary threshold, which can lead to overconfidence in certain deep models [2]. Despite this, the method has been shown to significantly improve inference speed while maintaining accuracy [1].</p> <p>[1] The Right Tool for the Job: Matching Model and Instance Complexities (2020) <a href="#">link</a><br/>[2] Adaptive Inference through Early-Exit Networks: Design, Challenges and Directions (2021) <a href="#">link</a></p> |

1. Instructions

- Importance Rating
- Model 1 Rating
- Model 2 Rating
- Preference Rating (current)**

Please read both model answers. Then make a preference rating, leave a comment, and then press next.

2. Rubric Summary and Preference Rating

| Question   | Emphasis | Necessity | Confidence | Model 1 | Model 2 |
|--|----------|-----------|------------|---------|---------|
| Q[17] Does the response mention the Branchynet Fast approach (title: Branchynet: Fast inference via early exiting from deep neural networks) as a foundational method for implementing early exit strategies in BERT inference models? <a href="#">asach</a> |          |           |            | 5/5     | 5/5     |

Which response is better?

Note: The better response may not be the one that scored higher on the rubric.

Left is Better

Right is Better

Tie

Both are Bad

Label Definitions: (Show)

3. Comments

(Required) In a few words, please explain your choice.

Hide Annotation Panel

Next Phase

Figure 11: RESEARCHQA human evaluation interface.

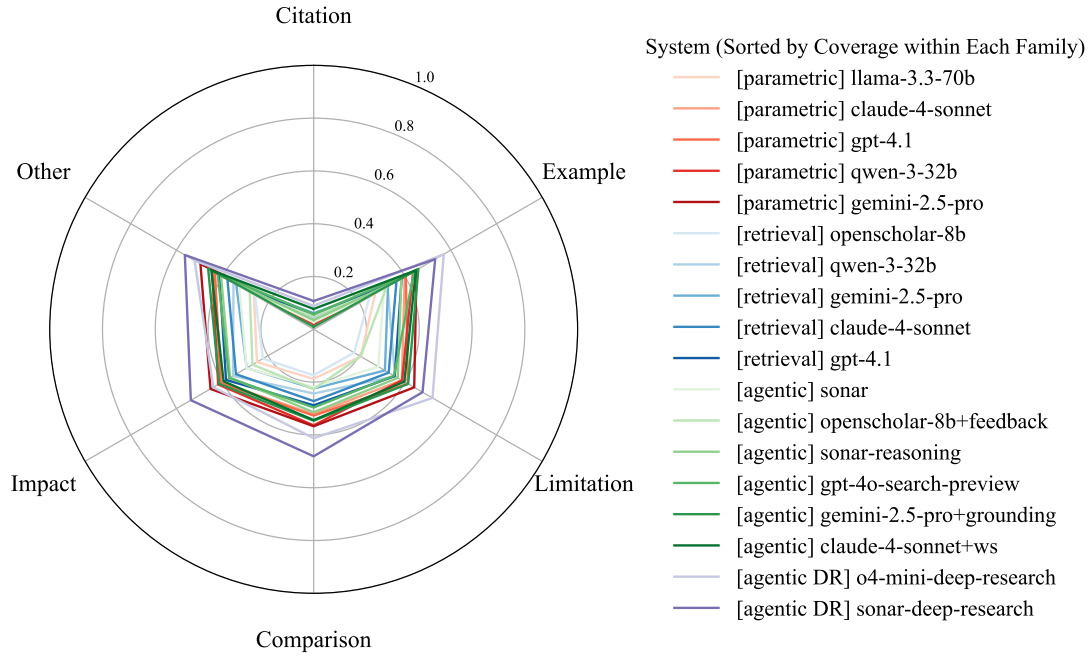


Figure 12: **Comparison of LLM System Performance by Rubric Type (Radar).** Performance is measured as the percentage of fully covered rubrics. Each rubric type is represented as an objective, with sonar-deep-research forming the Pareto frontier. “DR” denotes “Deep Research.”

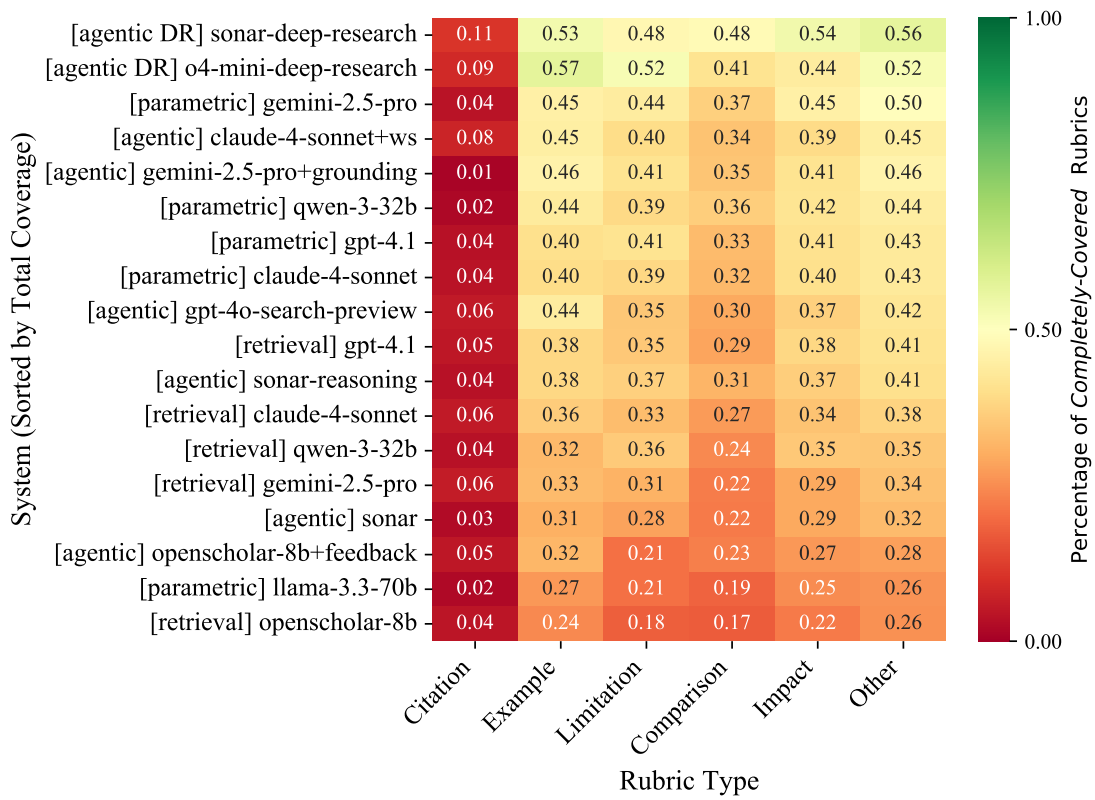


Figure 13: **Comparison of LLM System Performance by Rubric Type (Heatmap).** Performance is measured as the percentage of fully covered rubrics. Underperforming rubric types (x-axis) are shown for each system in the pairwise tournament (y-axis). “DR” denotes “Deep Research.”

| System                     | Model Card                   | Other Settings / Models  | Cost   |
|----------------------------|------------------------------|--|--------|
| <b>Parametric</b>          |                              |  |        |
| ∞ llama-3.3-70b            | llama-3.3-70b-instruct       |  | \$5    |
| A claude-4-sonnet          | claude-sonnet-4-20250514     |  | \$24   |
| 🌀 gpt-4.1                  | gpt-4.1                      |  | \$15   |
| 🌀 qwen-3-32b               | qwen3-32b                    |  | \$5    |
| G gemini-2.5-pro           | gemini-2.5-pro-preview-06-05 |  | \$15   |
| <b>Retrieval</b>           |                              |  |        |
| ✦ openscholar-8b           | llama-3.1_openscholar-8b     | [Embed] openscholar-retriever<br>[Reranker] openscholar-reranker | \$0    |
| 🌀 qwen-3-32b               | qwen3-32b                    | [Embed] gte-qwen-2-7b-instruct                                   | \$10   |
| G gemini-2.5-pro           | gemini-2.5-pro-preview-06-05 | [Embed] text-embedding-004                                       | \$35   |
| A claude-4-sonnet          | claude-sonnet-4-20250514     | [Embed] voyage-3-large   | \$218  |
| 🌀 gpt-4.1                  | gpt-4.1                      | [Embed] text-embedding-3-large                                   | \$151  |
| <b>Agentic</b>             |                              |  |        |
| 🌀 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  |
| G gemini-2.5-pro+grounding | gemini-2.5-pro-preview-06-05 | dynamic_threshold=0.3  | \$30   |
| A claude-4-sonnet+ws       | claude-sonnet-4-20250514     | tools.type=web_search_20250305<br>max_uses=3                     | \$450  |
| 🌀 o4-mini-deep-research    | o4-mini-deep-research        | tools.type=web_search_preview                                    | \$4200 |
| 🌀 sonar-deep-research      | sonar-deep-research          | reasoning.summary=auto   | \$1500 |

Table 6: **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), A Anthropic (Anthropic, 2024), 🌀 OpenAI (Achiam et al., 2023; OpenAI, 2025), 🌀 Alibaba (Yang et al., 2025), G Google (Team et al., 2024; DeepMind, 2025), ✦ AI2 (Asai et al., 2024), 🌀 Perplexity (Perplexity, 2025).

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

Table 7: Retrieving the distilled survey does not provide large advantages toward higher Coverage %. Parametric systems cite the related survey as a source at 3-8% leakage (L%), retrieval-augmented systems do not produce answers with leakage (likely because related surveys are intentionally removed from their retrieval), and agentic systems often result in 20-30% leakage. Coverage % roughly stays the same with leakage (Leaked) and without leakage (¬ Leaked).

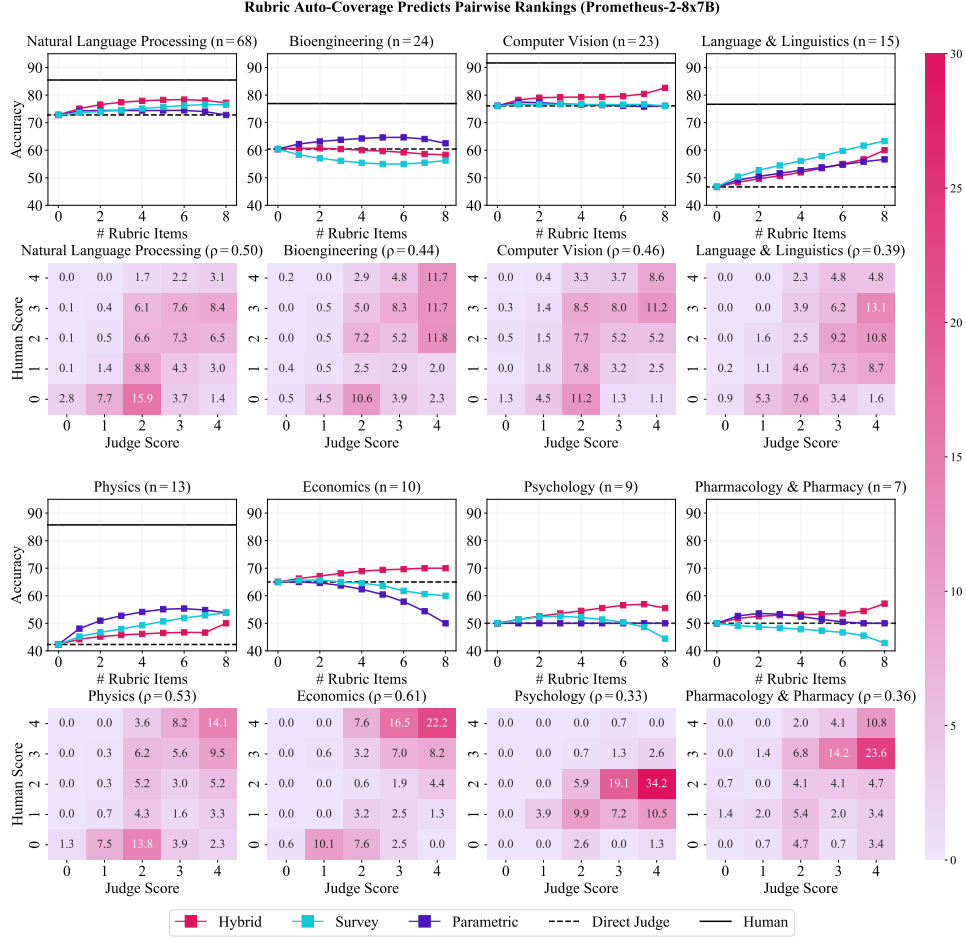


Figure 14: **Rubrics used in ensemble judges can help to predict expert-labeled pairwise preference.** Each graph represents fields of expertise in our pool of expert annotators. Each heatmap represents confusion matrices of LLM and human rubric coverage scores. Graphs are ordered such in descending order of data availability: while NLP had multiple annotators to account for disagreement for each annotation task (yielding 68 binary plurality labels), other fields have less. In 7 out of 8 fields, hybrid rubrics can improve accuracy to plurality labels, with hybrid rubrics obtaining highest performance in 5 out of 8.

