RAGAS: Automated Evaluation of Retrieval Augmented Generation

Shahul Es[†], **Jithin James**[†], **Luis Espinosa-Anke**[*◇], **Steven Schockaert**[*]
[†]Exploding Gradients
[*]CardiffNLP, Cardiff University, United Kingdom
[◇]AMPLYFI, United Kingdom
shahules786@gmail.com,jamesjithin97@gmail.com
{espinosa-ankel,schockaerts1}@cardiff.ac.uk

Abstract

We introduce **RAGAS** (**R**etrieval **A**ugmented **G**eneration **As**sessment), a framework for reference-free evaluation of Retrieval Augmented Generation (RAG) pipelines. RAG systems are composed of a retrieval and an LLM based generation module, and provide LLMs with knowledge from a reference textual database, which enables them to act as a natural language layer between a user and textual databases, reducing the risk of hallucinations. Evaluating RAG architectures is, however, challenging because there are several dimensions to consider: the ability of the retrieval system to identify relevant and focused context passages, the ability of the LLM to exploit such passages in a faithful way, or the quality of the generation itself. With RAGAS, we put forward a suite of metrics which can be used to evaluate these different dimensions *without having to rely on ground truth human annotations*. We posit that such a framework can crucially contribute to faster evaluation cycles of RAG architectures, which is especially important given the fast adoption of LLMs.

1 Introduction

Language Models (LMs) capture a vast amount of knowledge about the world, which allows them to answer questions without accessing any external sources. This idea of LMs as repositories of knowledge emerged shortly after the introduction of BERT (Devlin et al., 2019) and became more firmly established with the introduction of ever larger LMs (Roberts et al., 2020). While the most recent Large Language Models (LLMs) capture enough knowledge to rival human performance across a wide variety of question answering benchmarks (Bubeck et al., 2023), the idea of using LLMs as knowledge bases still has two fundamental limitations. First, LLMs are not able to answer questions about events that have happened after they were trained. Second, even the largest models struggle to memorise knowledge that is only rarely mentioned in the training corpus (Kandpal et al., 2022; Mallen et al., 2023). The standard solution to these issues is to rely on *Retrieval Augmented Generation (RAG)* (Lee et al., 2019; Lewis et al., 2020; Guu et al., 2020). Answering a question then essentially involves retrieving relevant passages from a corpus and feeding these passages, along with the original question, to the LM. While initial approaches relied on specialised LMs for retrieval-augmented language modelling (Khandelwal et al., 2020; Borgeaud et al., 2022), recent work has suggested that simply adding retrieved documents to the input of a standard LM can also work well (Khattab et al., 2022; Ram et al., 2023; Shi et al., 2023), thus making it possible to use retrieval-augmented strategies in combination with LLMs that are only available through APIs.

While the usefulness of retrieval-augmented strategies is clear, their implementation requires a significant amount of tuning, as the overall performance will be affected by the retrieval model, the considered corpus, the LM, or the prompt formulation, among others. Automated evaluation of retrieval-augmented systems is thus paramount. In practice, RAG systems are often evaluated in terms of the language modelling task itself, i.e. by measuring perplexity on some reference corpus. However, such evaluations are not always predictive of downstream performance (Wang et al., 2023c). Moreover, this evaluation strategy relies on the LM probabilities, which are not accessible for some closed models (e.g. ChatGPT and GPT-4). Question answering is another common evaluation task, but usually only datasets with short extractive answers are considered, which may not be representative of how the system will be used.

To address these issues, in this paper we present **RAGAS**[1], a framework for the automated assess-

[1]RAGAS is available at https://github.com/explodinggradients/ragas.

ment of retrieval augmented generation systems. We focus on settings where reference answers may not be available, and where we want to estimate different proxies for correctness, in addition to the usefulness of the retrieved passages. The RAGAS framework provides an integration with both llama-index and Langchain, the most widely used frameworks for building RAG solutions, thus enabling developers to easily integrate RAGAS into their standard workflow.

2 Related Work

Estimating faithfulness using LLMs The problem of detecting hallucinations in LLM generated responses has been extensively studied (Ji et al., 2023). Several authors have suggested the idea of predicting factuality using a few-shot prompting strategy (Zhang et al., 2023). Recent analyses, however, suggest that existing models struggle with detecting hallucination when using standard prompting strategies (Li et al., 2023; Azaria and Mitchell, 2023). Other approaches rely on linking the generated responses to facts from an external knowledge base (Min et al., 2023), but this is not always possible.

