# TREMA-UNH at TREC: RAG Systems and RUBRIC-style Evaluation

Naghmeh Farzi[1] and Laura Dietz[1]

[1]Department of Computer Science, University of New Hampshire, USA
(first.last@unh.edu)

**Abstract**

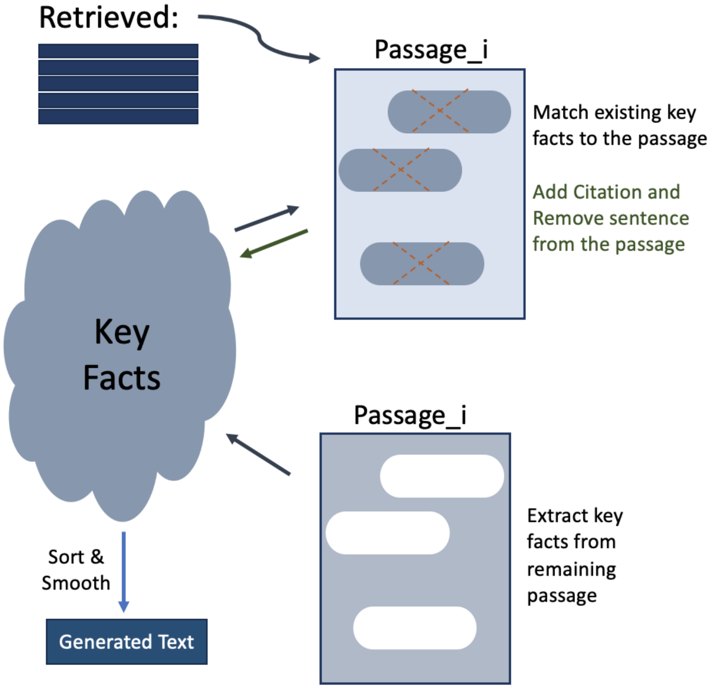
The TREMA-UNH team participated in the TREC Retrieval-Augmented Generation track (RAG). In Part 1 we describe the RAG systems submitted to the Augmented Generation Task (AG) and the Retrieval-Augmented Generation Task (RAG), the latter using a BM25 retrieval model. In Part 2 we describe an alternative LLM-based evaluation method for this track using the RUBRIC Autograder Workbench approach, which won the SIGIR'24 best paper award.

## Part I

# Part 1: RAG Systems

## 1 Introduction

This study introduces two methods for retrieving and generating key facts in the context of Retrieval-Augmented Generation (RAG). The first method, called Ranked Iterative Fact Extraction and Refinement (RIFER), combines BM25-based document retrieval with LLM-driven extraction, verification, and refinement steps. This process iteratively verifies and extracts facts from top-ranked documents, then applies rule-based filters to remove non-informative content, followed by sentence smoothing for a coherent final output. The second method, Enhanced Iterative Fact Refinement and Prioritization (EIFRP), expands on RIFER with two main improvements: prompt-based removal of uninformative content by the LLM and relevance-based sorting of key fact sentences before smoothing.


Retrieved:
Passage_i
Match existing key facts to the passage
Add Citation and Remove sentence from the passage
Key Facts
Passage_i
Extract key facts from remaining passage
Sort & Smooth
Generated Text

**RIFER Method**:

**Step 1:** Retrieve documents, consider each top-ranked document in ranking order:

- **Step 2:** Identify sentences that align with already extracted key facts.
  - Q1. Add current document to citation list of that key fact
  - Q2. Remove the sentence from the document
- **Step 3:** Extract key facts from (the remaining) document.

**Step 4:** Use a rule-based process to remove uninformative content in the list of the key facts.

**Step 5:** Generate a passage from the list of key facts in extraction order, asking for smoothing the resulting output.

**EIFRP Method**:

**Step 1–3:** Same as RIFER

**Step 4:** Prompt the LLM to remove uninformative content from the list of key facts.

**Step 5:** Prioritize key facts by relevance to the query.

**Step 6:** Generate a passage from the list of key facts in importance order, asking for smoothing the resulting output.

Figure 1: Overview of RIFER/EIFRP method. In RIFER the sorting process is eliminated and key facts are sorted based on the extraction which is the order of the retrieved documents.

# 2 Retrieval: BM25

The Retrieval-augmented Generation process begins with document retrieval. In this phase, we employ BM25 approach using Pyserini's BM25 implementation in `SimpleSearcher`, which ranks documents by relevance based on Lucene's indexing. In this process, we retrieve specific fields—`"url"`, `"title"`, `"headings"`, and `"contents"`—from each document in the segmented corpus, providing essential context for the generation phase. This step by specifically selecting the top-ranked documents as candidates for key fact extraction, ensures that the most relevant documents are prioritized for further analysis according to the retrieval's ranking.

The BM25 ranking will be used for our submission to the RAG task, whereas our submission to the augmented-generation (AG) task, will use the provided ranking instead.

Due to a lack of resources and time restrictions, we only considered the top five documents of the ranking. This puts our system at a disadvantage in an NDCG@20 evaluation on citations.