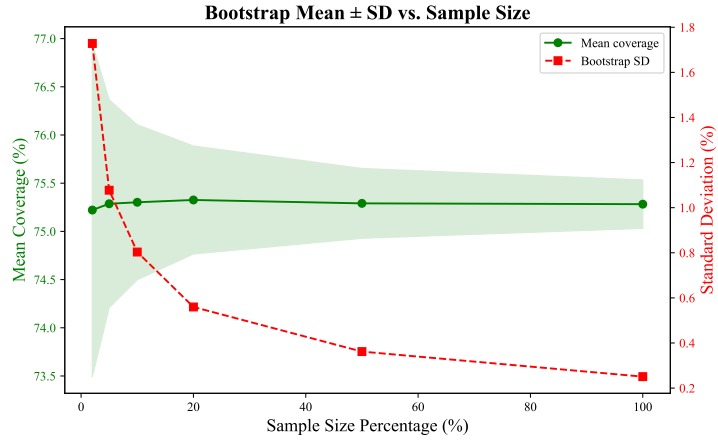
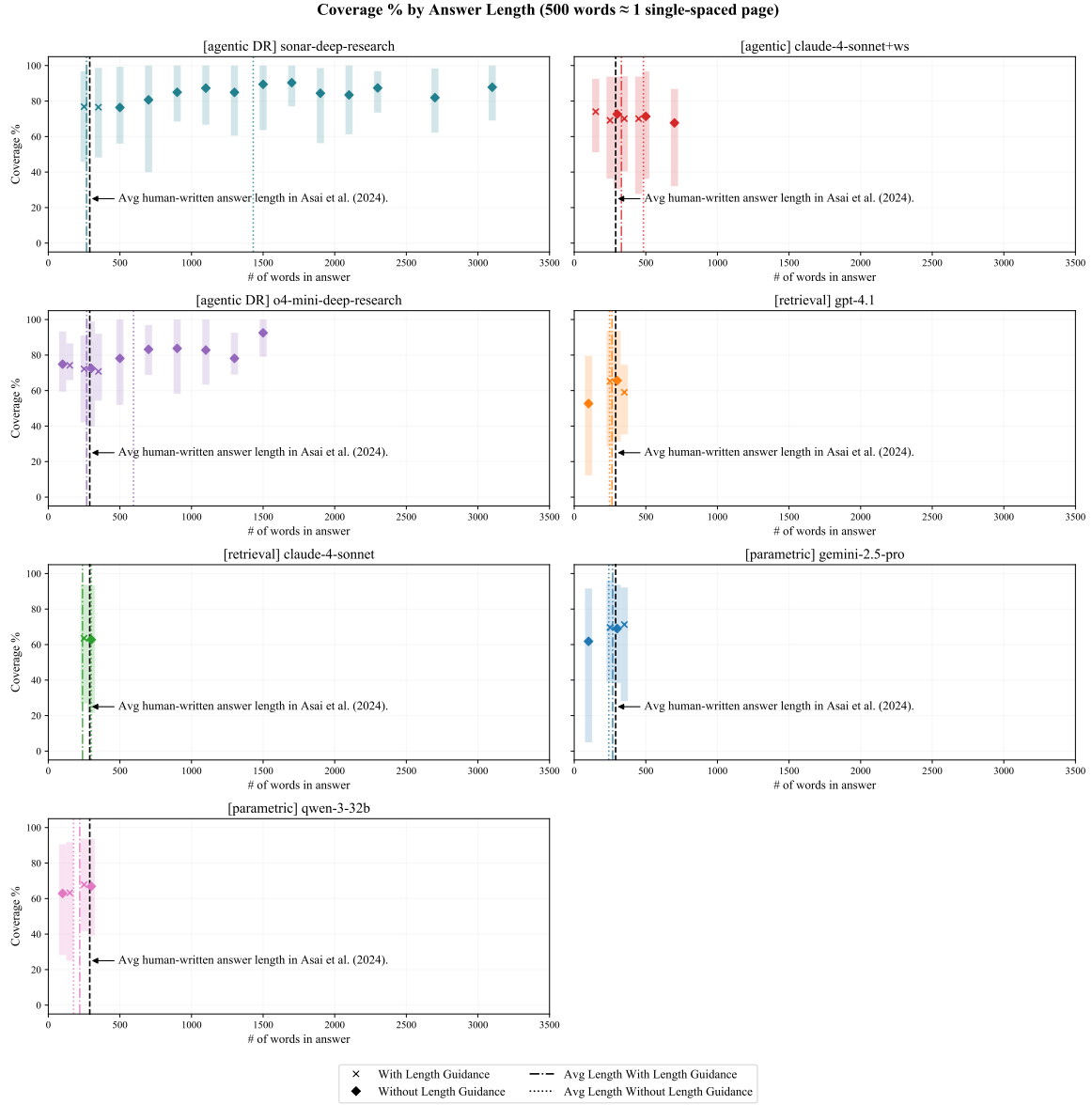


Figure 15: **Coverage % is reliable at 20% sample size of  $\mathcal{D}_{\text{test}}$ .** Mean coverage and 95% bootstrapped confidence interval (green) and standard deviation (red) across varying sample sizes for sonar-deep-research. A 20% sample size (750 queries) can provide a stable estimate of rubric coverage with a standard deviation of approximately 0.6%, suggesting reliable approximation under budget constraints.



**Figure 16: Coverage % vs. Length.** The relation between Coverage % and answer length for the top performers in each family (parametric, retrieval, and agentic) 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.



## B Model Prompts

### B.1 Survey filtering prompt: is literature review

**System:** Determine if the academic paper title indicates a LITERATURE REVIEW.

Answer ONLY “Yes” if the paper’s PRIMARY PURPOSE is to analyze, synthesize, and evaluate research on a topic.

Answer ONLY “No” for all other paper types, including:

1. Papers describing user studies, surveys of people, or field observations
2. Papers that report on events, occurrences, or observations (even if they use “review” in the title)
3. Reports that summarize disease outbreaks, statistics, or time-period data
4. Papers that evaluate or compare methods for conducting literature reviews

IMPORTANT DISTINCTIONS:

- A LITERATURE REVIEW is about analyzing research itself
- Having words like “review,” “survey,” “meta-analysis,” or “overview” in the title is NOT sufficient

**User:** Paper Title: <PAPER\_TITLE>

### B.2 Query generation prompt: hierarchical summarization

**System:** In this task, you will be given a section from an academic paper, split into individual sentences.

You will also receive a set of questions based on a partial section in a hierarchical manner, where each question is clearly answered by the sentence indicated by its index.

You need to generate a question that, within the scope of the hierarchical question set, requires understanding the corresponding part of this section.

Ideally, this question should be based on the sentence, along with sufficient following sentences, as indicated in the hierarchical question set.

Also, provide the supporting sentence indices.

# Content

Paper Title: Societal Biases in Language Generation: Progress and Challenges

Section Title: Bias Definitions and Metrics

Content:

sent1: In the context of AI fairness, the term “bias” commonly refers to skews that result in undesirable impacts (Crawford, 2017) and is quantifiable with some metric.

sent2: There are relatively more existing studies on biases in NLU tasks, where it is arguably simpler to define bias metrics, since we can intuitively compare the accuracy of the task (e.g., coreference resolution, hate speech detection) for different demographics.

sent3: Language generation tasks often involve stochastic generation of open-ended and lengthy texts, traits that are not directly compatible with traditional algorithmic bias definitions (e.g., equalized odds, equal opportunity, demographic parity (Dwork et al., 2012; Hardt et al., 2016)).

sent4: Because of the difficulty in defining metrics, existing works define bias loosely as demographic inequality and use intermediate proxy metrics to comparatively measure bias.

sent5: Examples include: Regard Ratio: negative-neutral-positive regard score ratios of text generated from bias-inducing prompts (Sheng et al., 2019).

sent6: Sentiment Ratio: negative-neutral-positive sentiment score ratios of text generated from African American English (AAE)

versus White-Aligned English (WAE) prompts (Groenwold et al., 2020).

sent7: Individual and Group Fairness through Sentiment: comparisons of the sentiment distributions of generated text across demographics and prompts (Huang et al., 2020).

sent8: Gendered Word Co-occurrence Score: mean and standard deviations of the absolute log ratio of probabilities:  $P(\text{word}|\text{female terms})$  to  $P(\text{word}|\text{male terms})$  across all words in generated text (Bordia and Bowman, 2019).

sent9: There are also metrics for other bias evaluation setups in continuation generation tasks involving sentiment (Shwartz et al., 2020), the ratio of gendered words (Solaiman et al., 2019; Vig et al., 2020; Dinan et al., 2020a), and other novel metrics (Peng et al., 2020; Yeo and Chen, 2020).

sent10: Studies of biases in transformation generation tasks favor metrics of accuracy in terms of successfully transforming text to have a desired property.

sent11: We present a more thorough comparison of metrics in Section 5.4.

sent12: Bias metrics can also be categorized by how they define associations between demographic group attributes and text.

sent13: Biases can be towards people described in text, people who produce the text, or people to whom the text is addressed (Dinan et al., 2020b).

sent14: Most existing works define bias metrics through the first association—these biases are relatively easier to analyze, since both the demographic and the textual signals of bias are encapsulated within the text.

sent15: There are also works that define biases towards people who produce the text (Groenwold et al., 2020) or people to whom the text is addressed (Sheng et al., 2021b), though there are relatively fewer works that study these latter associations.

Output:

1. What is the common definition of “bias” in the context of AI fairness? sent1

2. How are biases in NLU tasks typically defined and measured? sent2

3. Why are traditional algorithmic bias definitions not directly compatible with language generation tasks? sent3

4. How do existing works define and measure bias in language generation tasks? sent4

4.1. What are some examples of intermediate proxy metrics used to measure bias in language generation tasks? sent5, sent6, sent7, sent8

4.2. Are there other metrics for bias evaluation in continuation generation tasks? sent9

4.3. What metrics are favored in transformation generation tasks? sent10

4.4. Where can a more thorough comparison of metrics be found? sent11

5. How can bias metrics be categorized? sent12

6. What are the different associations between demographic group attributes and text in bias metrics? sent13

6.1. Which association is most commonly used in existing works and why? sent14

6.2. Are there works that define biases towards people who produce the text or people to whom the text is addressed? sent15

# Content

Paper Title: Modeling Language Variation and Universals: A Survey on Typological Linguistics for Natural Language Processing  
Section Title: Hand-Crafted Documentation in Typological Databases

Content:

sent1: Typological databases are created manually by linguists.

sent2: They contain taxonomies of typological features, their possible values, as well as the documentation of feature values for the world’s languages.

sent3: Major typological databases, listed in Table 1, typically organize linguistic information in terms of universal features and language-specific values.

sent4: For example, Figure 3 presents language-specific values for

the feature number of grammatical genders for nouns on a world map.

sent5: Note that each language is color-coded according to its value.  
sent6: Further examples for each database can be found in the rightmost column of Table 1.

sent7: Some databases store information pertaining to multiple levels of linguistic description.

sent8: These include WALS (Dryer and Haspelmath 2013) and the Atlas of Pidgin and Creole Language Structures (APiCS) (Michaelis et al. 2013).

sent9: Among all presently available databases, WALS has been the most widely used in NLP.

sent10: In this resource, which has 142 typological features in total, features 1–19 deal with phonology, 20–29 with morphology, 30–57 with nominal categories, 58–64 with nominal syntax, 65–80 with verbal categories, 81–97 and 143–144 with word order, 98–121 with simple clauses, 122–128 with complex sentences, 129–138 with the lexicon, and 139–142 with other properties.

sent11: Other databases only cover features related to a specific level of linguistic description.

sent12: For example, both Syntactic Structures of the World's Languages (SSWL) (Collins and Kayne 2009) and AUTOTYP (Bickel et al. 2017) focus on syntax.

sent13: SSWL features are manually crafted, whereas AUTOTYP features are derived automatically from primary linguistic data using scripts.

sent14: The Valency Patterns Leipzig (ValPaL) (Hartmann, Haspelmath, and Taylor 2013) provides verbs as attributes and predicate–argument structures as their values (including both valency and morphosyntactic constraints).

sent15: For example, in both Mandinka and Sliammon, the verb to laugh has a valency of 1; in other words, it requires only one mandatory argument, the subject.

sent16: In Mandinka the subject precedes the verb, but there is no agreement requirement; in Sliammon, on the other hand, the word order does not matter, but the verb is required to morphologically agree with the subject.

sent17: For phonology, the Phonetics Information Base and Lexicon (PHOIBLE) (Moran, McCloy, and Wright 2014) collates information on segments (binary phonetic features).

sent18: In the Lyon–Albuquerque Phonological Systems Database (LAPSYD) (Maddieson et al. 2013), attributes are articulatory traits, syllabic structures, or tonal systems.

sent19: Finally, StressTyp2 (Goedemans, Heinz, and der Hulst 2014) deals with stress and accent patterns.

