The Effectiveness of Retrieval Metrics in Evaluating the Results of Retrieval-Augmented Generation in the Question-Answering Domain

Vidya Srinivas
University of Washington
vysri@cs.washington.edu

Keisuke Kamahori
University of Washington
kamahori@cs.washington.edu

Yue Wu
University of Washington
yuew29@uw.edu

Khyati Morparia
University of Washington
khyati3@uw.edu

Abstract

Retrieval augmented generation aims to augment an LLM with external knowledge from a non-parametric database to enhance the reliability of its produced results for a given query. While retrieval systems have been shown to provide performance enhancements as compared to their purely parametric counterparts, it still remains a challenge to effectively and accurately evaluate the performance of these models as a human would. This is mainly due to the fact that syntax-based accuracy evaluation metrics fail to capture acronyms, synonyms, and semantically similar answers that humans would consider correct answers. This paper investigates the scenarios in which these failures occur and the difference between human-perceived accuracy of retrieval systems and syntax-based evaluation accuracy of these systems.

1 Introduction

The recent progress of large language models (LLMs) has given rise to much research in the direction of improving the performance and generalizability of these models. Pretrained LLMs demonstrate impressive capabilities but still suffer from outdated knowledge, hallucinations, and lack of transparency in the reasoning process. A recent paradigm, retrieval augmented generation (RAG), aims to address these challenges through the use of synergistic non-parametric memory and parametric memory to leverage the strengths of both external data stores and LLMs. While retrieval is being integrated into many systems due to the benefits above, it still remains a challenge to effectively evaluate a retrieval system in question-answer scenarios.

The main purpose of RAG systems for question answering is to allow the end user to obtain a more accurate answer from an LLM than they would just through the LLM with only parametric memory. Thus, the evaluation of such systems must reflect the human-perceived accuracy of the answer. Specifically, given a question and the answer generated by a retrieval system, one would want to evaluate how close the answer is to the ground truth answer provided in an evaluation dataset. These evaluation datasets usually provide questions that are to be answered and a ground truth answer or set of answers. The answer generated by a retrieval system is then compared to these answers in the evaluation process. However, this poses many challenges, due to the facts that answers can we formatted differently than what is given in the ground truth (i.e. acronyms), answers can have synonyms that are not recorded in the ground truth answer(s), and answers that semantically reflect the ground truth answer, but do not appear to be the same in plain text. Thus, metrics used to evaluate retrieval systems that only focus on the syntax of the generated answer and ground truth answer, such

CSE 547 (Spring 2024) Class Project

as exact match (EM), F1-Score, precision, and recall, fail to accurately capture the performance of retrieval systems.

We pose the following research questions:

1. In what scenarios do syntax-based metrics used for retrieval evaluation fail to capture the correctness of the answer provided by a retrieval system?
2. Do retrieval metrics reflect variation in performance of retrieval across multiple question types, such as more open-ended questions and choice-based quesions?
3. How does the performance of retrieval change given that all the relevant context to answer the given question is provided versus not provided?
4. What is the deviation of the estimates of performance provided by the metrics as compared to the estimates of performance calculated from human annotated samples?

Obtaining answers to the above questions can inform the creation of metrics that better capture the semantic relationships between ground truth and generated answers, which can allow future proposed retrieval systems to be evaluated more accurately and efficiently. Using the performance of an LLM without a non-parametric datastore as a baseline, we investigate the performance improvement of the same LLM augmented with a datastore that retrives facts relevant to the given question using syntax-based evaluation metrics. We then annotate a subset of questions, with 90% accuracy, into different categories using a different LLM with more parameters, and determine whether the performance of the syntax-based metrics changes across question categories. Finally, we human-annotate a subset of our dataset to see how accurately the syntax-based metrics reflect human-perceived question answering abilities.