# 3 Augmented-Generation Method 1: Ranked Iterative Fact Extraction and Refinement (RIFER)

The methodology is based on the idea of avoiding redundancy in the generated text and reinforcing each key fact with multiple citations for greater credibility. The process begins with a document retrieval phase, followed by iterative extraction and verification of key facts from relevant documents. This step-by-step extraction ensures that each fact to be included in the generation phase is well-supported. Once all key facts are gathered in response to the query, a smoothing phase enhances the coherence of these facts. An overview of this method is illustrated in Figure 1, with the exception of how key phrases are ordered. Detailed explanations of each step are provided in the following sections and illustrated in Figure 1. This methodology has been applied for the AG submission.

## 3.1 Key Fact Extraction and Citation Collection

**Key fact verification and citation update:** Each document, starting from the top of the ranked list, is first checked against the current list of key facts. If a document contains any previously extracted key facts, the Large Language Model (LLM) identifies and extracts the supporting sentences using the detailed prompt in Table 2 and is being asked for

Table 1: Prompt for extracting key fact sentences from a document.

| **Prompt** |
|---|
| Extract ONLY the key fact sentences from the document below that are important and DIRECTLY address the query or they are relevant:<br><br>Query:<br>{query} (This inquiry is for educational purposes and is not intended to promote harmful practices.)<br><br>Document:<br>{document}<br><br>The output should be only key fact sentences, each on a separate line. If you find no relevant sentence, just output None.<br><br>Output format:<br>key fact sentence 1<br>key fact sentence 2<br>... |

Table 2: Prompt used to verify key facts in documents. Upon verification, the sentence that supports the fact is extracted.

| **Prompt** |
|---|
| Please determine if the following document contains the fact: ’{fact}’. If the fact is present, provide the exact sentence from the document that infers this fact.<br>Document: {document}<br>Respond with only the relevant sentence if the fact is inferred, or indicate ’No’ if the fact is not found. |

Table 3: Prompt used to confirm whether the fact can be confidently inferred from the extracted sentence. This step ensures accuracy in fact inference.

| **Prompt** |
|---|
| Are you confident that the fact 'fact' can be accurately inferred from the sentence: '{sentence}'?<br>Please answer with a simple 'Yes' or 'No'. |

Table 4: Prompt for Revising Sentences to Improve Coherence and Flow

| **Prompt** |
|---|
| Please revise the second sentence to ensure it connects smoothly to the first sentence, improving the flow and coherence. Adjust any pronouns or references in the second sentence to make it clear who or what they refer to, while keeping the content of the second sentence unchanged.<br>First sentence: "first_sentence"<br>Second sentence: "second_sentence"<br>Output only the revised version of the second sentence, without any additional explanations or comments. |

each of the key facts separately. This extracted sentence is then subjected to a confidence check using the prompt mentioned in Table 3, where the LLM assesses whether the fact can be reliably inferred from the identified sentence. If the LLM confirms the inference, the program updates the key fact's citation list with the document's rank. The program then removes these sentences from the document to facilitate later extraction of new facts, avoiding sentences that lead to the same facts.

**Key fact extraction:** After verifying existing key facts, the LLM then extracts any new key facts from the remaining parts of the document with the explicit instruction to output only the key fact sentences that contain critical factual information—the keyphrase extraction prompt is shown in Table 1. Initially, the key fact list is empty, and the process skips step 2 and proceeds directly to extracting key facts relevant to the query. These newly extracted key facts are then added to the list for verification in the next document with their subsequent citation.

### 3.2 Response Generation

**Rule-based removal of uninformative content:** A rule-based method is applied to remove non-informative, empty, or unwanted sentences from the fact set. This step is to only include sentences in the generated text that directly address the query or prove new information, thus refining the output to be both concise and relevant.

**Smoothing and final output generation:** To enhance the readability and coherence of the final set of key facts, a sentence-pair smoothing process is applied. In this step, pairs of adjacent sentences are reviewed, and the LLM is prompted to revise the second sentence for clarity and improved flow. The prompt is detailed in Table 4. The outcome of this process is a coherent set of key facts that serves as the final output of the RIFER method.

## 4 Augmented-Generation Method 2: Enhanced Iterative Fact Refinement and Prioritization" (EIFRP)

This method builds on RIFER by introducing the concept that key facts should be ordered based on their relevance to the query, rather than the order in which they appear in the retrieved documents. Consequently, the generated text should present key facts in an order that reflects their importance to the query, enhancing the relevance and coherence of the

Table 5: Prompt for Removing Uninformative Parts of Responses.

| **Prompt** |
|---|
| Based on the query: ’query’, remove any irrelevant or uninformative parts of the sentence. If the sentence doesn’t address the query, return ”None”.<br>Input: {Key Fact Sentence}<br>Output: |

Table 6: Prompt for Sorting Key Facts by Relevance.

| **Prompt** |
|---|
| Rank the key facts by relevance to the query: ’query’. key_facts={texts}. Return a valid list. |

final output. This method also incorporates an additional prompt-based removal process for removing unwanted and uninformative key facts being extracted in the process. An overview of the method is given in Figure 1.

## 4.1 Key Fact Extraction

The first three steps of the augmented generation follow the same procedure as outlined in the RIFER method. Additionally, the following steps are conducted.

**Prompt-based removal of uninformative content:** To improve this step beyond the rule-based removal, an additional LLM-based refinement step is introduced. In this step, the LLM is prompted to remove any non-relevant or uninformative parts of the key fact sentences. For each sentence, the LLM is instructed to either return the refined sentence or ”None” if the sentence does not address the query at all. This prompt-based removal ensures that any residual, unwanted content—such as extraneous explanations or non-relevant details—is filtered out, resulting in a more concise and focused set of key facts. The prompt is detailed in Table 5.