Yet another strategy is to inspect the probabilities assigned to individual tokens, where we would expect the model to be less confident in hallucinated answers than in factual ones. For instance, BARTScore (Yuan et al., 2021) estimates factuality by looking at the conditional probability of the generated text given the input. Kadavath et al. (2022) use a variation of this idea. Starting from the observation that LLMs provide well-calibrated probabilities when answering multiple-choice questions, they essentially convert the problem of validating model generated answers into a multiple-choice question which asks whether the answer is true or false. Rather than looking at the output probabilities, Azaria and Mitchell (2023) propose to train a supervised classifier on the weights from one of the hidden layers of the LLM, to predict whether a given statement is true or not. While the approach performs well, the need to access the hidden states of the model makes it unsuitable for systems that access LLMs through an API.

For models that do not provide access to token probabilities, such as ChatGPT and GPT-4, different methods are needed. SelfCheckGPT (Manakul et al., 2023) addresses this problem by instead sampling multiple answers. Their core idea is that factual answers are more stable: when an answer is factual, we can expect that different samples will tend to be semantically similar, whereas this is less likely to be the case for hallucinated answers.

Automated evaluation of text generation systems LLMs have also been leveraged to automatically evaluate other aspects of generated text fragments, beyond factuality. For instance, GPTScore (Fu et al., 2023) uses a prompt that specifies the considered aspect (e.g. fluency) and then scores passages based on the average probability of the generated tokens, according to a given autoregressive LM. This idea of using prompts was previously also considered by Yuan et al. (2021), although they used a smaller fine-tuned LM (i.e. BART) and did not observe a clear benefit from using prompts. Another approach directly asks ChatGPT to evaluate a particular aspect of the given answer by providing a score between 0 and 100, or by providing a rating on a 5-star scale (Wang et al., 2023a). Remarkably, strong results can be obtained in this way, although it comes with the limitation of being sensitive to the design of the prompt. Rather than scoring individual answers, some authors have also focused on using an LLM to select the best answer among a number of candidates (Wang et al., 2023b), typically to compare the performance of different LLMs. However, care is needed with this approach, as the order in which the answers is presented can influence the result (Wang et al., 2023b).

In terms of how ground truth answers or, more generally, generations, have been typically used in the literature, most approaches have relied on the availability of one or more reference answers. For instance, BERTScore (Zhang et al., 2020) and MoverScore (Zhao et al., 2019) use contextualised embeddings, produced by a pre-trained BERT model, to compare the similarity between the generated answer and the reference answers. BARTScore (Yuan et al., 2021) similarly uses reference answers to compute aspects such as precision (estimated as the probability of generating the generated answer given the reference) and recall (estimated as the probability of generating the reference given the generated answer).

3 Evaluation Strategies

We consider a standard RAG setting, where given a question q, the system first retrieves some context $c(q)$ and then uses the retrieved context to generate an answer $a_s(q)$. When building a RAG system,

we usually do not have access to human-annotated datasets or reference answers. We therefore focus on metrics that are fully self-contained and reference-free. We focus in particular three quality aspects, which we argue are of central importance. First, **Faithfulness** refers to the idea that the answer should be grounded in the given context. This is important to avoid hallucinations, and to ensure that the retrieved context can act as a justification for the generated answer. Indeed, RAG systems are often used in applications where the factual consistency of the generated text w.r.t. the grounded sources is highly important, e.g. in domains such as law, where information is constantly evolving. Second, **Answer Relevance** refers to the idea that the generated answer should address the actual question that was provided. Finally, **Context Relevance** refers to the idea that the retrieved context should be focused, containing as little irrelevant information as possible. This is important given the cost associated with feeding long context passages to LLMs. Moreover, when context passages are too long, LLMs are often less effective in exploiting that context, especially for information that is provided in the middle of the context passage (Liu et al., 2023).

We now explain how these three quality aspects can be measured in a fully automated way, by prompting an LLM. In our implementation and experiments, all prompts are evaluated using the `gpt-3.5-turbo-16k` model, which is available through the OpenAI API[2].