2 Related Work

The original RAG paper (Lewis et al. [2020]) proposes an architecture that aims to provide a unified structure for NLP retrieval tasks that will work on a variety of tasks, including open-domain question answering, abstractive question answering, Jeopardy question generation, and fact verification. The proposed architecture consists of a neural retriever that retrieves the top-K latent documents conditioned on a query input. Then, a generator generates new tokens based on the context of the previous tokens, the original query input, and the retrieved documents. In doing so, this architecture allows the output of the LLM to be refined through knowledge from an external database, allowing it to be applied to a wider variety of tasks with better performance.

The initial retrieval step in a RAG system often returns multiple documents or passages that may not all be equally relevant to the input query. Re-ranking is the process of scoring and sorting these retrieved results to prioritize the most relevant ones. Several metrics have been proposed to evaluate the retrieval component of RAG systems. A common approach is to treat retrieval as a ranking problem and use standard information retrieval metrics like precision at k, recall at k, and mean average precision (MAP) . These metrics measure how well the retriever ranks relevant documents at the top for a given query. Another line of work proposes graded relevance metrics that go beyond just binary relevance judgments . Metrics like normalized discounted cumulative gain (NDCG) assign graded relevance scores to retrieved documents based on their utility for answering the query. This better captures partial relevance compared to precision/recall. However, human evaluation is also important to assess if the re-ranked results lead to better final outputs from the RAG system.

Evaluating the generated text output is another crucial aspect in retrieval. N-gram overlap metrics like ROUGE Lin [2004] are commonly used, but have known limitations in capturing semantic similarity. More recent work uses contextualized embeddings from pre-trained language models to better measure semantic relevance, such as BERTScore Zhang et al. [2020] and its extensions. Factual consistency Maynez et al. [2020] and hallucination detection are other important considerations for RAG systems, since the generated text should be faithful to the retrieved evidence. Human evaluation remains the gold standard but is expensive to conduct at scale.

ROUGE is a set of metrics that measure the overlap of n-grams between the generated text and reference texts. It is widely used for evaluating text summarization and generation tasks. The most common variants are ROUGE-N, which measures the n-gram overlap between the generated and reference texts, where higher scores indicate better word-level similarity, and ROUGE-L, which calculates the longest common subsequence between the generated and reference texts, giving credit to in-sequence word matches. Higher ROUGE scores generally correlate with better content coverage in the generated output compared to the references.

BERTScore computes a similarity score between the generated and reference texts using contextual embeddings from a pre-trained BERT model. It captures semantic similarity better than surface-level n-gram overlap metrics like ROUGE. The BERTScore is the cosine similarity between the BERT embeddings of the generated and reference texts. Higher scores indicate better semantic equivalence.

For knowledge-intensive RAG tasks, it is crucial to evaluate if the generated output is factually consistent with the retrieved evidence/context. Factual consistency score measures the degree to which the generated text contradicts or is unsupported by the evidence. Hallucination rate quantifies the fraction of generated outputs that contain hallucinated, or ungrounded, facts. These metrics are often calculated using model-based classifiers or human annotations.

3 Models and Methods

3.1 RAG System Overview

We give a brief overview of the main parts of retrieval in Figure 1. The query is obtained from the user, or in this case, a dataset, and passed to an embedding model which converts it to a different representation for fast search. The embedded query is then passed through an index search which searches a datastore for the top-k most relevant passages or documents. These retrieved documents are then concatenated with the original query in the augmentation process. The augmented query is passed to a large language model (LLM) through its prompt to generate the final answer to the query. The overall structure of the RAG system has many components that work together before the final response is generated. The main challenge with such a system is to actually retrieve documents that are relevant to the the query and ensure that they are the best documents that could have been retrieved for the given query. This poses different challenges that we will discuss later in this report.