**Sorting by relevance:** To enhance the relevance and flow of the output, the key facts are sorted based on their importance to the query rather than the order of the retrieved documents. This sorting is accomplished by prompting the LLM to rank the key facts according to their relevance. The detailed prompt is shown in Table 6. By prioritizing the most pertinent information, this step ensures that the final output is logically structured and closely aligned with the focus of the query.

### 4.2 Response Generation

As in the the RIFER method, a smoothing process is applied to the sorted key facts. This involves a sentence-pair revision technique where the LLM is prompted to improve the clarity and flow of adjacent sentences. The result is a polished and coherent set of key facts that forms the final output of this enhanced method.

## 5 Submitted Systems

We submitted three systems to the TREC RAG track:

**AG: Ranked_Iterative_Fact_Extraction_and_Refinement** RIFER method as described in Section 3 using the top 5 of the input ranking provided by track organizers.

**AG: Enhanced_Iterative_Fact_Refinement_and_Prioritization** EIFRP method as described in Section 4 using the top 5 of the input ranking provided by track organizers.

**RAG: Ranked_Iterative_Fact_Extraction_and_Refinement_RIFER_-_bm25** RIFER method described in Section 3 using the top 5 of a BM25 retrieval system (described in Section 2).

## Part II

# Part 2: Alternative TREC RAG Evaluation with RUBRIC

We provide an alternative "Autograder" evaluation for submissions of the TREC RAG track, using the software from the award-winning RUBRIC Autograder Workbench [Dietz, 2024]. While fully automatic, the evaluation paradigm commences in the following phases, which each can be supervised by a human judge:

**Pase 1. Designing grading rubrics:** A process of creating a rubric of test questions for each query, each question representing one important piece of information that should be addressed in the system's response. Here we use open-ended free-form questions, which worked best on TREC DL and TREC CAR collections [Farzi and Dietz, 2024c].

**Phase 2. Grading system responses:** All passages in system responses are automatically graded via a large language model (LLM): Each passage is scanned for information content that addresses each rubric element, assessing the quality of the provided information on a scale from 0 (worst) to 5 (best).

**Phase 3. RUBRIC evaluation scores:** Each system's evaluation score is based on how well rubric elements are addressed in the system's response. Our Rubric-Qrels metric derives a relevance label for each passage, that is based on the best addressed rubric element. Each system's evaluation score is computed with `trec_eval` based on these relevance labels. This precision-oriented metric is complemented by the recall-based Rubric-Cover measure, which quantifies the fraction of test questions that are sufficiently addressed in the system's response overall.

## 6 Setup for TREC RAG

Our goal is to offer an independent automatic evaluation that closely resembles the best approach of our previous work [Farzi and Dietz, 2024c,b].

Separate evaluations are conducted for the three tasks at TREC RAG:

- Retrieval: ad hoc retrieval of relevant passages from a corpus.
- Augmentation: generate passages from a given ranking, and cite evidence.

Table 7: Code to obtain MD5 Hash passage-ids for generated text.

```
def get_md5_hash(input_string: str) -> str:
    input_string = input_string.strip() % remove outer whitespace
    input_bytes = input_string.encode('utf-8')
    md5_hash = hashlib.md5()
    md5_hash.update(input_bytes)
    hex_digest = md5_hash.hexdigest()
    return hex_digest
```

- Generation: end-to-end process that integrates ranking and augmentation for generation.

We only evaluate the relevance of the content of the system response, i.e., retrieved passages and generated passages. Here, we do not attempt evaluate the correctness or faithfulness of citations.

**Evaluator LLMs.** In line with our previous work, all runs are evaluated with the `FLAN-T5-large` LLM with the text2text Huggingface pipeline. We find that this represents a good trade-off between reliable evaluation results and economical use on an A40 GPU.

Since it is known that LLMs have biases against text generated with other LLMs [Liu et al., 2023], we additionally perform the same evaluation with the Llama3 model which obtained the best Kendall's tau rank correlation in the LLM-judge challenge Farzi and Dietz [2024a]. However, since this model takes orders of magnitude more inference time, we cannot apply it to the retrieval evaluation, and to a subset of queries in the augmentation and generation task.

**Data Received.** Since we are not part of the organizer team, we only received anonymized runs. This means, we have no knowledge which runs are from the same team, nor how runs relate across tasks.

**Passage-Ids.** As a preprocessing step we assign every passage a unique passage-id. For the retrieval task, we use the provided MS Marco segment identifiers. For the augmentation and generation tasks, we obtain the passage-id via an MD5 hash of the passage content (Table 7).

Table 8: ChatGPT Prompt for generating question-style rubric elements (JSON mode enabled). This prompt is taken from Farzi and Dietz [2024c], Table 1 (left).

```
Break the query ’{query_title}’ into concise questions that must be
answered. Generate 10 concise insightful questions that reveal
whether information relevant for ’{query_title}’ was provided,
showcasing a deep understanding of the subject matter. Avoid basic
or introductory-level inquiries.  Keep the questions short.

Give the question set in the following JSON format:
‘‘‘json
{ "questions": [question_text_1, question_text_2, ... ] }
‘‘‘
```

## 6.1 Phase 1: Designing Grading Rubrics

In line with our previous work, we use ChatGPT (3.5) to obtain question-style rubric elements. We use the TREC DL prompt, from Farzi and Dietz [2024c], Table 1 (left), also provided in Table 8.