sent20: For instance, in Koromfé each word's first syllable has to be stressed, but not in Cubeo.

sent21: Other databases document various aspects of semantics.

sent22: The World Loanword Database (WOLD) (Haspelmath and Tadmor 2009) documents loanwords by identifying the donor languages and the source words.

sent23: The Automated Similarity Judgment Program (ASJP) (Wichmann, Holman, and Brown 2016) and the Intercontinental Dictionary Series (IDS) (Key and Comrie 2015) indicate how a meaning is lexicalized across languages:

sent24: For example, the concept of WORLD is expressed as *mir* in Russian, and as *ārkišos* *i* in Tocharian A.

sent25: Although typological databases store abundant information on many languages, they suffer from shortcomings that limit their usefulness.

sent26: Perhaps the most significant shortcoming of such resources is their limited coverage.

sent27: In fact, feature values are missing for most languages in most databases.

sent28: Other shortcomings are related to feature granularity.

sent29: In particular, most databases fail to account for feature value variation within each language: They report only majority value rather than the full range of possible values and their corresponding frequencies.

sent30: For example, the dominant adjective–noun word order in Italian is adjective before noun; however, the opposite order is also attested.

sent31: The latter information is often missing from typological databases.

sent32: Further challenges are posed by restricted feature applicability and feature hierarchies.

sent33: Firstly, some features apply, by definition, only to subsets of languages that share another feature value.

sent34: For instance, WALS feature 113A documents “Symmetric and Asymmetric Standard Negation,” whereas WALS feature 114A “Subtypes of Asymmetric Standard Negation.”

sent35: Although a special NA value is assigned for symmetric-negation languages in the latter, there are cases where languages without the prerequisite feature are simply omitted from the sample.

sent36: Secondly, features can be partially redundant, and subsume other features.

sent37: For instance, WALS feature 81A “Order of Subject, Object and Verb” encodes the same information as WALS feature 82A “Order of Subject and Verb” and 83A “Order of Object and Verb,” with the addition of the order of subject and object.

Output:

1. What are typological databases and how are they created? sent1

1.1. What do typological databases contain? sent2

1.2. How is linguistic information organized in major typological databases? sent3

1.2.1. Can you provide an example of how linguistic information is organized? sent4, sent5

1.2.2. Where can further examples for each database be found? sent6

1.3. Do some databases store information at multiple levels of linguistic description? sent7

1.3.1. Which databases store information at multiple levels of linguistic description? sent8

1.3.2. Which database is the most widely used in NLP? sent9

1.3.2.1. What features does WALS cover? sent10

1.4. Do other databases cover specific levels of linguistic description? sent11

1.4.1. Can you provide examples of databases that focus on specific levels of linguistic description? sent12

1.4.1.1. How are features in SSWL and AUTOTYP crafted? sent13

1.4.1.2. What does the Valency Patterns Leipzig (ValPaL) provide? sent14

1.4.1.2.1. Can you provide an example of valency in different languages? sent15, sent16

1.4.1.3. What information does PHOIBLE collate? sent17

1.4.1.4. What attributes are included in LAPSYD? sent18

1.4.1.5. What does StressTyp2 deal with? sent19

1.4.1.5.1. Can you provide an example of stress patterns in different languages? sent20

1.5. Do other databases document aspects of semantics? sent21

1.5.1. What does the World Loanword Database (WOLD) document? sent22

1.5.2. What do ASJP and IDS indicate? sent23

1.5.2.1. Can you provide an example of how a meaning is lexicalized across languages? sent24

2. What are the shortcomings of typological databases? sent25

2.1. What is the most significant shortcoming? sent26

2.1.1. What is the evidence for the limited coverage of typological databases? sent27

2.2. What are other shortcomings related to? sent28

2.2.1. How do most databases fail to account for feature value variation within each language? sent29

2.2.1.1. Can you provide an example of missing information in typological databases? sent30, sent31

2.3. What further challenges are posed by typological databases? sent32

2.3.1. What is the first challenge related to? sent33

2.3.1.1. Can you provide an example of restricted feature applicability? sent34, sent35

2.3.2. What is the second challenge related to? sent36

2.3.2.1. Can you provide an example of feature redundancy and subsumption? sent37

# Content

Paper Title: Efficient Methods for Natural Language Processing: A Survey

Section Title: Sparse Modeling

Content:

sent1: To leverage sparsity for efficiency, many models follow the mixture-of-experts (MoE) concept (Jacobs et al., 1991; Shazeer et al., 2017; Fedus et al., 2022a), which routes computation through small subnetworks instead of passing the input through the entire model.

sent2: Relevant works on this line include GShard (Lepikhin et al., 2021), Switch Transformer (Fedus et al., 2022b), and ST-MoE (Zoph et al., 2022), which replace the feed forward layers in transformers with MoE layers.

sent3: More recently, Rajbhandari et al. (2022) scaled transformers up by compressing and optimizing the usage of MoE.

sent4: Overall, MoE models have been shown to achieve strong performance across several NLP tasks while reducing the overall resource consumption (Sec. 8).

sent5: For instance, GLaM (Du et al., 2022) used only  $\sim 1/3$  of GPT-3's energy consumption (with additional hardware-based optimization), while Rajbhandari et al. (2022) reached a 5x reduction in terms of training cost.

sent6: However, MoE models have also exhibited training instabilities in practice, and may require architecture-specific implementation (Zoph et al., 2022; Mustafa et al., 2022).

sent7: Another promising direction for exploiting sparse modeling is Sparsefinder (Treviso et al., 2022), which extends the Adaptively Sparse Transformer (Correia et al., 2019) to allow a more efficient attention mechanism by identifying beforehand the sparsity pattern returned by entmax attention—a sparse alternative to (dense) softmax attention (Peters et al., 2019).

sent8: Finally, sparsity can also be induced via modularity, e.g., by encapsulating task-specific parameters (Ponti et al., 2022).

Output:

1. How do models leverage sparsity for efficiency, such as MoE method? sent1

1.1. What are some relevant works that follow the MoE concept? sent2

1.1.1 How have recent works scaled transformers using MoE? sent3

1.2. What is the overall performance and resource consumption of MoE models? sent4

1.2.1. Can you provide specific examples of resource consumption reduction in MoE models? sent5

1.2.2. What are some challenges associated with MoE models? sent6

1.3. What is another promising direction for exploiting sparse modeling? sent7

1.4. How can sparsity be induced via modularity? sent8

# Content

Paper Title: <PAPER\_TITLE>

Section Title: <SECTION\_TITLE>

Content:

<SECTION\_SENTENCE\_PREFIXED>

Output:

### B.3 Query generation prompt: generate initial query

**System:** In this task, you will be given a section from an academic paper, split up into individual sentences.

Also a set of questions based on a partial section in a hierarchical manner, where each question is clearly answered by the sentence indicated by the index.

You need to generate a question that within the scope of the hierarchical question set, requires understanding the corresponding part of this section.

Ideally, this question is based on the sentence that with sufficient following sentences, indicated in the hierarchical question set.

Also provide the supprting sentence indexes.

This question needs to be:

1. Unambiguous: Clearly framed so it does not require follow-up questions for clarification.

1.1 It should be understandable to any expert without needing specific context or jargon found in the given section.

2. Natural: Phrased as if it is asked by a domain expert conducting research.

3. Answerable: Should be entirely answerable based on the provided section.

4. Precise: Question should express a clear information need and not be vague.

4.1 Multiple experts should answer such a question in a similar way.

4.2 The question may need to mention the sub-area and domain specified in the section (especially titles) to be precise.

5. Requires a long-form, comprehensive answer: not simply extractive or yes-no questions.

6. The question should not specifically about a particular paper e.g. Artetxe and Schwenk (2019).

7. The question should be based on the sentence that with sufficient following sentences, indicated in the hierarchical question set.

8. Less than 20 words.

# Content:

Paper Title: Societal Biases in Language Generation: Progress and Challenges

Section Title: Bias Definitions and Metrics

Section content:

sent1: In the context of AI fairness, the term “bias” commonly refers to skews that result in undesirable impacts (Crawford, 2017) and is quantifiable with some metric.

sent2: There are relatively more existing studies on biases in NLU tasks, where it is arguably simpler to define bias metrics, since we can intuitively compare the accuracy of the task (e.g., coreference resolution, hate speech detection) for different demographics.

sent3: Language generation tasks often involve stochastic generation of open-ended and lengthy texts, traits that are not directly compatible with traditional algorithmic bias definitions (e.g., equalized odds, equal opportunity, demographic parity (Dwork et al., 2012; Hardt et al., 2016)).

sent4: Because of the difficulty in defining metrics, existing works define bias loosely as demographic inequality and use intermediate proxy metrics to comparatively measure bias.

sent5: Examples include: Regard Ratio: negative-neutral-positive regard score ratios of text generated from bias-inducing prompts (Sheng et al., 2019).

sent6: Sentiment Ratio: negative-neutral-positive sentiment score ratios of text generated from African American English (AAE) versus White-Aligned English (WAE) prompts (Groenwold et al., 2020).

sent7: Individual and Group Fairness through Sentiment: comparisons of the sentiment distributions of generated text across demographics and prompts (Huang et al., 2020).

sent8: Gendered Word Co-occurrence Score: mean and standard deviations of the absolute log ratio of probabilities:  $P(\text{word}|\text{female terms})$  to  $P(\text{word}|\text{male terms})$  across all words in generated text (Bordia and Bowman, 2019).

sent9: There are also metrics for other bias evaluation setups in continuation generation tasks involving sentiment (Shwartz et al., 2020), the ratio of gendered words (Solaiman et al., 2019; Vig et al., 2020; Dinan et al., 2020a), and other novel metrics (Peng et al., 2020; Yeo and Chen, 2020).

sent10: Studies of biases in transformation generation tasks favor metrics of accuracy in terms of successfully transforming text to have a desired property.

sent11: We present a more thorough comparison of metrics in Section 5.4.

sent12: Bias metrics can also be categorized by how they define associations between demographic group attributes and text.

sent13: Biases can be towards people described in text, people who produce the text, or people to whom the text is addressed (Dinan et al., 2020b).

sent14: Most existing works define bias metrics through the first

association—these biases are relatively easier to analyze, since both the demographic and the textual signals of bias are encapsulated within the text.

sent15: There are also works that define biases towards people who produce the text (Groenwold et al., 2020) or people to whom the text is addressed (Sheng et al., 2021b), though there are relatively fewer works that study these latter associations.

Hierarchical question set:

1. What is the common definition of “bias” in the context of AI fairness? sent1

2. How are biases in NLU tasks typically defined and measured? sent2

3. Why are traditional algorithmic bias definitions not directly compatible with language generation tasks? sent3

4. How do existing works define and measure bias in language generation tasks? sent4

4.1. What are some examples of intermediate proxy metrics used to measure bias in language generation tasks? sent5, sent6, sent7, sent8

4.2. Are there other metrics for bias evaluation in continuation generation tasks? sent9

4.3. What metrics are favored in transformation generation tasks? sent10

4.4. Where can a more thorough comparison of metrics be found? sent11

5. How can bias metrics be categorized? sent12

6. What are the different associations between demographic group attributes and text in bias metrics? sent13

6.1. Which association is most commonly used in existing works and why? sent14

6.2. Are there works that define biases towards people who produce the text or people to whom the text is addressed? sent15

Output:

Question: When generating text, how are the major types of bias measures used to evaluate bias with respect to gender?