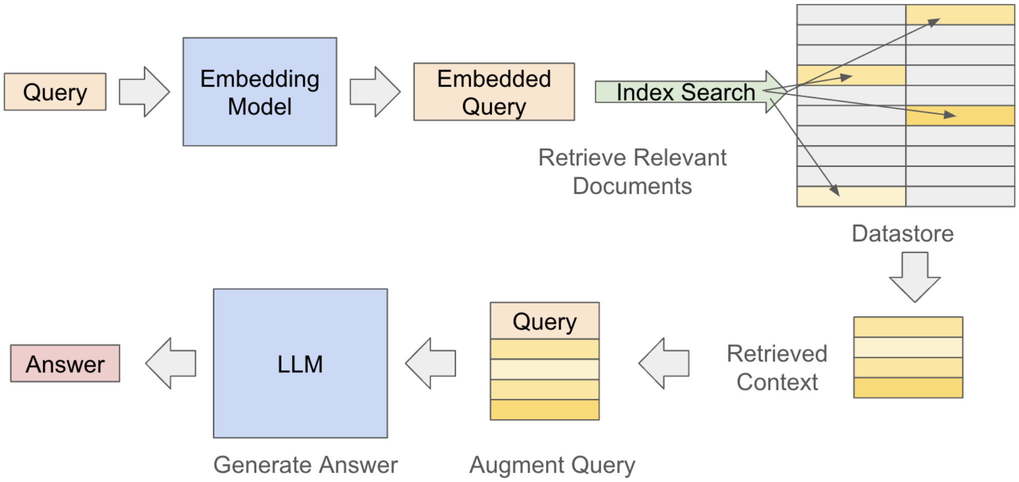

Figure 1: Overview of RAG System

3.2 Baseline Model and Implementation Details

We use meta-llama/Meta-Llama-3-8B-Instruct AI@Meta [2024] from Hugging Face as our baseline model in the implementation. This model is developed by Meta as a member of Meta Llama 3 family of large language models (LLMs), a collection of pretrained and instruction tuned generative text models in 8 and 70B sizes. The Llama 3 instruction tuned models are optimized for dialogue use cases, and only take text as input and text and code as output. Supervised fine-tuning and reinforcement learning with human feedback are used to train this model to align with human preferences for helpfulness and safety. Each retrieval involves three parts: tokenization, generation, and decoding. System prompt and user prompt are passed into the tokenizer so that tokens in the text become torch tensors. Then we pass these tensors into Llama3 model to let it generate a response. Finally we decode the outputs from the model with tokenizer again.

3.3 Dataset Collection

We use HotpotQA Yang et al. [2018], a question answering dataset featuring natural, multi-hop questions, with strong supervision for supporting facts to enable more explainable question answering systems. The dataset is generated by crowdsourcing based on Wikipedia articles, where crowd workers are shown multiple supporting context documents and asked to come up with questions that contain some part of each document to generate multi-hop questions, or questions that require information from different passages to be connected in order to answered accurately. We chose this dataset because it is open source and also is used by a variety of other RAG work as an evaluation set.

For the purposes of this project, we run our evaluation on a subset of the dev-distractor set, as the provided test set does not contain ground truth responses. The dev distractor set is a json file that contains a list of dictionaries of question-answer sets. For each item in the dev distractor list, there are key-value pairs including

1. **_id:** a unique id for this question-answer datapoint.
2. **answer:** a string that answers the question.
3. **question:** a string that asks a question.
4. **context:** a list. Each item in the list is a paragraph, which is represented as a list with two elemnts [title,sentences]. "sentences" is a list strings of sentences in this paragraph.
5. **supporting facts:** a list. Each item in the list contains a two-element list [title,sent_id], where titile denotes the title of the paragraph, and sent_id denotes the supporting fact's id in this paragraph.

Context provides passages that are relevant to the query and passages that are also not relevant, but may be confusing if not interpreted correctly, testing the robustness of a model. For easy retrieval, we preprocess the data by creating new dictionaries with each key as "_id" and value as the corresponding dataponit's dictionary. All further evaluation presented in this paper is run on 1250 out of the 7405 samples present in this validation set, as it requires some degree of human evaluation. We leave it to future work to do a more exhaustive evaluation using a larger number of samples.