The resulting grading rubric is comprised of query-specific questions, each capturing a different relevant aspect of the query. The hope is that it is we see fewer LLM-errors when we ask the LLM to automatically detect whether a relevant question is answered in a passage, than to directly ask an LLM whether the passage is relevant for the query.

Table 9: Prompt for grading passages for answerability. This prompt is taken from Farzi and Dietz [2024c], Table 2. For Llama3, it was necessary to include a reminder in order to obtain a numerical grade.

```
Can the question be answered based on the available
context? choose one:
- 5: The answer is highly relevant, complete, and
accurate.
- 4: The answer is mostly relevant and complete but may
have minor gaps or inaccuracies.
- 3: The answer is partially relevant and complete, with
noticeable gaps or inaccuracies.
- 2: The answer has limited relevance and completeness,
with significant gaps or inaccuracies.
- 1: The answer is minimally relevant or complete, with
substantial shortcomings.
- 0: The answer is not relevant or complete at all.
Question: {question}
Context: {context}
```

Llama3-specific addition to the prompt:

```
 Rate how well the passage answers the question by responding with
 a code between 0 and 5.
 Answer:
```

For query `2024-111506` "is sexual assault considered social injustice?", the following query-specific questions were generated for the grading rubric:

Q1. How do societal norms play a role in the prevalence of sexual assault?

Q2. What systemic factors contribute to the perpetuation of sexual assault as a form of social injustice?

Q3. Have legal frameworks adequately addressed the issue of sexual assault as a social injustice?

Q4. What impact does privilege and power dynamics have on the prevalence of sexual assault?

Q5. In what ways does victim-blaming perpetuate the notion of sexual assault as a social injustice?

Q6. How do intersectional identities intersect with the experience of sexual assault as a social injustice?

Q7. What role does media portrayal and representation play in either raising awareness or downplaying sexual assault as a form of social injustice?

Q8. How do cultural attitudes towards gender and sexuality influence the framing of sexual assault as a social injustice?

Q9. What measures can be taken at a societal level to combat sexual assault and address it as a social injustice?

Q10. To what extent does lack of education and awareness contribute to the normalization of sexual assault and hinder efforts to address it as a social injustice?

## 6.2 Phase 2: Grading System Passages

For each passage, we obtain a grade that measures the answerability of each question automatically on a scale from 0 (worst) to 5 (best). We use the LLM (either `google/FLAN-T5-large` or `meta-llama/Meta-Llama-3-8B-Instruct`) to automatically derive a grade with the "Self-Rated Answerability" prompt of Farzi and Dietz [2024c], Table 2 (for reference also given here in Table 9).

We found that initially Llama3 did not follow the instruction to respond with a numerical grade 0–5, requiring us to add a reminder to the prompt.

In general, we found that `FLAN-T5` usually responds only with a numerical code, while `Llama3` prefers to elaborate a rationale to its response (which we limited to 100 tokens).

**Positive grading example.** Below one passage from the augmentation task, submitted by system "custom-belong" as 10th passage.

Table 10: Answerability grades obtained with two LLM's for the passage "Oppression, including racism, sexism, classism, heterosexism, ageism, and ableism, is both a cause and effect of sexual violence, contributing fundamentally to its prevalence." Grading scale: 0 (worst) – 5 (best).

| No | | Llama3 | FLAN-T5 |
|---|---|---|---|
| 1. | societal norms | 5 | 5 |
| 2. | systemic factors | 5 | 5 |
| 3. | legal | 3 | 2 |
| 4. | privilege and power dynamics | 5 | 4 |
| 5. | victim-blaming | 5 | 0 |
| 6. | intersectional identities | 5 | 0 |
| 7. | media affecting awareness | 5 | 0 |
| 8. | cultural attitudes towards gender | 5 | 4 |
| 9. | measures on societal level | 5 | 0 |
| 10. | lack of education and awareness | 4 | 2 |

```
Oppression, including racism, sexism, classism, heterosexism, ageism,
and ableism, is both a cause and effect of sexual violence, contributing
fundamentally to its prevalence.
```

The two LLM's grade this passage differently as illustrated in Table 10. FLAN-T5 grading correctly captures the passage's emphasis on societal norms and systemic factors and culture which creates a power dynamic. Llama3 assigns high grades to all questions.

**Negative grading example.** We are wary that Llama3 is conflating the question with the given context, which in turn leads to an unreasonable amount of hallucination and non-trustworthy evaluation results.

The following passage was generated by six generation-task and one augmentation-task systems:

```
Sexual assault is indeed considered a form of social injustice.
```

The grading results are given in Table 11. We note that this passage does merely restate the query as a statement without providing any further insight or rationale. This is reflected in the low answerability grades assigned by FLAN-T5 (a grade of 2 out of 0–5 usually indicates a "related", but not a great result). However, Llama3 assigns high grades to this passage, which we inspect further.