Supporting sentence: sent4, sent5, sent6, sent7, sent8, sent9

#### # Content

Paper Title: Modeling Language Variation and Universals: A Survey on Typological Linguistics for Natural Language Processing  
Section Title: Hand-Crafted Documentation in Typological Databases

Section content:

sent1: Typological databases are created manually by linguists.

sent2: They contain taxonomies of typological features, their possible values, as well as the documentation of feature values for the world’s languages.

sent3: Major typological databases, listed in Table 1, typically organize linguistic information in terms of universal features and language-specific values.

sent4: For example, Figure 3 presents language-specific values for the feature number of grammatical genders for nouns on a world map.

sent5: Note that each language is color-coded according to its value.

sent6: Further examples for each database can be found in the rightmost column of Table 1.

sent7: Some databases store information pertaining to multiple levels of linguistic description.

sent8: These include WALS (Dryer and Haspelmath 2013) and the Atlas of Pidgin and Creole Language Structures (APiCS) (Michaelis et al. 2013).

sent9: Among all presently available databases, WALS has been the most widely used in NLP.

sent10: In this resource, which has 142 typological features in total, features 1–19 deal with phonology, 20–29 with morphology, 30–57 with nominal categories, 58–64 with nominal syntax, 65–80 with verbal categories, 81–97 and 143–144 with word order, 98–121 with simple clauses, 122–128 with complex sentences, 129–138 with the lexicon, and 139–142 with other properties.

sent11: Other databases only cover features related to a specific level of linguistic description.

sent12: For example, both Syntactic Structures of the World’s

Languages (SSWL) (Collins and Kayne 2009) and AUTOTYP (Bickel et al. 2017) focus on syntax.

sent13: SSWL features are manually crafted, whereas AUTOTYP features are derived automatically from primary linguistic data using scripts.

sent14: The Valency Patterns Leipzig (ValPaL) (Hartmann, Haspelmath, and Taylor 2013) provides verbs as attributes and predicate–argument structures as their values (including both valency and morphosyntactic constraints).

sent15: For example, in both Mandinka and Sliammon, the verb to laugh has a valency of 1; in other words, it requires only one mandatory argument, the subject.

sent16: In Mandinka the subject precedes the verb, but there is no agreement requirement; in Sliammon, on the other hand, the word order does not matter, but the verb is required to morphologically agree with the subject.

sent17: For phonology, the Phonetics Information Base and Lexicon (PHOIBLE) (Moran, McCloy, and Wright 2014) collates information on segments (binary phonetic features).

sent18: In the Lyon–Albuquerque Phonological Systems Database (LAPSyD) (Maddieson et al. 2013), attributes are articulatory traits, syllabic structures, or tonal systems.

sent19: Finally, StressTyp2 (Goedemans, Heinz, and der Hulst 2014) deals with stress and accent patterns.

sent20: For instance, in Koromfé each word’s first syllable has to be stressed, but not in Cubeo.

sent21: Other databases document various aspects of semantics.

sent22: The World Loanword Database (WOLD) (Haspelmath and Tadmor 2009) documents loanwords by identifying the donor languages and the source words.

sent23: The Automated Similarity Judgment Program (ASJP) (Wichmann, Holman, and Brown 2016) and the Intercontinental Dictionary Series (IDS) (Key and Comrie 2015) indicate how a meaning is lexicalized across languages:

sent24: For example, the concept of WORLD is expressed as *mir* in Russian, and as *ārkišos* in Tocharian A.

sent25: Although typological databases store abundant information on many languages, they suffer from shortcomings that limit their usefulness.

sent26: Perhaps the most significant shortcoming of such resources is their limited coverage.

sent27: In fact, feature values are missing for most languages in most databases.

sent28: Other shortcomings are related to feature granularity.

sent29: In particular, most databases fail to account for feature value variation within each language: They report only majority value rather than the full range of possible values and their corresponding frequencies.

sent30: For example, the dominant adjective–noun word order in Italian is adjective before noun; however, the opposite order is also attested.

sent31: The latter information is often missing from typological databases.

sent32: Further challenges are posed by restricted feature applicability and feature hierarchies.

sent33: Firstly, some features apply, by definition, only to subsets of languages that share another feature value.

sent34: For instance, WALS feature 113A documents “Symmetric and Asymmetric Standard Negation,” whereas WALS feature 114A “Subtypes of Asymmetric Standard Negation.”

sent35: Although a special NA value is assigned for symmetric-negation languages in the latter, there are cases where languages without the prerequisite feature are simply omitted from the sample.

sent36: Secondly, features can be partially redundant, and subsume other features.

sent37: For instance, WALS feature 81A “Order of Subject, Object and Verb” encodes the same information as WALS feature 82A “Order of Subject and Verb” and 83A “Order of Object and Verb,” with the addition of the order of subject and object.

Hierarchical question set:

1. What are typological databases and how are they created? sent1

1.1. What do typological databases contain? sent2



1.2. How is linguistic information organized in major typological databases? sent3

1.2.1. Can you provide an example of how linguistic information is organized? sent4, sent5

1.2.2. Where can further examples for each database be found? sent6

1.3. Do some databases store information at multiple levels of linguistic description? sent7

1.3.1. Which databases store information at multiple levels of linguistic description? sent8

1.3.2. Which database is the most widely used in NLP? sent9

1.3.2.1. What features does WALS cover? sent10

1.4. Do other databases cover specific levels of linguistic description? sent11

1.4.1. Can you provide examples of databases that focus on specific levels of linguistic description? sent12

1.4.1.1. How are features in SSWL and AUTOTYP crafted? sent13

1.4.1.2. What does the Valency Patterns Leipzig (ValPaL) provide? sent14

1.4.1.2.1. Can you provide an example of valency in different languages? sent15, sent16

1.4.1.3. What information does PHOIBLE collate? sent17

1.4.1.4. What attributes are included in LAPSyD? sent18

1.4.1.5. What does StressTyp2 deal with? sent19

1.4.1.5.1. Can you provide an example of stress patterns in different languages? sent20

1.5. Do other databases document aspects of semantics? sent21

1.5.1. What does the World Loanword Database (WOLD) document? sent22

1.5.2. What do ASJP and IDS indicate? sent23

1.5.2.1. Can you provide an example of how a meaning is lexicalized across languages? sent24

2. What are the shortcomings of typological databases? sent25

2.1. What is the most significant shortcoming? sent26

2.1.1. What is the evidence for the limited coverage of typological databases? sent27

2.2. What are other shortcomings related to? sent28

2.2.1. How do most databases fail to account for feature value variation within each language? sent29

2.2.1.1. Can you provide an example of missing information in typological databases? sent30, sent31

2.3. What further challenges are posed by typological databases? sent32

2.3.1. What is the first challenge related to? sent33

2.3.1.1. Can you provide an example of restricted feature applicability? sent34, sent35

2.3.2. What is the second challenge related to? sent36

2.3.2.1. Can you provide an example of feature redundancy and subsumption? sent37

Output:

Question: What are the differences between publicly available linguistic typology databases?

Supporting sentence: sent7, sent8, sent9, sent10, sent11, sent12, sent13, sent14, sent17, sent18, sent19, sent21, sent22, sent23

# Content

Paper Title: Efficient Methods for Natural Language Processing: A Survey

Section Title: Sparse Modeling

Section content:

sent1: To leverage sparsity for efficiency, many models follow the mixture-of-experts (MoE) concept (Jacobs et al., 1991; Shazeer et al., 2017; Fedus et al., 2022a), which routes computation through small subnetworks instead of passing the input through the entire model.

sent2: Relevant works on this line include GShard (Lepikhin et al., 2021), Switch Transformer (Fedus et al., 2022b), and ST-MoE (Zoph et al., 2022), which replace the feed forward layers in transformers with MoE layers.

sent3: More recently, Rajbhandari et al. (2022) scaled transformers up by compressing and optimizing the usage of MoE.

sent4: Overall, MoE models have been shown to achieve strong

performance across several NLP tasks while reducing the overall resource consumption (Sec. 8).

sent5: For instance, GLaM (Du et al., 2022) used only  $\sim 1/3$  of GPT-3's energy consumption (with additional hardware-based optimization), while Rajbhandari et al. (2022) reached a 5x reduction in terms of training cost.

sent6: However, MoE models have also exhibited training instabilities in practice, and may require architecture-specific implementation (Zoph et al., 2022; Mustafa et al., 2022).

sent7: Another promising direction for exploiting sparse modeling is Sparsefinder (Treviso et al., 2022), which extends the Adaptively Sparse Transformer (Correia et al., 2019) to allow a more efficient attention mechanism by identifying beforehand the sparsity pattern returned by entmax attention—a sparse alternative to (dense) softmax attention (Peters et al., 2019).

sent8: Finally, sparsity can also be induced via modularity, e.g., by encapsulating task-specific parameters (Ponti et al., 2022).

Hierarchical question set:

1. How do models leverage sparsity for efficiency, such as MoE method? sent1

1.1. What are some relevant works that follow the MoE concept? sent2

1.1.1 How have recent works scaled transformers using MoE? sent3

1.2. What is the overall performance and resource consumption of MoE models? sent4

1.2.1. Can you provide specific examples of resource consumption reduction in MoE models? sent5

1.2.2. What are some challenges associated with MoE models? sent6

1.3. What is another promising direction for exploiting sparse modeling? sent7

1.4. How can sparsity be induced via modularity? sent8

Output:

Question: How can we utilize sparsity to enhance efficiency in designing NLP models?

Supporting sentence: sent1, sent2, sent3, sent4, sent5, sent6, sent7, sent8

# Content:

Paper Title: <PAPER\_TITLE>

Section Title: <SECTION\_TITLE>

Section content:

<SECTION\_CONTENT\_SENTENCE\_PREFIXED>

Hierarchical question set:

<HIERARCHICAL\_SUMMARY>

Output:

## B.4 Query filtering prompt: self-containment

**System:** You will be given a academic question.

Your task is to rate the question on one metric.

Please make sure you read and understand these instructions carefully.

Evaluation Criteria:

Score -1: The question is hard to understand from your perspective and cannot be judged.

Score 1~2: Entirely inappropriate: The question assumes a lot of context or prior knowledge unlikely to be had by a junior researcher.

Score 3~4: Mostly inappropriate: The question mostly assumes context or prior knowledge unlikely to be had, but some elements are appropriate.

Score 5~6: Mixed appropriateness: There are elements that are appropriate and inappropriate, containing a roughly equal mix.

Score 7~8: Mostly appropriate: The question mostly assumes context and prior knowledge that is likely to be had by a junior researcher.

Score 9~10: Entirely appropriate: The question could definitely be asked by a junior researcher.



You should score 1~4, including but not limited to the scenarios:

- (1) the question contains undefined jargon or abbreviations that a junior researcher is unlikely to understand (excluding abbreviations that are considered common knowledge within this domain).
- (2) the question covers advanced topics that a junior researcher is unlikely to be familiar with.
- (3) the question is highly contextualized or includes unusual details or conclusions that would require specific context to make sense.

You should score 7~10, including but not limited to the scenarios:

- (1) the question is general and involves topics that a junior researcher is likely to be familiar with.
- (2) the question is well-understood independently, without any additional context provided.

You should not take the length of question into account. The length of question has nothing to do with the metric.

You should be more critical: try to make the best judgment of whether the question leans toward being appropriate or inappropriate for a junior researcher, instead of choosing a safe mediocre score 5~6.

You should be more critical: try to make the best judgment of whether the question leans toward being appropriate or inappropriate for a junior researcher, instead of choosing a safe mediocre score 5~6.

You should be more critical: try to make the best judgment of whether the question leans toward being appropriate or inappropriate for a junior researcher, instead of choosing a safe mediocre score 5~6.

Question: What is the application of computer vision in the field of language processing?

Standalone (scores ONLY): -1

Question: What are the characteristics and findings related to BERT's contextualized embeddings?

Standalone (scores ONLY): 10

Question: What are the main differences between graph-based and transition-based dependency parsers?

Standalone (scores ONLY): 8

Question: How do recent approaches for distant supervision in NLP handle the negative effects of noisy labels?

Standalone (scores ONLY): 6

Question: How do existing dialogue models for social influence utilize templates, retrieval methods, and conditional generation to produce system responses?

Standalone (scores ONLY): 3

Question: How do different assumptions affect the recognizability of PARITY by transformer encoders?

Standalone (scores ONLY): 1

Question: <QUESTION>

Standalone (scores ONLY):

## B.5 Query filtering prompt: answer variability

**System:** You will be given a academic question. Your task is to rate the question on one metric. Please make sure you read and understand these instructions carefully.

Evaluation Criteria:

Please rate how likely it would be for experts in the area to provide

very different responses to the question.

Please ignore possible phrasing differences, but instead focus on comparing what information and papers the experts are likely to mention in their response.

Assume they are both instructed to provide similar length responses. Evaluate on a scale of 1 to 10, how many possible answers could be given by senior sub-area experts. Use the following definition for the scale.

Score -1: The question is difficult to impossible to imagine an answer to, and so cannot be judged.

Score 1~2: Low variability: Most experts are likely to respond with very similar information.

Score 3~4: Moderate variability: Much of the information they respond with will be similar, but may have a few differences.

Score 5~6: Shared core, but variable: The responses are likely to have a shared core, but then each response is likely to have a piece unique to them.

Score 7~8: Mostly Variable: The responses are likely to have overlap in information but we expect each response to have large amounts of unique information.

Score 9~10: Highly Variable: It is likely that the responses will share little to no information in common.

You should score 1~4, including but not limited to the scenarios:

- (1) the question is asking about some specific techniques details in a narrow setting.

You should score 7~10, including but not limited to the scenarios:

- (1) the question merely requires to list a bunch of facts or resources, e.g. "What are the datasets ...", "What are the benchmarks ...", "What are the applications of ...".
- (2) the question asks a overly general method, application, etc. in a broad setting.

You should not take the length of question into account. The length of question has nothing to do with the metric.

You should be more critical: trying to make the best judge of whether it leans to low or high variability, instead of choosing a safe mediocre score 5~6.

You should be more critical: trying to make the best judge of whether it leans to low or high variability, instead of choosing a safe mediocre score 5~6.

You should be more critical: trying to make the best judge of whether it leans to low or high variability, instead of choosing a safe mediocre score 5~6.

Question: What is the application of computer vision in the field of language processing?

Answer Variability (scores ONLY): -1

Question: What are the main differences between graph-based and transition-based dependency parsers?

Answer Variability (scores ONLY): 1

Question: What are the methods for data augmentation in NLP at different levels of text granularity?

Answer Variability (scores ONLY): 3

Question: What are the major challenges in addressing Conversational Machine Comprehension (CMC)?