3.4 Metric Description

For evaluation steps, we use EM, F1-score, precision, and recall as metrics. For a candidate sentence and a reference sentence, precision, P, is defined as

$$P = \frac{\#exact\_matches}{total\_\#\_of\_n{-}grams\_in\_candidate}$$

recall, R, is defined as

$$R = \frac{\#\_exact\_matches}{total\_\#\_of\_n{-}grams\_in\_reference}$$

We use the harmonic mean of precision and recall to compute f-1 score, F_1:

$$F_1 = \frac{2}{P^{-1} + R^{-1}}$$

For each metric, we take the average score across all prediction-ground truth pairs.

3.5 Baseline Testing

We first establish a baseline for model performance based on two retrieval configurations run on HotpotQA. The first configuration is purely based on LLM parametric memory. We use Meta's

Llama3-8B AI@Meta [2024] as our LLM. We run the query with the prompt “You are a qa test machine, you need to answer the [Question], you only need to come out the correct answer without anyother words." We then test the performance of Llama-3 with gold passage retrieval. Gold passages are those which are the most relevant from Wikipedia to our question, as determined by the authors of the Hotpot dataset, who used crowdsourcing to generate the questions. We run our system with the query “You are a qa test machine, you need to answer the [Question] from given the [Context], you only need to come out the correct answer without anyother words." We would expect that this configuration shows significant improvement over the first configuration given that teh LLM now has more relevant context for answering the question. We measure by four main metrics: exact match score, F1-score, precison and recall.

To calculate the above metrics properly, the ground truth responses and the LLM output first had to be normalized. We apply the following steps to do so. All the strings were converted to lowercase so that case would not cause our matching-based metrics to return false negatives. All punctuation was removed from strings, again to avoid false negatives. Articles such as ‘a’, ‘the’ were removed and whitespace was fixed to be consistent between strings (a space between each word).

During baseline testing, we also conducted a prompt evaluation to determine whether the provided prompt changed the perceived performance of the system as measured by the metrics. This was essential as syntax-based metrics perform n-gram comparisons (compare sets of n words to find matches). Therefore, all answers had to be in the most concise form possible. We used the following prompts for our evaluation using the second configuration presented above (LLM augmented with gold context).

1. “You are an assistant for question-answering tasks. Use the following pieces of retrieved context to answer the question. If you don’t know the answer, just say that you don’t know. Answer the query with a succinct response within 10 tokens."
2. “Only answer every question with 10 words, does not need to be full sentences. Use the following pieces of retrieved context and your existing knowledge to answer the question with 10 words max. If the context does not help, ignore it."
3. “You are a qa test machine, you need to answer the [Question] from given the [Context], you only need to come out the correct answer without anyother words."

3.6 Question Type Annotation

After establishing baselines, we hypothesize that the metrics that we use for evaluation will reflect that retrieval benefits certain types of questions more than others. Specifically, we hypothesized that retrieval would better benefit open-ended question types rather than yes/no or true/false type questions, as these types of questions tend to be more challenging to answer and require more information to answer. The question types were chosen as follows:

1. **Fine-grained (FG):** question contains additional information about the entity that it is asking about
2. **Coarse-grained (CG):** question asks about an entity with no additional information
3. **Yes/no (YN):** question can be answered as either “yes" or “no" or in one word
4. **Multiple choice (MC):** question provides multiple choices and asks to choose one
5. **Time frame (TF):** question asks for a date or a range of dates or times
6. **Comparisons (CMP):** question is a comparison that requires context about both choices

These question types were chosen from a series of papers that generate datasets for question answering, including Yang et al. [2018], Joshi et al. [2017], Yang et al. [2015], and capture the general question themes that are considered in these papers which are thought to encapsulate the variety of questions available in popular QA datasets. Each of the question types were specified in a prompt to GPT 3-5 along with descriptions and examples of each. 1250 examples from the HotpotQA dataset were then provided and were annotated with a type. These served as the question tags for the future evaluations.