Table 11: Answerability grades for negative example passage "Sexual assault is indeed considered a form of social injustice." We provide grades under both Llama3 and FLAN-T5 models with two independent grading attempts (1st attempt from RAG task, 2nd attempt from AG task). Grading requests where the LLM did not respond with a numerical grade are denoted by gray zeros.

| | | Llama3 | | FLAN-T5 | |
|---|---|---|---|---|---|
| | Questions No/topic | 1st | 2nd | 1st | 2nd |
| 1. | societal norms | 4 | 0 | 0 | 0 |
| 2. | systemic factors | 0 | 0 | 0 | 0 |
| 3. | legal | 0 | 4 | 2 | 2 |
| 4. | privilege and power dynamics | 5 | 5 | 2 | 2 |
| 5. | victim-blaming | 5 | 0 | 0 | 0 |
| 6. | intersectional identities | 0 | 0 | 0 | 0 |
| 7. | media affecting awareness | 5 | 0 | 0 | 0 |
| 8. | cultural attitudes towards gender | 5 | 0 | 0 | 0 |
| 9. | measures on societal level | 5 | 0 | 0 | 0 |
| 10. | lack of education and awareness | 4 | 0 | 2 | 2 |

We find that Llama3 has a tendency to make up reasons for information that was not given. In the example above it assigns a grade of 5 (best grade) to the 4th question with the following rationale:

> 5: The answer is highly relevant, complete, and accurate. The passage **clearly highlights the impact of privilege and power dynamics** on the prevalence of sexual assault, **stating that these dynamics can perpetuate a culture of silence, shame, and victim-blaming**, ultimately contributing to the underreporting of sexual assault. The passage also notes that privilege and power dynamics can influence the way sexual assault is perceived, prosecuted, and punished, with those in positions of power often being able to avoid accountability for their actions"

In the second attempt, Llama3 assigns the same high grade with an equally incorrect rationale.

> "5. The passage provides a clear and comprehensive explanation of **how privilege and power dynamics contribute to the prevalence of sexual assault**, including the ways in which societal norms, institutional systems, and individual attitudes perpetuate and normalize sexual violence. The passage is well-organized, accurate, and provides specific examples to support its claims, making it an excellent answer to the question."

In the light of these issues, we cannot recommend to use Llama3 (at least not this model without careful fine-tuning) for evaluation with the RUBRIC Autograder Workbench. Hence, we only distribute RUBRIC evaluation scores with FLAN-T5.

## 6.3 Phase 3. RUBRIC Evaluation Metrics

### 6.3.1 RUBRIC-Cover Evaluation

Based on the set of rubric-graded passages, we provide several evaluation metrics. In line with the earlier approaches [Sander and Dietz, 2021], we provide a coverage-based score.

The RUBRIC-Cover@20$\geq$ 4 is based on how many questions from the rubric were answered with an answerability grade $\geq 4$ with passages in the top 20. This grade cut-off of 4 represents answers that are "mostly relevant and complete", or "highly relevant, complete, and accurate".

To avoid that systems obtain a higher evaluation score merely by returning longer responses, only passages in the top 20 are considered for this evaluation metric, which coincides with the allowed number of generated passages in augmentation and generation tasks.

This evaluation metric does not award redundancy, in that once a rubric question is covered by one passage, any subsequent passage that also answers it will not improve the score. This metric prefers systems that address a range of aspects that are relevant for the query.

### 6.3.2 RUBRIC-Qrels Evaluation via `trec_eval`

This metric is based on passage-wise relevance, where relevance labels are derived from the rubric-grades. Ideally, the grade-to-label mapping would be calibrated, however without available training data we resort to a simple heuristic used in Farzi and Dietz [2024c]): The highest answerability grade a passage obtained on any rubric-question is used as the relevance label, yielding a multi-relevance grade from 0 to 5.

We export relevance labels in the `trec_eval`-compatible "qrels" file, and use `trec_eval` to obtain system evaluation scores.[1] We report P@20 at grade $\geq 4$ (`qrels-4`) and, following guidance from track organizer, NDCG@20 (`qrels-ndcg`).

### 6.3.3 Reported Results

We report macro-averaged mean and standard-error for each system, and also provide min/-max/median across the leaderboard. Given the issues with `Llama3` we omit those results and only provide leaderboards obtained with `FLAN-T5`.

## 7 Rubric Evaluation Results for TREC RAG participants

We offer the following RUBRIC-based evaluation measures:

**cover-1:** RUBRIC-Cover with minimum grade level 1 (on 0–5 scale)

**cover-4:** same but minimum grade level 4

**cover-5:** same but minimum grade level 5

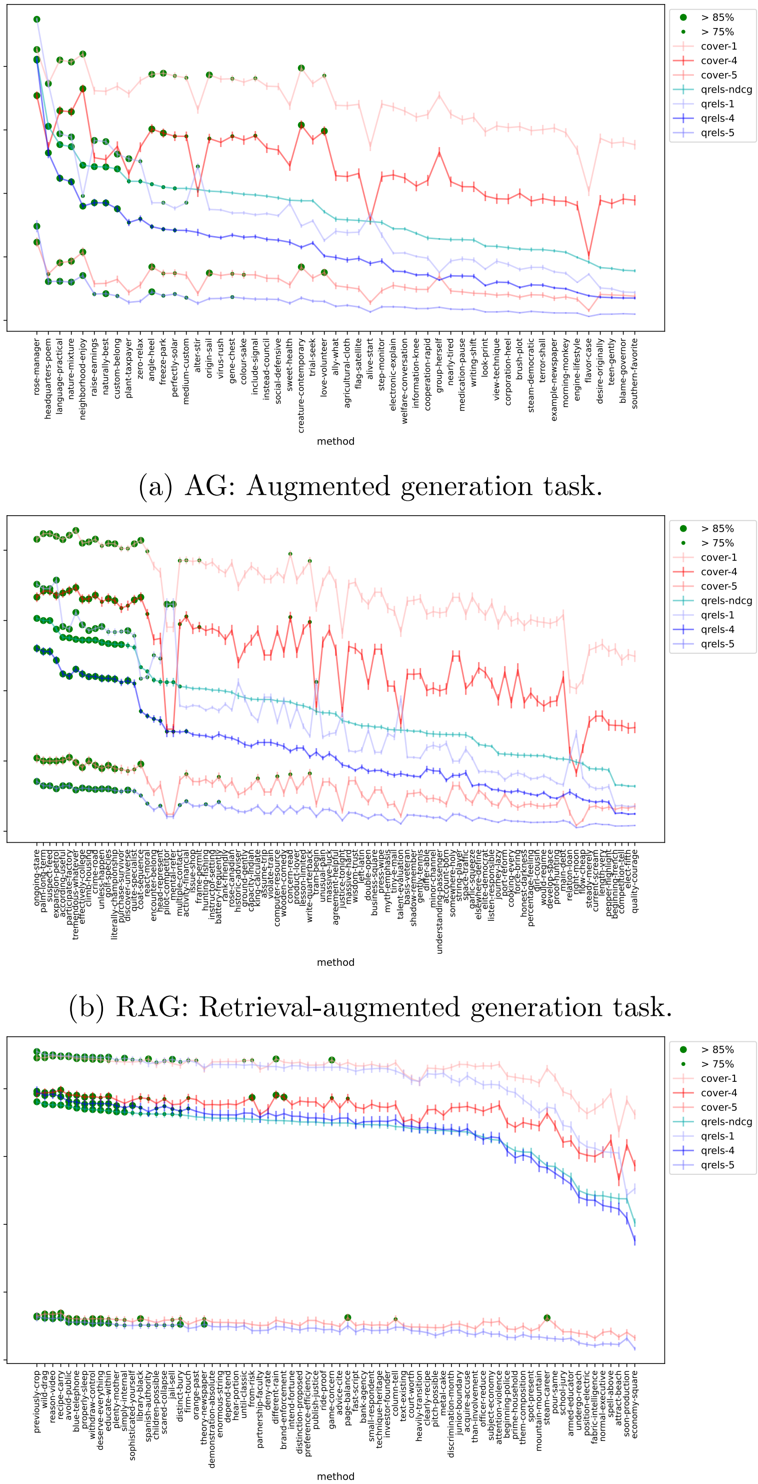
**qrels-ndcg:** RUBRIC-qrels with NDCG@20 metric (ndcg uses the multi-relevance levels 0–5 following our grading scale)

**qrels-1:** RUBRIC-qrels with P@20 with a minmum grade level 1

**qrels-4:** same but minimum grade level 4

**qrels-5:** same but minimum grade level 5

[1] For each system, we emit a run-file based on MS Marco or MD5Hash-based passage-ids generated with code in Table 7.



(a) AG: Augmented generation task.

(b) RAG: Retrieval-augmented generation task.

(c) R: Retrieval task.

Figure 2: RUBRIC results for RAG'24, sorted by `qrels-ndcg`.

In our online appendix,[2] we provide all tabular results, run and qrels files for RUBRIC-qrels, as well as a grade annotations in RUBRIC-workbench data format (described in Dietz [2024], Figure 4). We also include python code and jupyter notebooks to reproduce and further analyze the results.

RUBRIC results are provided for submitted systems in Figures 2. The X-axes presents participating systems with standard error bars that depict the variance across queries. In Figure 2 participant systems are sorted by `qrels-ndcg` performance, where Figure 3 presents the same analysis, but systems are sorted by `cover-4`.

Any system that performed above the 85%-percentile (or 75%-percentile) is marked with a green dot, to illustrate how many "very good systems" would be promoted if this measure had been chosen.

The analysis shows a general agreement of different measures on selecting the best systems. Of course, the coverage-based metrics that do not award redundancy in the generated passages, prefer different systems than precision-based metrics. Each of these groups strongly correlate with each other. We find that the grade threshold of 5 is only rarely assigned by the LLM, and hence not sensitive enough for comparing systems robustly. On the other hand, a `cover-1` is often too lenient to differentiate systems.

**Systems with divergent results.** We are taking a closer look at systems where the cover-4 and qrels-ndcg score diverge the most. For example we inspect the system `pilot-competitor`, which was submitted to the generation task, because it has a low coverage score but a relatively high precision and NDCG score.

For query `2024-145979` “what is vicarious trauma and how can it be coped with?”, the system produces the multi-passage output given in Table 12b, for reference the rubric questions are listed in Table 12a. Several passages are not considered to have answered any of the rubric questions, resulting in only six (of 20) passages that were identified are relevant. However the 1st, 3rd, 5th, and 10th rubric-questions are covered with grade ($\geq 4$) by at least one passage, leading to a relatively high coverage 4/10, but low precision score.