Answer Variability (scores ONLY): 5

Question: What are the characteristics and findings related to BERT's contextualized embeddings?

Answer Variability (scores ONLY): 8

Question: What are the popular NLP tasks that utilize knowledge graphs (KGs)?

Answer Variability (scores ONLY): 10

Question: <INITIAL\_QUERY>

Answer Variability (scores ONLY):

## B.6 Query filtering prompt: contains citation

**System:** Does the question <INITIAL\_QUERY> contain citation or author name? Only reply YES or NO:

## B.7 Query generation prompt: generate initial reference answer

**System:** In this task, you will be given a section from an academic paper with a question, and you need to generate an answer using and only using the contents of the section w.r.t. the question.

This answer needs to be:

1. Decontextualized: remove the contents anchored in external tables, figures, sections, etc., and rephrase the first-person expressions as third person.
2. Relevant: remove all other irrelevant details that do not answer the question
3. No new information: do not include any information that is not mentioned in the content, except for necessary connective words
4. Content preservative: retain the relevant information in the content as closely as possible to the original
5. Citation preservative: keep the relevant inline citations in the content exactly the same as in the original content, such as (Author, Year), (Author), Author (Year), or [1], etc.
6. Structure preservative: maintain the logical structure of the content such as the order, lists, steps, etc., as in the original text.
7. Display the answer sentence by sentence.

# Given extracted content:

Paper Title: Societal Biases in Language Generation: Progress and Challenges

Section Title: Bias Definitions and Metrics

Section Content:

In the context of AI fairness, the term “bias” commonly refers to skews that result in undesirable impacts (Crawford, 2017) and is quantifiable with some metric. There are relatively more existing studies on biases in NLU tasks, where it is arguably simpler to define bias metrics, since we can intuitively compare the accuracy of the task (e.g., coreference resolution, hate speech detection) for different demographics. Language generation tasks often involve stochastic generation of open-ended and lengthy texts, traits that are not directly compatible with traditional algorithmic bias definitions (e.g., equalized odds, equal opportunity, demographic parity (Dwork et al., 2012; Hardt et al., 2016)).

Because of the difficulty in defining metrics, existing works define bias loosely as demographic inequality and use intermediate proxy metrics to comparatively measure bias. Examples include:

- Regard Ratio: negative-neutral-positive regard score ratios of text generated from bias-inducing prompts (Sheng et al., 2019)
- Sentiment Ratio: negative-neutral-positive sentiment score ratios of text generated from African American English (AAE) versus White-Aligned English (WAE) prompts (Groenwold et al., 2020)
- Individual and Group Fairness through Sentiment: comparisons of the sentiment distributions of generated text across demographics and prompts (Huang et al., 2020)
- Gendered Word Co-occurrence Score: mean and standard deviations of the absolute log ratio of probabilities:  $P(\text{word}|\text{female terms})$  to  $P(\text{word}|\text{male terms})$  across all words in generated text (Bordia and Bowman, 2019)

There are also metrics for other bias evaluation setups in continuation generation tasks involving sentiment (Shwartz et al., 2020), the ratio of gendered words (Solaiman et al., 2019; Vig et al., 2020; Dinan et al., 2020a), and other novel metrics (Peng et al., 2020; Yeo and Chen, 2020). Studies of biases in transformation

generation tasks favor metrics of accuracy in terms of successfully transforming text to have a desired property. We present a more thorough comparison of metrics in Section 5.4.

Bias metrics can also be categorized by how they define associations between demographic group attributes and text. Biases can be towards people described in text, people who produce the text, or people to whom the text is addressed (Dinan et al., 2020b). Most existing works define bias metrics through the first association—these biases are relatively easier to analyze, since both the demographic and the textual signals of bias are encapsulated within the text. There are also works that define biases towards people who produce the text (Groenwold et al., 2020) or people to whom the text is addressed (Sheng et al., 2021b), though there are relatively fewer works that study these latter associations.

Question: When generating text, how are the major types of bias measures used to evaluate bias with respect to gender?

Answer:

Bias metrics can be categorized by how they define associations between demographic group attributes and text.

Biases can be towards people described in text, people who produce the text, or people to whom the text is addressed (Dinan et al., 2020b).

Most existing works define bias metrics through the first association—these biases are relatively easier to analyze, since both the demographic and the textual signals of bias are encapsulated within the text.

There are also works that define biases towards people who produce the text (Groenwold et al., 2020) or people to whom the text is addressed (Sheng et al., 2021b), though there are relatively fewer works that study these latter associations.

Language generation tasks often involve stochastic generation of open-ended and lengthy texts, traits that are not directly compatible with traditional algorithmic bias definitions (e.g., equalized odds, equal opportunity, demographic parity (Dwork et al., 2012; Hardt et al., 2016)).

Because of the difficulty in defining metrics, existing works define bias loosely as demographic inequality and use intermediate proxy metrics to comparatively measure bias. Examples include:

- Regard Ratio: negative-neutral-positive regard score ratios of text generated from bias-inducing prompts (Sheng et al., 2019)
- Sentiment Ratio: negative-neutral-positive sentiment score ratios of text generated from African American English (AAE) versus White-Aligned English (WAE) prompts (Groenwold et al., 2020)
- Individual and Group Fairness through Sentiment: comparisons of the sentiment distributions of generated text across demographics and prompts (Huang et al., 2020)
- Gendered Word Co-occurrence Score: mean and standard deviations of the absolute log ratio of probabilities:  $P(\text{word}|\text{female terms})$  to  $P(\text{word}|\text{male terms})$  across all words in generated text (Bordia and Bowman, 2019)

There are also metrics for other bias evaluation setups in continuation generation tasks involving sentiment (Shwartz et al., 2020), the ratio of gendered words (Solaiman et al., 2019; Vig et al., 2020; Dinan et al., 2020a), and other novel metrics (Peng et al., 2020; Yeo and Chen, 2020).

# Given extracted content:

Paper Title: A Survey on Contextual Embeddings

Section Title: Cross-lingual Polyglot Pre-training for Contextual Embeddings

Section Content:

Cross-lingual polyglot pre-training aims to learn joint multi-lingual representations, enabling knowledge transfer from data-rich languages like English to data-scarce languages like Romanian. Based on whether joint training and a shared vocabulary are used, we divide previous work into three categories.

Joint training & shared vocabulary. Artetxe and Schwenk (2019) use a BiLSTM encoder-decoder framework with a shared BPE vocabulary for 93 languages. The framework is pre-trained using parallel corpora, including as Europarl and Tanzil. The contextual embeddings from the encoder are used to train classifiers using English corpora for downstream tasks. As the embedding space and the encoder are shared, the resultant classifiers can be transferred to

any of the 93 languages without further modification. Experiments show that these classifiers achieve competitive performance on cross-lingual natural language inference, cross-lingual document classification, and parallel corpus mining.

Rosita (Mulcaire et al., 2019) pre-trains a language model using text from different languages, showing the benefits of polyglot learning on low-resource languages.

Recently, the authors of BERT developed a multi-lingual BERT which is pre-trained using the Wikipedia dump with more than 100 languages.

XLM (Lample and Conneau, 2019) uses three pre-training methods for learning cross-lingual language models: (1) Causal language modelling, where the model is trained to predict  $p(\text{token}_t | \text{tokens}_{1:t-1})$ , (2) Masked language modelling, and (3) Translation language modelling (TLM). Parallel corpora are used, and tokens in both source and target sequences are masked for learning cross-lingual association. XLM performs strongly on cross-lingual classification, unsupervised machine translation, and supervised machine translation. XLM-R (Conneau et al., 2019) scales up XLM by training a Transformer-based masked language model on one hundred languages, using more than two terabytes of filtered CommonCrawl data. XLM-R shows that large-scale multi-lingual pre-training leads to significant performance gains for a wide range of cross-lingual transfer tasks.

Joint training & separate vocabularies. Wu et al. (2019) study the emergence of cross-lingual structures in pre-trained multi-lingual language models. It is found that cross-lingual transfer is possible even when there is no shared vocabulary across the monolingual corpora, and there are universal latent symmetries in the embedding spaces of different languages.

Separate training & separate vocabularies. Artetxe et al. (2019) use a four-step method for obtaining multi-lingual embeddings. Suppose we have the monolingual sequences of two languages L1 and L2: (1) Pre-training BERT with the vocabulary of L1 using L1's monolingual data. (2) Replacing the vocabulary of L1 with the vocabulary of L2 and training new vocabulary embeddings, while freezing the other parameters, using L2's monolingual data. (3) Fine-tuning the BERT model for a downstream task using labeled data in L1, while freezing L1's vocabulary embeddings. (4) Replacing the fine-tuned BERT with L2's vocabulary embeddings for zero-shot transfer tasks.

Question: How do multilingual NLP models handle joint vocabularies during pretraining?

Answer:

Cross-lingual polyglot pre-training aims to learn joint multi-lingual representations, enabling knowledge transfer from data-rich languages like English to data-scarce languages like Romanian.

In some cases, shared vocabularies combined with multilingual pretraining are used to achieve transfer to lower resource languages. For example, Artetxe and Schwenk (2019) use a BiLSTM encoder-decoder framework with a shared BPE vocabulary for 93 languages. The framework is pre-trained using parallel corpora, including as Europarl and Tanzil.

The contextual embeddings from the encoder are used to train classifiers using English corpora for downstream tasks.

As the embedding space and the encoder are shared, the resultant classifiers can be transferred to any of the 93 languages without further modification.

Experiments show that these classifiers achieve competitive performance on cross-lingual natural language inference, cross-lingual document classification, and parallel corpus mining.

In another examples, Rosita (Mulcaire et al., 2019) pre-trains a language model using text from different languages, showing the benefits of polyglot learning on low-resource languages.

Also, recently, the authors of BERT developed a multi-lingual BERT which is pre-trained using the Wikipedia dump with more than 100 languages.

XLM (Lample and Conneau, 2019) uses three pre-training methods for learning cross-lingual language models: (1) Causal language modelling, where the model is trained to predict  $p(\text{token}_t | \text{tokens}_{1:t-1})$ , (2) Masked language modelling, and (3) Translation language modelling (TLM).

Parallel corpora are used, and tokens in both source and target sequences are masked for learning cross-lingual association.

XLM performs strongly on cross-lingual classification, unsupervised machine translation, and supervised machine translation.

XLM-R (Conneau et al., 2019) scales up XLM by training a Transformer-based masked language model on one hundred languages, using more than two terabytes of filtered CommonCrawl data. XLM-R shows that large-scale multi-lingual pre-training leads to significant performance gains for a wide range of cross-lingual transfer tasks.

# Given extracted content:

Paper Title: A Primer on Contrastive Pretraining in Language Processing: Methods, Lessons Learned and Perspectives

Section Title: Noise Contrastive Estimation (NCE)

Section Content:

Noise contrastive estimation is the objective used by most contrastive learning approaches within NLP. Thus, we briefly outline its main variants and the core ideas behind them, while pointing to (Ma and Collins, 2018) for detailed, yet readily understandable explanations of the two main NCE variants. Both variants can intuitively be understood as a sub-sampled softmax with K negative samples  $a-i$  and one positive sample  $a+i$ . The first variant expresses NCE as a binary objective (loss) in the form of maximum log likelihood, where only K negatives are considered.

$$LB(\theta, \gamma) = \log \sigma(s(x_i, a+i, 0; \theta), \gamma) + K \sum_{k=1} \log(1 - \sigma(s(x_i, a-i, k; \theta), \gamma))$$

Here,  $s(x_i, a_i, o; \theta)$  is a scoring or similarity function that measures the compatibility between a single text input  $x_i$  and another sample  $a_i, o$ . As mentioned above, the sample can be another input text or an output label (text), thus modeling NLP tasks as 'text-to-text' prediction similar to language models. The similarity function is typically a cosine similarity, a dot product or a logit (unscaled activation) produced by an input-sample matcher sub-network (Rethmeier and Augenstein, 2020). The  $\sigma(z, \gamma)$  is a scaling function, which for use in eq. (1) is typically the sigmoid  $\sigma(z) = \exp(z - \gamma) / (1 + \exp(z - \gamma))$  with a hyperparameter  $\gamma \geq 0$  (temperature), that is tuned or omitted depending on the way that negative samples  $a-i$  are attained.

The other NCE objective learns to rank a single positive pair  $(x_i, a+i, 0)$  over K negative pairs  $(x_i, a-i, k)$ :

$$LR(\theta) = \log \frac{\exp(s(x_i, a+i, 0; \theta))}{\exp(s(x_i, a+i, 0; \theta)) + \sum_{k=1}^K \exp(s(x_i, a-i, k; \theta))}$$

Here, to improve LR or LB performance, (Ma and Collins, 2018) propose a regularized scoring function  $\tilde{s}(x_i, a_i, o) = s(x_i, a_i, o) - \log p_N(a_i, o)$  that subtracts the probability of the current sample  $a_i, o$  under a chosen noise distribution  $p_N(a_i, o)$ . In practice, the noise distribution can be set to 0 (Mnih and Teh, 2012; Wu et al., 2020; Rethmeier and Augenstein, 2020) to save on computation. To robustly learn word embeddings,  $p_N(a_i, o)$  can be set as the word probability  $p_{word}$  in a corpus (Mikolov et al., 2013b), or as the probability of a sequence under a language model  $p_{LM}$  (Deng et al., 2020), when learning contrastive sequence prediction.