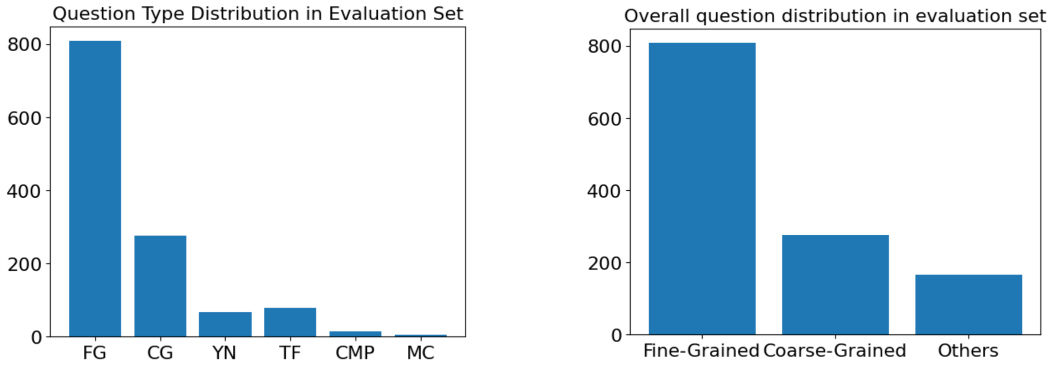

Figure 2: The left graph shows distribution of all six questions types in the evaluation set; the right graph shows the distribution for fine/coarse grained and other (choice-based) questions

3.7 Human Annotation

To compare how well the previously mentioned metrics reflect human evaluation, we hand-annotate a randomly selected sample of 100 questions. For each of these questions, we provide the human annotator the question, the GPT 3-5 annotated question type, the LLM-provided answer, the LLM answer when given gold context passages, the gold context passages, and the ground truth answer. We then ask the user four questions: if the annotated question category is right, if the LLM answer matches the ground truth answer, if the LLM answer with gold context matches the ground truth answer, and if the context contains *all* the necessary information needed to provide an answer to the question. For each of the questions above, the user was given a choice of a 0 or 1 binary response. Each user was provided with a full description of the question type annotations as well as a full description of the questions that we wanted them to answer. We then calculated metrics conditioned on the different human-annotated factors to provide a holistic view of the performance of RAG under varied conditions.

4 Results and Findings

The following section describes the results of the experiments that we conducted in the previous section and insights that were drawn during the experimental and evaluation process. We conducted three types of evaluation: baseline evaluation, question type-based evaluation, and human annotation evaluation. We also present a small case study that details some of the RAG failure modes discovered during evaluation.

4.1 Baseline Evaluation

Our results match our intuition that if the LLM is supplemented with relevant information for a given query, it can outperform the LLM that does not have this additional context. In our initial evaluations, we measure four main metrics: exact match score, F1-score, precision and recall. Table 4.1 below displays the results of our evaluations.

Table 1: Results of baseline evaluations. “LLM" refers to generation using only Llama3-8B while “Gold" refers to generations taking into account gold passages.

Name	EM	F1-Score	Precision	Recall
LLM	0.1912	0.2698	0.2964	0.2619
Gold	0.3672	0.4857	0.5206	0.4808

Additionally, we conducted a prompt evaluation with the configurations specified in Section 3.5. The results of our evaluation are in Table 4.1 below.

Table 2: Results of using different prompts for the LLM prompt augmented with gold context passages. The prompts are as described in Section 3.5. More verbose prompts lead to longer responses which reduce the performance as measured by syntax-based metrics.