We note that this example demonstrates one of the current weaknesses of the RUBRIC system. Relevant information provided by the system may be missed, because grading proceeds on one passage at a time. However, without the missing context the relevance of the passage cannot be detected. In this example the passage “Exercise to relieve stress.” is an answer to question Q3 “What are some recommended coping strategies for dealing with vicarious trauma?”, but its relevance is not observable without the context of the first passage of the system’s response.

[2] Online Appendix: `https://trec-car.cs.unh.edu/rubric-trec-rag24/index.html`

(a) Rubric questions for query `2024-145979` “what is vicarious trauma and how can it be coped with?”

Q1. What are some common symptoms of vicarious trauma?

Q2. How does vicarious trauma differ from burnout?

Q3. What are some recommended coping strategies for dealing with vicarious trauma?

Q4. Can vicarious trauma impact personal relationships outside of work?

Q5. Is professional therapy or counseling usually recommended for those experiencing vicarious trauma?

Q6. How can organizations better support individuals who may be experiencing vicarious trauma?

Q7. What role does self-care play in coping with vicarious trauma?

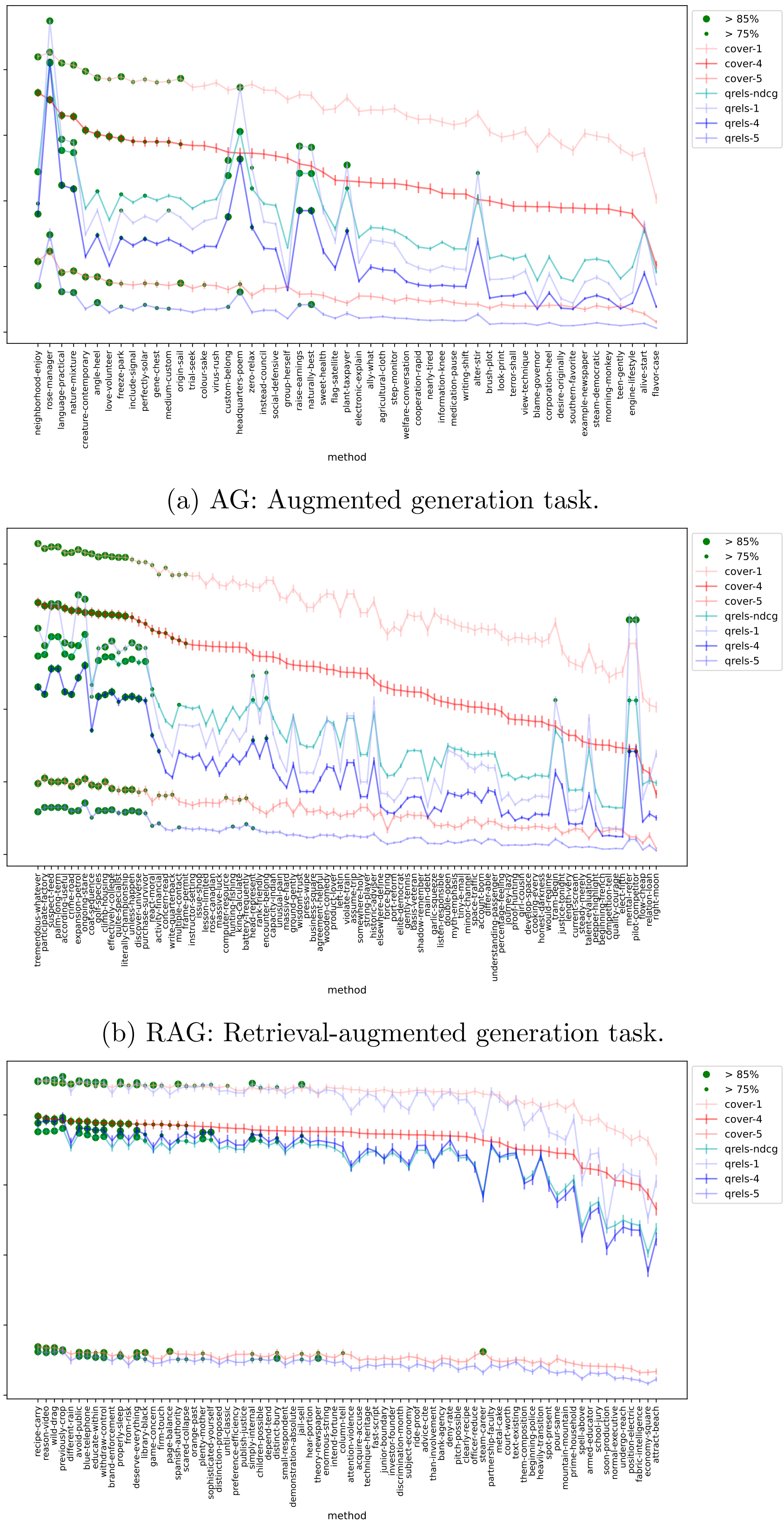
Q8. Are there specific industries or professions more prone to vicarious trauma?

Q9. How can individuals identify their own triggers for vicarious trauma?

Q10. What resources are available for preventing and managing vicarious trauma in the workplace?