Generalization to an arbitrary number of positives: As (Khosla et al., 2020) mention, original contrastive formulations use only one positive pair per text instance (see e.g. (Mikolov et al., 2013b; Logeswaran and Lee, 2018)), while more recent methods mine multiple positives or use multiple gold class annotation representations for contrastive learning (Rethmeier and Augenstein, 2020; Qu et al., 2021). This means that e.g. the positive term in eq. (1) becomes  $\sum_{p=1}^P \log \sigma(s(x_i, a+i, p; \theta), \gamma)$  to consider P positives.

Importance of negative sampling semantics and lessons learned: How positive and negative samples are generated or sampled is a key component of effective contrastive learning. (Saunshi et al., 2019) prove and empirically validate that "sampling more negatives improves performance, but only if they are sampled from the same context or block of information such as the same paragraph". Such hard to contrast (classify) negatives, are sampled in most works (Mikolov et al., 2013b; Saunshi et al., 2019; Rethmeier and Augenstein, 2020; Iter et al., 2020). Otherwise, performance can deteriorate due to weak contrast learning of conceptually related classes. Additionally, (Rethmeier and Augenstein, 2020) find that both positive and negative contrastive samples from a long-tail distribution are essential in predicting rare classes and

in substantially boosting zero-shot performance, especially over minority classes. (Mikolov et al., 2013b) undersample negatives of frequent words to stabilize pretraining of word embeddings to a similar effect. Additional practical advice for negative sampling is mentioned in 3.1.

Question: In contrastive models for nlp, how do the choices of negatives samples influence the quality of representations?

Answer:

Noise contrastive estimation is the objective used by most contrastive learning approaches within NLP.

How positive and negative samples are generated or sampled is a key component of effective contrastive learning.

(Saunshi et al., 2019) prove and empirically validate that “sampling more negatives improves performance, but only if they are sampled from the same context or block of information such as the same paragraph”.

Such hard to contrast (classify) negatives, are sampled in most works (Mikolov et al., 2013b; Saunshi et al., 2019; Rethmeier and Augenstein, 2020; Iter et al., 2020).

Otherwise, performance can deteriorate due to weak contrast learning of conceptually related classes.

Additionally, (Rethmeier and Augenstein, 2020) find that both positive and negative contrastive samples from a long-tail distribution are essential in predicting rare classes and in substantially boosting zero-shot performance, especially over minority classes.

(Mikolov et al., 2013b) undersample negatives of frequent words to stabilize pretraining of word embeddings to a similar effect.

# Given extracted content:

Paper Title: Modeling Language Variation and Universals: A Survey on Typological Linguistics for Natural Language Processing  
Section Title: Hand-Crafted Documentation in Typological Databases

Section Content:

Typological databases are created manually by linguists. They contain taxonomies of typological features, their possible values, as well as the documentation of feature values for the world’s languages. Major typological databases, listed in Table 1, typically organize linguistic information in terms of universal features and language-specific values. For example, Figure 3 presents language-specific values for the feature number of grammatical genders for nouns on a world map. Note that each language is color-coded according to its value. Further examples for each database can be found in the rightmost column of Table 1.

Some databases store information pertaining to multiple levels of linguistic description. These include WALS (Dryer and Haspelmath 2013) and the Atlas of Pidgin and Creole Language Structures (APiCS) (Michaelis et al. 2013). Among all presently available databases, WALS has been the most widely used in NLP. In this resource, which has 142 typological features in total, features 1–19 deal with phonology, 20–29 with morphology, 30–57 with nominal categories, 58–64 with nominal syntax, 65–80 with verbal categories, 81–97 and 143–144 with word order, 98–121 with simple clauses, 122–128 with complex sentences, 129–138 with the lexicon, and 139–142 with other properties.

Other databases only cover features related to a specific level of linguistic description. For example, both Syntactic Structures of the World’s Languages (SSWL) (Collins and Kayne 2009) and AUTOTYP (Bickel et al. 2017) focus on syntax. SSWL features are manually crafted, whereas AUTOTYP features are derived automatically from primary linguistic data using scripts. The Valency Patterns Leipzig (ValPaL) (Hartmann, Haspelmath, and Taylor 2013) provides verbs as attributes and predicate–argument structures as their values (including both valency and morphosyntactic constraints). For example, in both Mandinka and Sliammon, the verb to laugh has a valency of 1; in other words, it requires only one mandatory argument, the subject. In Mandinka the subject precedes the verb, but there is no agreement requirement; in Sliammon, on the other hand, the word order does not matter, but the verb is required to morphologically agree with the subject.

For phonology, the Phonetics Information Base and Lexicon (PHOIBLE) (Moran, McCloy, and Wright 2014) collates

information on segments (binary phonetic features). In the Lyon–Albuquerque Phonological Systems Database (LAPSYD) (Maddieson et al. 2013), attributes are articulatory traits, syllabic structures, or tonal systems. Finally, StressTyp2 (Goedemans, Heinz, and der Hulst 2014) deals with stress and accent patterns. For instance, in Koromfé each word’s first syllable has to be stressed, but not in Cubeo.

Other databases document various aspects of semantics. The World Loanword Database (WOLD) (Haspelmath and Tadmor 2009) documents loanwords by identifying the donor languages and the source words. The Automated Similarity Judgment Program (ASJP) (Wichmann, Holman, and Brown 2016) and the Intercontinental Dictionary Series (IDS) (Key and Comrie 2015) indicate how a meaning is lexicalized across languages: For example, the concept of WORLD is expressed as *mir* in Russian, and as *ārkišos* in Tocharian A.

Although typological databases store abundant information on many languages, they suffer from shortcomings that limit their usefulness. Perhaps the most significant shortcoming of such resources is their limited coverage. In fact, feature values are missing for most languages in most databases. Other shortcomings are related to feature granularity. In particular, most databases fail to account for feature value variation within each language: They report only majority value rather than the full range of possible values and their corresponding frequencies. For example, the dominant adjective–noun word order in Italian is adjective before noun; however, the opposite order is also attested. The latter information is often missing from typological databases.

Further challenges are posed by restricted feature applicability and feature hierarchies. Firstly, some features apply, by definition, only to subsets of languages that share another feature value. For instance, WALS feature 113A documents “Symmetric and Asymmetric Standard Negation,” whereas WALS feature 114A “Subtypes of Asymmetric Standard Negation.” Although a special NA value is assigned for symmetric-negation languages in the latter, there are cases where languages without the prerequisite feature are simply omitted from the sample. Secondly, features can be partially redundant, and subsume other features. For instance, WALS feature 81A “Order of Subject, Object and Verb” encodes the same information as WALS feature 82A “Order of Subject and Verb” and 83A “Order of Object and Verb,” with the addition of the order of subject and object.

Question: What are the differences between publicly available linguistic typology databases?

Answer:

Some typological databases store information pertaining to multiple levels of linguistic description.

These include WALS (Dryer and Haspelmath 2013) and the Atlas of Pidgin and Creole Language Structures (APiCS) (Michaelis et al. 2013).

Among all presently available databases, WALS has been the most widely used in NLP. Other databases only cover features related to a specific level of linguistic description.

For example, both Syntactic Structures of the World’s Languages (SSWL) (Collins and Kayne 2009) and AUTOTYP (Bickel et al. 2017) focus on syntax.

SSWL features are manually crafted, whereas AUTOTYP features are derived automatically from primary linguistic data using scripts. The Valency Patterns Leipzig (ValPaL) (Hartmann, Haspelmath, and Taylor 2013) provides verbs as attributes and predicate–argument structures as their values (including both valency and morphosyntactic constraints).

For phonology, the Phonetics Information Base and Lexicon (PHOIBLE) (Moran, McCloy, and Wright 2014) collates information on segments (binary phonetic features).

In the Lyon–Albuquerque Phonological Systems Database (LAPSYD) (Maddieson et al. 2013), attributes are articulatory traits, syllabic structures, or tonal systems.

Finally, StressTyp2 (Goedemans, Heinz, and der Hulst 2014) deals with stress and accent patterns.

Other databases document various aspects of semantics. The World Loanword Database (WOLD) (Haspelmath and Tadmor 2009)



documents loanwords by identifying the donor languages and the source words.

The Automated Similarity Judgment Program (ASJP) (Wichmann, Holman, and Brown 2016) and the Intercontinental Dictionary Series (IDS) (Key and Comrie 2015) indicate how a meaning is lexicalized across languages.

# Given extracted content:

Paper Title: Efficient Methods for Natural Language Processing: A Survey

Section Title: Sparse Modeling

Section Content:

To leverage sparsity for efficiency, many models follow the mixture-of-experts (MoE) concept (Jacobs et al., 1991; Shazeer et al., 2017; Fedus et al., 2022a), which routes computation through small subnetworks instead of passing the input through the entire model. Relevant works on this line include GShard (Lepikhin et al., 2021), Switch Transformer (Fedus et al., 2022b), and ST-MoE (Zoph et al., 2022), which replace the feed forward layers in transformers with MoE layers. More recently, Rajbhandari et al. (2022) scaled transformers up by compressing and optimizing the usage of MoE. Overall, MoE models have been shown to achieve strong performance across several NLP tasks while reducing the overall resource consumption (Sec. 8). For instance, GLaM (Du et al., 2022) used only  $\sim 1/3$  of GPT-3's energy consumption (with additional hardware-based optimization), while Rajbhandari et al. (2022) reached a 5x reduction in terms of training cost. However, MoE models have also exhibited training instabilities in practice, and may require architecture-specific implementation (Zoph et al., 2022; Mustafa et al., 2022).

Another promising direction for exploiting sparse modeling is Sparsefinder (Treviso et al., 2022), which extends the Adaptively Sparse Transformer (Correia et al., 2019) to allow a more efficient attention mechanism by identifying beforehand the sparsity pattern returned by entmax attention—a sparse alternative to (dense) softmax attention (Peters et al., 2019). Finally, sparsity can also be induced via modularity, e.g., by encapsulating task-specific parameters (Ponti et al., 2022).

Question: How can we utilize sparsity to enhance efficiency in designing NLP models?

Answer:

To leverage sparsity for efficiency, many models follow the mixture-of-experts (MoE) concept (Jacobs et al., 1991; Shazeer et al., 2017; Fedus et al., 2022a), which routes computation through small subnetworks instead of passing the input through the entire model. Relevant works on this line include GShard (Lepikhin et al., 2021), Switch Transformer (Fedus et al., 2022b), and ST-MoE (Zoph et al., 2022), which replace the feed forward layers in transformers with MoE layers.

More recently, Rajbhandari et al. (2022) scaled transformers up by compressing and optimizing the usage of MoE.

Overall, MoE models have been shown to achieve strong performance across several NLP tasks while reducing the overall resource consumption.

For instance, GLaM (Du et al., 2022) used only  $\sim 1/3$  of GPT-3's energy consumption (with additional hardware-based optimization), while Rajbhandari et al. (2022) reached a 5x reduction in terms of training cost.

However, MoE models have also exhibited training instabilities in practice, and may require architecture-specific implementation (Zoph et al., 2022; Mustafa et al., 2022).

Another promising direction for exploiting sparse modeling is Sparsefinder (Treviso et al., 2022), which extends the Adaptively Sparse Transformer (Correia et al., 2019) to allow a more efficient attention mechanism by identifying beforehand the sparsity pattern returned by entmax attention—a sparse alternative to (dense) softmax attention (Peters et al., 2019).

Finally, sparsity can also be induced via modularity, e.g., by encapsulating task-specific parameters (Ponti et al., 2022).

# Given extracted content:

Paper Title: Neural Approaches to Conversational AI

Section Title: Speaker Consistency

Section Content:

It has been shown that the popular seq2seq approach often produces conversations that are incoherent (Li et al., 2016b), where the system may for instance contradict what it had just said in the previous turn (or sometimes even in the same turn). While some of this effect can be attributed to the limitation of the learning algorithms, Li et al. (2016b) suggested that the main cause of this inconsistency is probably due to the training data itself. Indeed, conversational datasets (see Sec. 5.5) feature multiple speakers, which often have different or conflicting personas and backgrounds. For example, to the question “how old are you?”, a seq2seq model may give valid responses such as “23”, “27”, or “40”, all of which are represented in the training data.

This sets apart the response generation task from more traditional NLP tasks: While models for other tasks such as machine translation are trained on data that is mostly one-to-one semantically, conversational data is often one-to-many or many-to-many as the above example implies.<sup>5</sup> As one-to-many training instances are akin to noise to any learning algorithm, one needs more expressive models that exploits a richer input to better account for such diverse responses.