Name	EM	F1-Score	Precision	Recall
Prompt 1	0.0100	0.1560	0.1042	0.4705
Prompt 2	0.0100	0.2212	0.1508	0.5445
Prompt 3	0.3672	0.4857	0.5206	0.4808

From our experiments, we observe that it is better to use fewer words for the prompt and not specify the instructions in verbose language. Tagging the question and context in the prompt helps performance. Prompt 3 generates the best results with the lowest latency. Given other prompts, the LLM generates extremely verbose response that is often incorrect even though it has access to gold passages which generally contain the answer to the query; this also downgrades performance with respect to purely syntax-based evaluation metrics. Our evaluation indicates that the quality of the prompt can significantly affect the quality of a RAG system evaluation based on purely syntax-based metrics.

4.2 Question-Type Annotation

We describe the results of the performance of syntax-based metrics on different types of questions. Question type annotations were provided by GPT 3-5. On an evaluation sample of 100 human annotated samples, we found that the accuracy of the GPT 3-5 based annotation is around 90%. We present the results of evaluation with syntax-based metrics conditioned on question type in Table 4.2. The distribution of questions types in our 1250-question evaluation set is 64.64%, 22.24%, 5.36%, 6.32%, 1.12%, 0.40%, for FG, CG, YN, TF, CMP, MC, respectively.

Overall, it is clear that the LLM with gold context outperforms the LLM without gold context, raising the metrics by 15.30%, 21.28%, 21.60%, and 22.57% across EM, F1-score, precision, and recall, respectively for all types of questions. However, the standard deviation of the increase in question types is in the 1-7% range, indicating that retrieval does not seem to benefit any one type of question much more than the other, which is contrary to our expectations. Generally, we expect to see the trend that retrieval and gold context passages will benefit more open-ended question types significantly more that closed question types such as multiple choice and yes/no questions. We explore the reasons for this in the next section.

Table 3: Performance conditioned on GPT 3-5 annotated question type. The names are as described in Section 3.5. “LLM" refers to generation using only Llama3-8B while “Gold" refers to generations taking into account gold passages.

Name	EM	F1-Score	Precision	Recall
Fine-grained (LLM)	0.1513	0.2399	0.2676	0.2313
Fine-grained (Gold)	0.3395	0.4773	0.5159	0.5159
Coarse-grained (LLM)	0.1920	0.2685	0.2950	0.2597
Coarse-grained (Gold)	0.3696	0.4858	0.5197	0.4818
Yes/no (LLM)	0.1804	0.2578	0.2826	0.2501
Yes/no (Gold)	0.3618	0.4864	0.5236	0.4793
Multiple choice (LLM)	0.6000	0.6000	0.6000	0.6000
Multiple choice (Gold)	0.6000	0.7333	0.7000	0.8000
Time frame (LLM)	0.1446	0.2278	0.2545	0.2196
Time frame (Gold)	0.3363	0.4693	0.5095	0.4614
Comparisons (LLM)	0.1896	0.2682	0.2948	0.2603
Comparisons (Gold)	0.3688	0.4871	0.5220	0.4822

4.3 Human Annotation and Case Study

To determine why our hypothesis was incorrect, we conduct a case study on 100 samples with the help of a human annotator. We record whether the LLM response matches the ground truth, whether the Gold response matches the ground truth, whether our question type annotation is accurate, and whether the relevant context to answer the question is fully contained in the gold context passages. Particularly, this means that even if partial relevant context was contained in the passages, if the full question could not be answered from the passages, this was considered as not having the context to answer the question.

From human annotation evaluation, we find that the LLM-only approach is correct 22% of the time and the LLM augmented with the gold context is correct 69% of the time. However, if we further explore by conditioning on the quality of the context passages, we discover that the quality of the passages matters significantly. Specifically, referring to Table 4.3, when all the context is available to answer the question in the context prompts, retrieval offers a 46% increase in accuracy over the LLM-only approach (rows 1 and 2). Even in cases with partial context, retrieval offers a 33% increase in accuracy. However, the difference in the gold context performance in both cases is quite notable, with the performance being 28% lower in cases with only partial context. For clarification, we specify that the Gold and LLM configurations are the same as mentioned earlier; the full and partial context are only mentioned to distinguish the different groups.