Faithfulness We say that the answer $a_s(q)$ is faithful to the context $c(q)$ if the claims that are made in the answer can be inferred from the context. To estimate faithfulness, we first use an LLM to extract a set of statements, $S(a_s(q))$. The aim of this step is to decompose longer sentences into shorter and more focused assertions. We use the following prompt for this step[3]:

> *Given a question and answer, create one or more statements from each sentence in the given answer.*
> *question:* `[question]`
> *answer:* `[answer]`

where `[question]` and `[answer]` refer to the given question and answer. For each statement s_i in S, the LLM determines if s_i can be inferred from $c(q)$ using a verification function $v(s_i, c(q))$. This verification step is carried out using the following prompt:

> *Consider the given context and following statements, then determine whether they are supported by the information present in the context. Provide a brief explanation for each statement before arriving at the verdict (Yes/No). Provide a final verdict for each statement in order at the end in the given format. Do not deviate from the specified format.*
> *statement:* `[statement 1]`
> ...
> *statement:* `[statement` n`]`

The final faithfulness score, F, is then computed as $F = \frac{|V|}{|S|}$, where $|V|$ is the number of statements that were supported according to the LLM and $|S|$ is the total number of statements.

Answer relevance We say that the answer $a_s(q)$ is relevant if it directly addresses the question in an appropriate way. In particular, our assessment of answer relevance does not take into account factuality, but penalises cases where the answer is incomplete or where it contains redundant information. To estimate answer relevance, for the given answer $a_s(q)$, we prompt the LLM to generate n potential questions q_i based on $a_s(q)$, as follows:

> *Generate a question for the given answer.*
> *answer*: `[answer]`

We then obtain embeddings for all questions using the `text-embedding-ada-002` model, available from the OpenAI API. For each q_i, we calculate the similarity $\text{sim}(q, q_i)$ with the original question q, as the cosine between the corresponding embeddings. The answer relevance score, AR, for question q is then computed as:

$$\text{AR} = \frac{1}{n} \sum_{i=1}^{n} \text{sim}(q, q_i) \quad (1)$$

This metric evaluates how closely the generated answer aligns with the initial question or instruction.

Context relevance The context $c(q)$ is considered relevant to the extent that it exclusively contains information that is needed to answer the question. In particular, this metric aims to penalise the

[2] https://platform.openai.com

[3] To help clarify the task, we include a demonstration as part of the prompt. This demonstration is not explicitly shown in the listing of the prompts throughout this paper.

inclusion of redundant information. To estimate context relevance, given a question q and its context $c(q)$, the LLM extracts a subset of sentences, S_{ext}, from $c(q)$ that are crucial to answer q, using the following prompt:

> *Please extract relevant sentences from the provided context that can potentially help answer the following question. If no relevant sentences are found, or if you believe the question cannot be answered from the given context, return the phrase "Insufficient Information". While extracting candidate sentences you're not allowed to make any changes to sentences from given context.*

The context relevance score is then computed as:

$$\text{CR} = \frac{\text{number of extracted sentences}}{\text{total number of sentences in } c(q)} \quad (2)$$

4 The WikiEval Dataset

To evaluate the proposed framework, we ideally need examples of question-context-answer triples which are annotated with human judgments. We can then verify to what extent our metrics agree with human assessments of faithfulness, answer relevance and context relevance. Since we are not aware of any publicly available datasets that could be used for this purpose, we created a new dataset, which we refer to as *WikiEval*[4]. To construct the dataset, we first selected 50 Wikipedia pages covering events that have happened since the start of 2022[5]. In selecting these pages, we prioritised those with recent edits. For each of the 50 pages, we then asked ChatGPT to suggest a question that can be answered based on the introductory section of the page, using the following prompt:

> *Your task is to formulate a question from given context satisfying the rules given below:*
> *1. The question should be fully answered from the given context.*
> *2. The question should be framed from a part that contains non-trivial information.*
> *3. The answer should not contain any links.*
> *4. The question should be of moderate difficulty.*
> *5. The question must be reasonable and must be understood and responded to by humans.*
> *6. Do not use phrases that 'provided context', etc in the question*
> *context:*

We also used ChatGPT to answer the generated question, when given the corresponding introductory section as context, using the following prompt:

> *Answer the question using the information from the given context.*
> *question:* `[question]`
> *context:* `[context]`

All questions were annotated along the three considered quality dimensions by two annotators. Both annotators were fluent in English and were given clear instructions about the meaning of the three considered quality dimensions. For faithfulness and context relevance, the two annotators agreed in around 95% of cases. For answer relevance, they agreed in around 90% of the cases. Disagreements were resolved after a discussion between the annotators.

Faithfulness To obtain human judgements about faithfulness, we first used ChatGPT to answer the question without access to any additional context. We then asked the annotators to judge which of the two answers was the most faithful (i.e. the standard one or the one generated without context), given the question and corresponding Wikipedia page.