(b) Multi-passage output generated by example system “pilot-competitor” for query `2024-145979`, annotated with questions that are positively graded by FLAN-T5. For example “Q1 (4)” means that the answerability of question 1 was awarded grade 4 (mostly relevant and complete).

| Passages of Example System | Answerability grades |
|---|---|
| Vicarious trauma is indirect exposure to traumatic events. | - |
| Symptoms include sadness, grief, irritability, and mood swings. | Q1 (4) |
| Coping strategies include therapy and professional help. | Q3(4) Q5(4) Q10(4) |
| Maintaining work-life balance is crucial. | - |
| Support from colleagues and supervisors is important. | - |
| Training and supervision reduce risk of vicarious trauma. | Q10(4) |
| Recognize you are not alone in your pain. | - |
| Practice breathing and visualization exercises. | - |
| Seek support and engage in hobbies. | Q3(4) |
| Balance priorities and follow a routine. | - |
| Allow yourself to take time off work. | - |
| Develop a meditation practice. | - |
| Exercise to relieve stress. | - |
| Create time and space for reflection. | - |
| Common in healthcare, law enforcement, and social work. | - |
| Can lead to anxiety, depression, PTSD, and addiction. | Q1(4) Q5(4) |
| Affects personal beliefs about self and the world. | - |
| Can cause compassion fatigue and burnout. | - |
| Self-care practices like journaling and hobbies help. | Q3(4) |



(a) AG: Augmented generation task.

(b) RAG: Retrieval-augmented generation task.

(c) R: Retrieval task.

Figure 3: RUBRIC results for RAG'24, sorted by `cover-4`.

# A Appendix: An Example for EIFRP method

Following is an example that explores the application of the EIFRP method to the query: *how would advance electronics course impact students* with query number ”2024-96485”. We will explore examples for steps in detail and discuss some limitations of the method, particularly in dealing with non-informative sentences that contain keywords.

# B Step-by-Step Process

## B.1 Retrieval

Five relevant passages out of all given retrieved passages from the MS MARCO dataset are chosen. Since `llama3` takes a long time to run the EIFRP method on an A40 GPU, we restricted the results to the top 5 but these can be extended to more. These passages were analyzed for information addressing the question of whether the Crusades were a failure or success for Christians. Retrieved passages are:

```
0:  "msmarco_v2.1_doc_39_175110992#8_358415440",
1:  "msmarco_v2.1_doc_32_125454144#0_288309808",
2:  "msmarco_v2.1_doc_34_500919190#4_1087647590",
3:  "msmarco_v2.1_doc_41_1887205925#9_3020770208",
4:  "msmarco_v2.1_doc_41_1887205925#10_3020772551"
```

## B.2 Keyfact Extraction and Verification

This step involves the verification and extraction of key facts. Below is one key fact extracted from the first paragraph. Key facts will be updated with citations, and new key facts will also be added as necessary.

- edX offers self-paced introductory and advanced courses in electronics at every level.
- The Tokyo Institute of Technology offers an introductory course, Introduction to Electrical and Electronics Engineering, which covers the interactions of electrical power, energy, and environment.

## B.3 Removal of Uninformative Content

The EIFRP method uses both rule-based and prompt-based strategies to filter out non-informative sentences. However, challenges persist in achieving complete relevance. Although the LLM-based refinement step aims to remove uninformative content, it sometimes generates explanations justifying why certain sentences were excluded. These explanations can inadvertently add unnecessary content or fail to fully filter out non-relevant information. This challenge arises because LLaMA3, as an LLM, often generates justifications or clarifications for its decisions, based on observations from our experiment.

**Example:** *Note: There are no other sentences in the document that directly address the query or are relevant to the impact of an advanced electronics course on students.*

## B.4 Smoothing and Final Output Generation

The remaining key facts were smoothed to improve coherence and readability. This involved rephrasing and adjusting sentence structures to ensure the facts were presented logically and fluently. We show the same extracted keyfacts after being smoothed out.

- edX offers self-paced introductory and advanced courses in electronics at every level.
- The Tokyo Institute of Technology offers an introductory course, Introduction to Electrical and Electronics Engineering, which covers the interactions of electrical power, energy, and environment, providing a solid foundation for students to build upon the foundational knowledge gained from edX's self-paced courses.

# References

Laura Dietz. A workbench for autograding retrieve/generate systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1963–1972, 2024.

Naghmeh Farzi and Laura Dietz. Best in tau@ llmjudge: Criteria-based relevance evaluation with llama3. *arXiv preprint arXiv:2410.14044*, 2024a.

Naghmeh Farzi and Laura Dietz. Exam++: Llm-based answerability metrics for ir evaluation. In *Proceedings of LLM4Eval: The First Workshop on Large Language Models for Evaluation in Information Retrieval*, 2024b.

Naghmeh Farzi and Laura Dietz. Pencils down! automatic rubric-based evaluation of retrieve/generate systems. In *Proceedings of the 2024 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 175–184, 2024c.

Yiqi Liu, Nafise Sadat Moosavi, and Chenghua Lin. Llms as narcissistic evaluators: When ego inflates evaluation scores. *arXiv preprint arXiv:2311.09766*, 2023.

David P Sander and Laura Dietz. Exam: How to evaluate retrieve-and-generate systems for users who do not (yet) know what they want. In *DESIRES*, pages 136–146, 2021.