Table 4: Results of using human annotation as a metric for LLM methods and augmented with gold context passages method. The names are as described in Section 3.5. More verbose prompts lead to longer responses which reduce the performance as measured by syntax-based metrics.

Name	Accuracy
Gold correct on questions with full context	0.7674
LLM correct on questions with full context	0.3023
Gold correct on questions with partial context	0.4912
LLM correct on questions with partial context	0.1579

We collected samples of failure-mode cases for evaluation syntax-based metrics during human annotation. In particular, we found that some questions were badly formed (incorrect grammar, repeated words), and the answers were incorrect in some of these cases. Some answers as recorded in the dataset were not recorded correctly. The most interesting case was the one in which gold context passages contained partially relevant information to the answer, but in which multi-hop question answering could not be verified (i.e. the question was not answerable based on only the context in the passages).

5 Discussion

In this section, we discuss our results and address our research questions. Syntax-based metrics for retrieval fail to capture the correctness of the answer provided in scenarios where the generated answer is too long, contains synonyms or abbreviations, or contains information that semantically captures the gist of the answer but does not exactly match the provided ground truth answers. Though related work addresses this concern, it is still an important area of research to determine what kind of automatic evaluation metrics are available for these scenarios, as they are still annotated by human evaluators in most cases.

The syntax-based metrics do not show large variation across different question types. According to human evaluation, this seems to be because the context provided does not fully encapsulate the information needed to answer multi-hop questions. Assuming that perfect context was provided for all questions, we would expect that retrieval would provide the most benefit to fine-grained questions and comparison questions, and then the other types of questions, due to the fact that fine-grained questions contain more relevant information in the question prompt.

The performance of retrieval is significantly better when the provided context is good and captures all parts of the question. Certain related works aim to evaluate faithfulness of the answer to the context passages provided, but it is also important to evaluate whether the retrieved passages were the best passages that could be captured from the external datastore. This would allow retrievers to be evaluated standalone, which would provide a large benefit during evaluation, conditioning on the retriever quality.

Finally, syntax-based metrics seem to underestimate the performance of retrieval, and also do not take into account many factors mentioned before, such as the quality and correctness of the retrieved passages. Determining a robust and accurate set of metrics that enable auto-evaluation quality close to that of human annotators still remains an open research area.

6 Individual Contributions

- Vidya Srinivas: wrote code for prompt testing, question type annotation, evaluation, and human annotation, and baseline evaluation; ran baseline evaluations and wrote analysis, question type evaluations and analysis; human annotations and analysis; annotated all human-evaluated samples; wrote the project final report; generated graphs and tables
- Keisuke Kamahori: Initial proposal of the project, preparation of vector database and index (not used in the final evaluation)
- Yue Wu: Evaluated different metrics, including F1-score, recall, precision (which are used for final evaluation) and ROUGE, BLUE (not included in final evaluation). Wrote up/modified Models and Methods, and data collection parts for the report. Made presentation slides and the video presentation.
- Khyati Morparia: Performed parameter exploration for evaluation, optimizing our processes, and preparing detailed milestone reports. I provided insights, effectively articulated our project's objectives. I also crafted compelling narratives for our final presentation and video, highlighting project background, goals, and plans.

References

AI@Meta. Llama 3 model card. 2024. URL `https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md`.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, July 2017. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33: 9459–9474, 2020.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

Joshua Maynez, Anya Narayan-Chen, Fabio Petroni, Emily Dinan, Andrey Szlam, Douwe Kiela, and Anita de Waard. Faithfulness in retrieval-augmented generation. *arXiv preprint arXiv:2012.15839*, 2020.

Yi Yang, Wen-tau Yih, and Christopher Meek. WikiQA: A challenge dataset for open-domain question answering. In Lluís Màrquez, Chris Callison-Burch, and Jian Su, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2013–2018, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1237. URL `https://aclanthology.org/D15-1237`.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

Tianyi Zhang, Vishrav Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation via contextualized embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 2020.