Answer relevance We first used ChatGPT to obtain candidate answers with lower answer relevance, using the following prompt:

> *Answer the given question in an incomplete manner.*
> *question:* `[question]`

We then asked human annotators to compare this answer, and indicate which of the two answers had the highest answer relevance.

Context relevance To measure this aspect, we first added additional sentences to the context by scraping back-links to the corresponding Wikipedia page. In this way, we were able to add information to the context that was related but less relevant for

[4] https://huggingface.co/datasets/explodinggradients/WikiEval

[5] That is, beyond the reported training cutoff of the model we used in our experiments.

	Faith.	Ans. Rel.	Cont. Rel.
RAGAs	**0.95**	**0.78**	**0.70**
GPT Score	0.72	0.52	0.63
GPT Ranking	0.54	0.40	0.52

Table 1: Agreement with human annotators in pairwise comparisons of faithfulness, answer relevance and context relevance, using the WikEval dataset (accuracy).

answering the question. For the few pages without any back-links, we instead used ChatGPT to complete the given context.

5 Experiments

Table 1 analyses the agreement between the metrics proposed in Section 3 and the human assessments from the proposed WikiEval dataset. Each WikiEval instance requires the model to compare two answers or two context fragments. We count how often the answer/context preferred by the model (i.e. with highest estimated faithfulness, answer relevance, or context relevance) coincides with the answer/context preferred by the human annotators. We report the results in terms of accuracy (i.e. the fraction of instances on which the model agrees with the annotators).

To put the results in context, we compare our proposed metrics (shown as *RAGAs* in Table 1) with two baseline methods. For the first method, shown as *GPT Score*, we ask ChatGPT to assign a score between 0 and 10 for the three quality dimensions. To this end, we use a prompt that describes the meaning of the quality metric and then asks to score the given answer/context in line with that definition. For instance, for evaluating faithfulness, we used the following prompt:

> *Faithfulness measures the information consistency of the answer against the given context. Any claims that are made in the answer that cannot be deduced from context should be penalized.*
> *Given an answer and context, assign a score for faithfulness in the range 0-10.*
> *context*: `[context]`
> *answer*: `[answer]`

Ties, where the same score is assigned by the LLM to both answer candidates, were broken randomly. The second baseline, shown as *GPT Ranking*, instead asks ChatGPT to select the preferred answer/-context. In this case, the prompt again includes a definition of the considered quality metric. For instance, for evaluating answer relevance, we used the following prompt:

> *Answer Relevancy measures the degree to which a response directly addresses and is appropriate for a given question. It penalizes the present of redundant information or incomplete answers given a question. Given an question and answer, rank each answer based on Answer Relevancy.*
> *question*: `[question]`
> *answer 1*: `[answer 1]`
> *answer 2*: `[answer 2]`

The results in Table 1 show that our proposed metrics are much closer aligned with the human judgements than the predictions from the two baselines. For faithfulness, the RAGAs prediction are in general highly accurate. For answer relevance, the agreement is lower, but this is largely due to the fact that the differences between the two candidate answers are often very subtle. We found context relevance to be the hardest quality dimension to evaluate. In particular, we observed that ChatGPT often struggles with the task of selecting the sentences from the context that are crucial, especially for longer contexts.

6 Conclusions

We have highlighted the need for automated reference-free evaluation of RAG systems. In particular, we have argued the need for an evaluation framework that can assess faithfulness (i.e. is the answer grounded in the retrieved context), answer relevance (i.e. does the answer address the question) and context relevance (i.e. is the retrieved context sufficiently focused). To support the development of such a framework, we have introduced *WikiEval*, a dataset which human judgements of these three different aspects. Finally, we have also described RAGAs, our implementation of the three considered quality aspects. This framework is easy to use and can provide deverlopers of RAG systems with valuable insights, even in the absence of any ground truth. Our evaluation on WikiEval has shown that the predictions from RAGAs are closely aligned with human predictions, especially for faithfulness and answer relevance.

References

Amos Azaria and Tom M. Mitchell. 2023. The internal state of an LLM knows when its lying. *CoRR*, abs/2304.13734.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. 2022. Improving language models by retrieving from trillions of tokens. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 2206–2240. PMLR.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023. Gptscore: Evaluate as you desire. *CoRR*, abs/2302.04166.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. 2022. Language models (mostly) know what they know. *CoRR*, abs/2207.05221.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2022. Large language models struggle to learn long-tail knowledge. *CoRR*, abs/2211.08411.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2020. Generalization through memorization: Nearest neighbor language models. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. 2022. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive NLP. *CoRR*, abs/2212.14024.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. Halueval: A large-scale hallucination evaluation benchmark for large language models. *CoRR*, abs/2305.11747.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *CoRR*, abs/2303.08896.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *CoRR*, abs/2305.14251.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *CoRR*, abs/2302.00083.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2023. REPLUG: retrieval-augmented black-box language models. *CoRR*, abs/2301.12652.