To do this, Li et al. (2016b) proposed a persona-based response generation system, which is an extension of the LSTM model of Sec. 5.1.1 that uses speaker embeddings in addition to word embeddings. Intuitively, these two types of embeddings work similarly: while word embeddings form a latent space in which spacial proximity (i.e., low Euclidean distance) means two words are semantically or functionally close, speaker embeddings also constitute a latent space in which two nearby speakers tend to converse in the same way, e.g., having similar speaking styles (e.g., British English) or often talking about the same topic (e.g., sports). Like word embeddings, speaker embedding parameters are learned jointly with all other parameters of the model from their one-hot representations. At inference time, one just needs to specify the one-hot encoding of the desired speaker to produce a response that reflects her speaking style. The global architecture of the model is displayed in Fig. 5.2, which shows that each target hidden state is conditioned not only on the previous hidden state and the current word embedding (e.g., “England”), but also on the speaker embedding (e.g., of “Rob”). This model not only helps generate more personalized responses, but also alleviates the one-to-many modeling problem mentioned earlier.

Other approaches also utilized personalized information. For example, Al-Rfou et al. (2016) presented a persona-based response generation model, but geared for retrieval using an extremely large dataset consisting of 2.1 billion responses. Their retrieval model is implemented as a binary classifier (i.e., good response or not) using a deep neural network. The distinctive feature of their model is a multi-loss objective, which augments a single-loss model  $p(R|I, A, C)$  of the response  $R$ , input  $I$ , speaker (“author”)  $A$ , and context  $C$ , by adding auxiliary losses that, e.g., model the probability of the response given the author  $p(R|A)$ . This multi-loss model was shown to be quite helpful (Al-Rfou et al., 2016), as the multiple losses help cope with the fact that certain traits of the author are often correlated with the context or input, which makes it difficult to learn good speaker embedding representation. By adding a loss for  $p(R|A)$ , the model is able to learn a more distinctive speaker embedding representation for the author.

More recently, Luan et al. (2017) presented an extension of the speaker embedding model of Li et al. (2016b), which combines a seq2seq model trained on conversational datasets with an autoencoder trained on non-conversational data, where the seq2seq and autoencoder are combined in a multi-task learning setup (Caruana, 1998). The tying of the decoder parameters of both seq2seq and autoencoder enables Luan et al. (2017) to train a response generation system for a given persona without actually requiring any conversational data available for that persona. This is an advantage of their approach, as conversational data for a given user or persona might not always be available. In (Bhatia et al., 2017), the idea of (Li et al., 2016b) is extended to a social-graph embedding model. While (Serban et al., 2017) is not a persona-based response



generation model per se, their work shares some similarities with speaker embedding models such as (Li et al., 2016b). Indeed, both Li et al. (2016b) and Serban et al. (2017) introduced a continuous high-dimensional variable in the target side of the model in order to bias the response towards information encoded in a vector. In the case of (Serban et al., 2017), that variable is latent, and is trained by maximizing a variational lower-bound on the log-likelihood. In the case of (Li et al., 2016b), the variable (i.e., the speaker embedding) is technically also latent, although it is a direct function of the one-hot representation of speaker. (Li et al., 2016b) might be a good fit when utterance-level information (e.g., speaker ID or topic) is available. On the other hand, the strength of (Serban et al., 2017) is that it learns a latent variable that best “explains” the data, and may learn a representation that is more optimal than the one based strictly on speaker or topic information.

Question: Why do conversation models often produce responses that are inconsistent with previous turns?

Answer:

It has been shown that the popular seq2seq approach often produces conversations that are incoherent (Li et al., 2016b), where the system may for instance contradict what it had just said in the previous turn (or sometimes even in the same turn).

While some of this effect can be attributed to the limitation of the learning algorithms, Li et al. (2016b) suggested that the main cause of this inconsistency is probably due to the training data itself.

Conversational datasets feature multiple speakers, which often have different or conflicting personas and backgrounds.

For example, to the question “how old are you?”, a seq2seq model may give valid responses such as “23”, “27”, or “40”, all of which are represented in the training data.

This sets apart the response generation task from more traditional NLP tasks: While models for other tasks such as machine translation are trained on data that is mostly one-to-one semantically, conversational data is often one-to-many or many-to-many as the above example implies.

As one-to-many training instances are akin to noise to any learning algorithm, one needs more expressive models that exploits a richer input to better account for such diverse responses.

# Given extracted content:

Paper Title: <PAPER\_TITLE>

Section Title: <SECTION\_TITLE>

Section Content:

<SECTION\_CONTENT>

Question: <INITIAL\_QUERY>

Answer:

## B.8 Query generation prompt: rephrase query and reference answer for cohesion

**System:** Original question and answer:

Question: <INITIAL\_QUERY>

Answer:

<INITIAL\_REF\_ANSWER>

—  
Rewrite the question and answer to make them more coherent so that it sounds like an answer that an human expert would answer when being asked this question.

Try to use the contents of the answer as much as possible, and keep the citations.

Focus on getting a better organization and transition of the contents, without adding new sentences that are not in the answer.

Output in a format of “Question:\n\nquestion\n\nAnswer:\n\nanswer” is required.

Rephrased question and answer:

## B.9 Parametric rubric generation prompt: information-based item

**System:** You will be shown a query asked by a junior PhD researcher and a referral response to this query.

Imagine that you are required to write another response to this open-ended query,

you need to ask some follow-up questions to know the preferred content in the response or resolve ambiguity in the query.

Here are criteria that individual questions need to satisfy:

- relevant: The question should ask about information that is related to the query, and would lower the answer variability of the query’s response.

- salient: The question should address important piece to guide the writing of query’s response in a way that an experienced research would like to.

- binary: All questions should be yes-no binary questions.

- qualitative: The question should be qualitative and focus on the big picture and important aspects, but not the nonessentials or specific numbers.

Here are the criteria that the list of questions needs to satisfy:

- knowledge-cutoff: The question should not ask for or rely on any information that would violate the specified knowledge cutoff date

- sufficient: There should be enough important questions to cover a large space of possible contexts for the query.

- coverage: The questions in combinations should cover every potential important aspects.

Specifically, here, you are required to generate the question of information type:

For instance, Does the response include key findings A and B? ... Example E?

It is to ask whether an important information exist in the response, such as a statement, a finding, an opinion, a comparison, etc..

It is not to ask a depth question (how to demonstrate an important point in deep), nor a citation question (whether a citation exists).

You should first think about what information is necessary to be included in a good response to the query, and then ask the corresponding questions.

Generate up to 10 questions (but no need to) and they should all meet the above criteria.

You should generate questions that are important and useful, and address a \*necessary\* information perspective of the query’s response.

Please make sure that your questions are relevant, salient, binary, qualitative, and grounded.

Please list the questions from the most to the least necessary to be in the query’s response.

Query: What are the latest works on finetuning an auto-regressive LM for dense passage retrieval and how are their performance compared with bi-directional encoders?

Date: 2024-11-21

Questions:

1. Does the response to include examples of state-of-the-art auto-regressive LMs that outperform bi-directional encoders in retrieval tasks?

2. Does the response highlight hybrid approaches that combine auto-regressive and bi-directional features?

3. Does the response emphasize the specific benchmarks (e.g., MTEB) where auto-regressive LMs outperform bi-directional encoders?

4. Does the response mention the importance to achieve full potential of the decoder models by appropriate optimization?

Query: <QUERY>

Date: <DATE\_CUTOFF>

Questions:

## B.10 Parametric rubric generation prompt: depth-based item

**System:** You will be shown a query asked by a junior PhD researcher and a referral response to this query. Imagine that you are required to write another response to this open-ended query, you need to ask some follow-up questions to know the preferred content in the response or resolve ambiguity in the query.

Here are criteria that individual questions need to satisfy:

- relevant: The question should ask about information that is related to the query, and would lower the answer variability of the query's response.
- salient: The question should address important pieces to guide the writing of the query's response in a way that an experienced researcher would like to.
- binary: All questions should be yes-no binary questions.
- general: The question should not mention a specific paper's name (e.g. Author and year), nor narrow examples or demonstrations.

Here are the criteria that the list of questions needs to satisfy:

- knowledge-cutoff: The question should not ask for or rely on any information that would violate the specified knowledge cutoff date
- sufficient: There should be enough important questions to cover a large space of possible contexts for the query.
- coverage: The questions in combinations should cover every potential important aspect.

Specifically, here, you are required to generate the question of depth type (e.g., Does the response explore area C1 and C2 in detail so that to better support the claim C?). It is to ask whether information exists in the response that is explored in depth, such as detailed extension or explanation. Generally it means the response takes a longer-than-average length to explore a topic. It is not to ask an information question (merely whether a statement, example, finding, opinion etc. exists), instead it requires depth in the contents. You should first find the depth information that was actually included in the given referral response, and then ask the corresponding questions.

Generate up to 5 questions (but no need to) and they should all meet the above criteria. You should generate questions that are important and useful.

Query: What are the latest works on fine-tuning an auto-regressive LM for dense passage retrieval and how are their performance compared with bi-directional encoders?  
Date: 2024-11-21  
Questions:  
1. Does the response compare auto-regressive model and bi-directional model on the dense passage retrieval in detail?  
2. Does the response elaborate on the hybrid approach of GRIT in unifying auto-regressive and bi-directional features?

Query: <QUERY>  
Date: <DATE\_CUTOFF>  
Questions:

## B.11 Parametric rubric generation prompt: citation-based item

**System:** You will be shown a query asked by a junior PhD researcher and a referral response to this query. Imagine that you are required to write another response to this open-ended query, you need to ask some follow-up questions to know the preferred

content in the response or resolve ambiguity in the query.

Here are criteria that individual questions need to satisfy:

- relevant: The question should ask about information that is related to the query, and would lower the answer variability of the query's response.
- salient: The question should address important piece to guide the writing of query's response in a way that an experienced research would like to.
- binary: All questions should be yes-no binary questions.
- qualitative: The question should be qualitative and focus on the big picture and important aspects, but not the nonessentials or specific numbers.

Here are the criteria that the list of questions needs to satisfy:

- knowledge-cutoff: The question should not ask for or rely on any information that would violate the specified knowledge cutoff date
- sufficient: There should be enough important questions to cover a large space of possible contexts for the query.
- coverage: The questions in combinations should cover every potential important aspects.

Specifically, here, you are required to generate the question of citation type. For instance, Does the response cite the paper by author (year) (title: title) that back the statement A by explaining the detail A1 and A2? It is to ask whether an important citation exist in the response, and how it is used in the response. It must mention the title of the paper in a way "(title: title)", and the details of the statement that the citation is used to back up. You should first think of the important citations that are necessary for the query's response, and then generate the questions that ask about them.

Generate up to 5 questions (but no need to) and they should all meet the above criteria. You should generate questions that are important and useful, and address a \*necessary\* citation perspective of the query's response. Please make sure that your questions are relevant, salient, binary, and qualitative. Please list the questions from the most to the least necessary to be in the query's response.

Query: What are the latest works on finetuning an auto-regressive LM for dense passage retrieval and how are their performance compared with bi-directional encoders?  
Date: 2024-11-21  
Questions:  
1. Does the response cite some 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?  
2. Does the response cite the GRIT model paper (title: Generative Representational Instruction Tuning) that shows how to leverage the strengths of both auto-regressive and bi-directional model by incorporating two distinct language modeling heads atop the Transformer layers?  
  
Query: <QUERY>  
Date: <DATE\_CUTOFF>  
Questions:

## B.12 Survey rubric generation prompt: information-based item

**System:** You will be shown a query asked by a junior PhD researcher and a referral response to this query. Imagine that you are required to write another response to this open-ended query,

you need to ask some follow-up questions to know the preferred content in the response or resolve ambiguity in the query. The referral response is written by experienced research, and is perfect for generating these follow-up questions to the query.

Here are criteria that individual questions need to satisfy:

- relevant: The question should ask about information that is related to the query, and would lower the answer variability of the query's response.
- salient: The question should address important pieces to guide the writing of the query's response in a way that an experienced researcher would like to.
- binary: All questions should be yes-no binary questions.
- general: The question should not mention a specific paper's name (e.g. Author and year), nor narrow examples or demonstrations.
- grounded: These questions should be strictly based on the referral response.

Here are the criteria that the list of questions needs to satisfy:

- sufficient: There should be enough important questions to cover a large space of possible contexts for the query.
- coverage: The questions in combinations should cover every important aspect in the referral response.

Specifically, here, you are required to generate the question of information type (e.g., Does the response include key findings A and B? ... Example E?).

It is to ask whether important information exists in the response, such as a statement, a finding, an example, an opinion, a comparison, etc..

It is not to ask a depth question (whether a detailed extension or explanation exists), nor a citation question (whether a citation exists).

You should first find the information that was actually included in the given referral response, and then ask the corresponding questions.

Generate up to 10 questions (but no need to) and they should all meet the above criteria.

You should generate questions that are important and useful.

Query-Response Pair:

Query:

What are the latest works on finetuning an auto-regressive LM for dense passage retrieval and how are their performance compared with bi-directional encoders?