Jiaan Wang, Yunlong Liang, Fandong Meng, Haoxiang Shi, Zhixu Li, Jinan Xu, Jianfeng Qu, and Jie Zhou. 2023a. Is chatgpt a good NLG evaluator? A preliminary study. *CoRR*, abs/2303.04048.

Peiyi Wang, Lei Li, Liang Chen, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023b. Large language models are not fair evaluators. *CoRR*, abs/2305.17926.

Shufan Wang, Yixiao Song, Andrew Drozdov, Aparna Garimella, Varun Manjunatha, and Mohit Iyyer. 2023c. KNN-LM does not improve open-ended text generation. *CoRR*, abs/2305.14625.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 27263–27277.

Tianhua Zhang, Hongyin Luo, Yung-Sung Chuang, Wei Fang, Luc Gaitskell, Thomas Hartvigsen, Xixin Wu, Danny Fox, Helen Meng, and James R. Glass. 2023. Interpretable unified language checking. *CoRR*, abs/2304.03728.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 563–578, Hong Kong, China. Association for Computational Linguistics.

A Examples from WikiEval

Tables 2, 3 and 4 show examples from the WikiEval dataset, focusing in particular on answers with high and low faithfulness (Table 2), high and low answer relevance (Table 3), and high and low context relevance (Table 4).

Question	Context	Answer
Who directed the film Oppenheimer and who stars as J. Robert Oppenheimer in the film?	Oppenheimer is a 2023 biographical thriller film written and directed by Christopher Nolan. Based on the 2005 biography American Prometheus by Kai Bird and Martin J. Sherwin, the film chronicles the life of J. Robert Oppenheimer, a theoretical physicist who was pivotal in developing the first nuclear weapons as part of the Manhattan Project, and thereby ushering in the Atomic Age. Cillian Murphy stars as Oppenheimer, with Emily Blunt as Oppenheimer's wife Katherine "Kitty" Oppenheimer.	*High Faithfulness*: Christopher Nolan directed the film Oppenheimer. Cillian Murphy stars as J. Robert Oppenheimer in the film. *Low Faithfulness*: James Cameron directed the film Oppenheimer. Tom Cruise stars as J. Robert Oppenheimer in the film.

Table 2: Example from WikiEval, showing answers with high and low faithfulness.

Question	Answer
When is the scheduled launch date and time for the PSLV-C56 mission, and where will it be launched from?	*High answer relevance*: The PSLV-C56 mission is scheduled to be launched on Sunday, 30 July 2023 at 06:30 IST / 01:00 UTC. It will be launched from the Satish Dhawan Space Centre, Sriharikota, Andhra Pradesh, India. *Low answer relevance*: The scheduled launch date and time for the PSLV-C56 mission have not been provided.The PSLV-C56 mission is an important space mission for India. It aims to launch a satellite into orbit to study weather patterns.

Table 3: Example from WikiEval, showing answers with high and low answer relevance.

Question	Context
When was the Chimnabai Clock Tower completed, and who was it named after?	*High context relevance*: The Chimnabai Clock Tower, also known as the Raopura Tower, is a clock tower situated in the Raopura area of Vadodara, Gujarat, India. It was completed in 1896 and named in memory of Chimnabai I (1864–1885), a queen and the first wife of Sayajirao Gaekwad III of Baroda State. *Low context relevance*: The Chimnabai Clock Tower, also known as the Raopura Tower, is a clock tower situated in the Raopura area of Vadodara, Gujarat, India. It was completed in 1896 and named in memory of Chimnabai I (1864–1885), a queen and the first wife of Sayajirao Gaekwad III of Baroda State. It was built in Indo-Saracenic architecture style. History. Chimnabai Clock Tower was built in 1896. The tower was named after Chimnabai I (1864–1885), a queen and the first wife of Sayajirao Gaekwad III of Baroda State. It was inaugurated by Mir Kamaluddin Hussainkhan, the last Nawab of Baroda. During the rule of Gaekwad, it was a stoppage for horse drawn trams. The clock tower was erected at the cost of 25,000 (equivalent to 9.2 million or USD 120,000 in 2023).

Table 4: Example from WikiEval, showing answers with high and low context relevance.