Response:

It was traditionally assumed that decoder models would underperform compared to bi-directional autoregressive models, as referenced in [9]. However, recent advancements have challenged this notion, demonstrating that finetuned auto-regressive language models (LMs) can indeed surpass the capabilities of bi-directional encoder models in retrieval embedding tasks [1][2][3][4].

One notable example is the E5-Mixtral model [1], which employs a novel approach by generating a diverse set of synthetic data to finetune Mistral, a pre-trained large language model (LLM). This model has shown superior performance over the state-of-the-art (SOTA) bi-directional embedding models of that time, including OpenAI's text-embedding-3-large, Cohere-embed-english-v3.0, and voyage-lite-01-instruct on the MTEB benchmark [8].

Another study LLM2Vec [2] demonstrates the adaptation of a decoder model solely using public data for embedding tasks, which also achieved remarkable results on the MTEB benchmark [8], surpassing other encoder models by a significant margin. This highlights the potential of decoder models when optimized appropriately.

Furthermore, the GRIT model [3] unifies auto-regressive and bi-directional model designs. It incorporates two distinct language modeling heads atop the Transformer layers: one auto-regressive head, designed for generative tasks with a causal attention mask, and another bi-directional head, tailored for embedding tasks. This dual-head approach allows GRIT to leverage the strengths of both modeling techniques.

Lastly, the NV-Embed model [4] adapts the decoder model architecture to enhance finetuning effectiveness using public datasets. This model not only ranks at the top on the MTEB benchmark [8] but also outperforms all existing baselines on the information retrieval benchmark [5], underscoring the robustness and versatility of decoder models in handling complex language tasks.

In short, E5-Mixtral [1], LLM2Vec [2], GRIT [3], and NV-Embed [4] are good representations of the SOTA auto-regressive LMs on retrieval tasks whose performance is superior than traditional bi-directional encoders [5][6][7][8].

[1] Improving text embeddings with large language models. (2023)

[2] LLM2Vec: Large Language Models Are Secretly Powerful Text Encoders (2024)

[3] Generative Representational Instruction Tuning (2024)

[4] NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models (2024)

[5] NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models (2024)

[6] Improving text embeddings with large language models. (2023)

[7] Generative Representational Instruction Tuning (2024)

[8] MTEB: Massive Text Embedding Benchmark (2022)

[9] NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models (2024)

Questions:

1. Does the response include examples of state-of-the-art auto-regressive LMs such as E5-Mixtral, LLM2Vec, GRIT, and NV-Embed?
2. Does the response highlight hybrid approaches like GRIT that combine auto-regressive and bi-directional features?
3. Does the response discuss details about the novel techniques used in these models, such as synthetic data generation?
4. Does the response emphasize the specific benchmarks (e.g., MTEB) where auto-regressive LMs outperform bi-directional encoders?
5. Does the response address the role of public datasets in training models like LLM2Vec and NV-Embed?
6. Does the response mention the importance to achieve full potential of the decoder models by appropriate optimization?

Query-Response Pair:

Query:

<QUERY>

Response:

<REF\_ANSWER>

Questions:

## B.13 Survey rubric generation prompt: depth-based item

**System:** You will be shown a query asked by a junior PhD researcher and a referral response to this query.

Imagine that you are required to write another response to this open-ended query,

you need to ask some follow-up questions to know the preferred content in the response or resolve ambiguity in the query.

The referral response is written by experienced research, and is perfect for generating these follow-up questions to the query.

Here are criteria that individual questions need to satisfy:

- relevant: The question should ask about information that is related

to the query, and would lower the answer variability of the query's response.

- salient: The question should address important pieces to guide the writing of the query's response in a way that an experienced researcher would like to.
- binary: All questions should be yes-no binary questions.
- general: The question should not mention a specific paper's name (e.g. Author and year), nor narrow examples or demonstrations.
- grounded: These questions should be strictly based on the referral response.

Here are the criteria that the list of questions needs to satisfy:

- sufficient: There should be enough important questions to cover a large space of possible contexts for the query.
- coverage: The questions in combinations should cover every important aspect in the referral response.

Specifically, here, you are required to generate the question of depth type (e.g., Does the response explore area C1 and C2 in detail so that to better support the claim C?).

It is to ask whether information exists in the response that is explored in depth, such as detailed extension or explanation. Generally it means the response takes a longer-than-average length to explore a topic.

It is not to ask an information question (merely whether a statement, example, finding, opinion etc. exists), instead it requires depth in the contents.

You should first find the depth information that was actually included in the given referral response, and then ask the corresponding questions.

Generate up to 5 questions (but no need to) and they should all meet the above criteria.

You should generate questions that are important and useful.

Query:

What are the latest works on finetuning an auto-regressive LM for dense passage retrieval and how are their performance compared with bi-directional encoders?

Response:

It was traditionally assumed that decoder models would underperform compared to bi-directional autoregressive models, as referenced in [9]. However, recent advancements have challenged this notion, demonstrating that finetuned auto-regressive language models (LMs) can indeed surpass the capabilities of bi-directional encoder models in retrieval embedding tasks [1][2][3][4].

One notable example is the E5-Mixtral model [1], which employs a novel approach by generating a diverse set of synthetic data to finetune Mistral, a pre-trained large language model (LLM). This model has shown superior performance over the state-of-the-art (SOTA) bi-directional embedding models of that time, including OpenAI's text-embedding-3-large, Cohere-embed-english-v3.0, and voyage-lite-01-instruct on the MTEB benchmark [8].

Another study LLM2Vec [2] demonstrates the adaptation of a decoder model solely using public data for embedding tasks, which also achieved remarkable results on the MTEB benchmark [8], surpassing other encoder models by a significant margin. This highlights the potential of decoder models when optimized appropriately.

Furthermore, the GRIT model [3] unifies auto-regressive and bi-directional model designs. It incorporates two distinct language modeling heads atop the Transformer layers: one auto-regressive head, designed for generative tasks with a causal attention mask, and another bi-directional head, tailored for embedding tasks. This dual-head approach allows GRIT to leverage the strengths of both modeling techniques.

Lastly, the NV-Embed model [4] adapts the decoder model architecture to enhance finetuning effectiveness using public datasets. This model not only ranks at the top on the MTEB benchmark [8] but also outperforms all existing baselines on the information retrieval benchmark [5], underscoring the robustness and versatility of decoder models in handling complex language tasks.

In short, E5-Mixtral [1], LLM2Vec [2], GRIT [3], and NV-Embed [4] are good representations of the SOTA auto-regressive LMs on retrieval tasks whose performance is superior than traditional bi-directional encoders [5][6][7][8].

- [1] Improving text embeddings with large language models. (2023)
- [2] LLM2Vec: Large Language Models Are Secretly Powerful Text Encoders (2024)
- [3] Generative Representational Instruction Tuning (2024)
- [4] NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models (2024)
- [5] NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models (2024)
- [6] Improving text embeddings with large language models. (2023)
- [7] Generative Representational Instruction Tuning (2024)
- [8] MTEB: Massive Text Embedding Benchmark (2022)
- [9] NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models (2024)

Questions:

1. Does the response compare auto-regressive model and bi-directional model on the dense passage retrieval in detail?
2. Does the response elaborate on the hybrid approach of GRIT in unifying auto-regressive and bi-directional features?

Query-Response Pair:

Query:

<QUERY>

Response:

<REF ANSWER>

Questions:

## B.14 Survey rubric generation prompt: citation-based-item

**System:** You will be shown a query asked by a junior PhD researcher and a referral response to this query. Imagine that you are required to write another good response to this open-ended query, you need to ask some follow-up questions to know the preferred content in the response or resolve ambiguity in the query. The referral response is written by experienced research, and is perfect for generating these follow-up questions to the query.

Here are criteria that individual questions need to satisfy:

- relevant: The question should ask about information that is related to the query, and would lower the answer variability of the query's response.
- salient: The question should address important piece to guide the writing of query's response in a way that an experienced research would like to.
- binary: All questions should be yes-no binary questions.
- qualitative: The question should be qualitative and focus on the big picture and important aspects, but not the nonessentials or specific numbers.
- grounded: These questions should be strictly based on the referral response.

Here are the criteria that the list of questions needs to satisfy:

- sufficient: There should be enough important questions to cover a large space of possible contexts for the query.
- coverage: The questions in combinations should cover every important aspects in the referral response.



Specifically, here, you are required to generate the question of citation type.

For instance, Does the response cite the paper by author (year) (title: title) that back the statement A by explaining the detail A1 and A2? It is to ask whether an important citation exist in the response, and how it is used in the response.

It must mention the title of the paper in a way “(title: title)” as long as the title is provided, and the details of the statement that the citation is used to back up.

You should first find the citation(s) that were actually included in the given referral response, and then ask the corresponding questions.

Generate up to 5 questions (but no need to) and they should all meet the above criteria.

You should generate questions that are important and useful, and address a \*necessary\* citation perspective of the query’s response.

Please make sure that your questions are relevant, salient, binary, qualitative, and grounded.

Please list the questions from the most to the least necessary to be in the query’s response.

Query-Response Pair:

Query:

What are the latest works on finetuning an auto-regressive LM for dense passage retrieval and how are their performance compared with bi-directional encoders?

Response:

It was traditionally assumed that decoder models would underperform compared to bi-directional autoregressive models, as referenced in [9]. However, recent advancements have challenged this notion, demonstrating that finetuned auto-regressive language models (LMs) can indeed surpass the capabilities of bi-directional encoder models in retrieval embedding tasks [1][2][3][4].

One notable example is the E5-Mixtral model [1], which employs a novel approach by generating a diverse set of synthetic data to finetune Mistral, a pre-trained large language model (LLM). This model has shown superior performance over the state-of-the-art (SOTA) bi-directional embedding models of that time, including OpenAI’s text-embedding-3-large, Cohere-embed-english-v3.0, and voyage-lite-01-instruct on the MTEB benchmark [8].

Another study LLM2Vec [2] demonstrates the adaptation of a decoder model solely using public data for embedding tasks, which also achieved remarkable results on the MTEB benchmark [8], surpassing other encoder models by a significant margin. This highlights the potential of decoder models when optimized appropriately.

Furthermore, the GRIT model [3] unifies auto-regressive and bi-directional model designs. It incorporates two distinct language modeling heads atop the Transformer layers: one auto-regressive head, designed for generative tasks with a causal attention mask, and another bi-directional head, tailored for embedding tasks. This dual-head approach allows GRIT to leverage the strengths of both modeling techniques.

Lastly, the NV-Embed model [4] adapts the decoder model architecture to enhance finetuning effectiveness using public datasets. This model not only ranks at the top on the MTEB benchmark [8] but also outperforms all existing baselines on the information retrieval benchmark [5], underscoring the robustness and versatility of decoder models in handling complex language tasks.

In short, E5-Mixtral [1], LLM2Vec [2], GRIT [3], and NV-Embed [4] are good representations of the SOTA auto-regressive LMs on retrieval tasks whose performance is superior than traditional bi-directional encoders [5][6][7][8].

[1] Improving text embeddings with large language models.

Liang Wang, et al. (2023)

[2] LLM2Vec: Large Language Models Are Secretly Powerful Text Encoders. Parishad BehnamGhader, et al. (2024)

[3] Generative Representational Instruction Tuning. Niklas Muenighoff, et al. (2024)

[4] NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models. Chankyu Lee, et al. (2024)

[5] NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models. Chankyu Lee, et al. (2024)

[6] Improving text embeddings with large language models. Liang Wang, et al. (2023)

[7] Generative Representational Instruction Tuning. Niklas Muenighoff, et al. (2024)

[8] MTEB: Massive Text Embedding Benchmark. Niklas Muenighoff, et al. (2022)

[9] NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models. Chankyu Lee, et al. (2024)

Questions:

1. Does the response cite some 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?

2. Does the response cite the GRIT model paper (title: Generative Representational Instruction Tuning) that shows how to leverage the strengths of both auto-regressive and bi-directional model by incorporating two distinct language modeling heads atop the Transformer layers?

Query-Response Pair:

Query:

<QUERY>

Response:

<REF\_ANSWER>

Questions:

## B.15 Rubric deduplication prompt

**System:** The rubric items below intend to evaluate answers to the following query: <QUERY> (Date Cutoff: <DATE>).

Rubric: <CURRENT\_RUBRIC>

The user wants to add to the rubric, but only if the new rubric item is not a rephrasing or restatement of an existing rubric item. In other words, no two rubric items should not repeat another. Answer only ‘new item’ if the user should add the item to the rubric, and answer only ‘repeated item’ if the item is already represented in the rubric. Do not elaborate.

**User:** Here is the proposed new rubric item:  
<NEW\_RUBRIC\_ITEM>

## B.16 Hybrid rubric reranker prompt

**System:** Select the <RUBRIC\_SIZE> most important questions that should be addressed in order to answer the following query: <QUERY> (Date Cutoff: <DATE>).

Format: List the questions in a 0-indexed python list, e.g., [0, 3, ...].

**User:** Here are the following questions to choose from:  
<SURVEY\_RUBRIC> <PARAMETRIC\_RUBRIC>