Knowledge Grounding in Large Language Models: An Empirical Study

Anonymous Authors

Abstract—**Large language models (LLMs) have seen significant advancements in quality and adoption, yet their tendency to hallucinate remains a critical issue in applications requiring precision. Retrieval-Augmented Generation (RAG) has emerged as a promising solution to this problem, leveraging external information to enhance response accuracy. However, it remains unclear when LLMs prioritize RAG-provided context over their internal parametric knowledge, raising questions about their reliability in knowledge grounding.**

This paper investigates how LLMs respond to questions when contextual information contradicts with their parametric knowledge. We develop a diverse dataset featuring questions across a wide variety of topics, and use model-generated answers to different but similar questions to create a counterparametric answers. These are provided as part of the context of a new query, thus enabling us to determine whether the model sourced the answer to this question from the given context or its parametric memory.

We evaluate a variety architectures and sizes, including Seq2Seq encoder-decoder models `flan-t5-xl` and `flan-t5-xxl` and decoder-only models `Meta-Llama-3.1-8B-Instruct` and `Meta-Llama-3.1-70B-Instruct`, analyzing their responses to determine their reliance on RAG context versus internal knowledge. Our findings reveal that encoder-decoder Seq2Seq models tend to exhibit stronger knowledge grounding than decoder-only models. Additionally, smaller decoder-only models tend to outperform larger ones in this area.

This analysis could help develop strategies to mitigate hallucinations in RAG-augmented LLMs, ultimately improving their reliability in knowledge-intensive tasks.

I. Introduction

Large language models have become central to many NLP applications, such as question answering [1, 2], reasoning tasks [3], and code generation [4]. Despite their impressive capabilities, hallucinations continue to pose serious problems by outputting factually incorrect outputs with a tone of high confidence [2]. For tasks where precision is paramount, such as factual QA or medical and legal domains, reducing hallucinations is critical [5].

Retrieval-augmented generation (RAG) [6] aims to mitigate hallucinations by supplying relevant context from an external index. In principle, providing accurate and verifiable text at inference time should guide the model toward correct answers. However, even with the addition of a context generated by RAG, LLMs may override provided evidence with the parametric knowledge coming from their training data. This is especially common when the context contradicts the model's knowledge [7, 8].

This phenomenon relates to *knowledge grounding*: how well a model ensures its answers are based on external, verifiable sources rather than solely relying on parametric memory [6]. Recent studies show that factors such as model architecture, size, and training method influence this interplay [7, 9, 10]. Yet, it remains unclear under what conditions LLMs override their intrinsic knowledge in favor of given context.

The knowledge grounding of a model might be affected by the properties of that model, such as its architecture and size, and on the topic and context of the question. Any analysis must incorporate several models with a questions from a variety of topics.

This study contributes to a deeper understanding of knowledge grounding in large language models, offering insights for designing more reliable RAG systems. By choosing architectures that better incorporate given context, developers can reduce undesired hallucinations.

Ultimately, improving knowledge grounding is vital for building more trustworthy language models for knowledge-intensive tasks.

II. Related Work

The success of machine learning models based on transformer architecture [11] has enabled the development of large-scale language models such as GPT-3 [1] and Llama [10]. Despite their advancements, factual reliability remains a significant issue.

Recent studies such as Jiang et al. [2] highlighted the prevalence of hallucinations across tasks, particularly in factual contexts. Other studies, such as Ghader et al. [12], emphasize the challenge of ensuring accuracy in generated text.

These concerns have prompted a wave of research focused on evaluating and mitigating hallucinations. Building on this, Cheng et al. [13] systematically explores how parametric and contextual knowledge interact, identifying scenarios where contextual knowledge can degrade performance, even when complementary.

Retrieval-Augmented Generation (RAG) [6] attempts to improve factual accuracy by integrating external knowledge during inference. However RAG does not always ensure that language models prioritize the retrieved evidence over their parametric knowledge as evidenced by the research by Yu et al. and Hsia et al. [7, 8]: even when presented with contradictory context, models often rely on their inherent memory. Our study builds on these observations, examining this behavior across various model architectures and sizes.

The distinction between parametric knowledge (stored in the model's weights) and contextual knowledge (provided in the input) has been a focal point of several studies. Qinan Yu et al. [7] and Chenxi Whitehouse et al. [14] investigated how factors like training data, architecture, and fine-tuning affect the interplay between these two knowledge sources.

Through this lens, our work contributes to the understanding of how model architecture, and size shape knowledge grounding in large language models.

III. METHODS

This study investigates the behavior of large language models (LLMs) when presented with context that contradicts their parametric, learned knowledge. To achieve this, we develop a comprehensive framework for evaluating the knowledge grounding of LLMs across different architectures and model sizes.

A. Dataset Creation

1) Rationale and comparison to prior datasets: The foundation of this work is a representative dataset of questions designed to test the interplay between parametric and contextual knowledge in LLMs. This dataset must satisfy three properties:

1. Short, unambiguous answers
Questions must be constructed to elicit concise answers, enabling precise comparison and interpretation. This avoids ambiguity and minimizes variability in answers, which is critical for identifying parametric versus contextual sources.

2. Coverage of diverse topics
The dataset must span a wide range of domains, from historical events to scientific concepts, to mitigate biases inherent in training data [15]. This diversity ensures a robust evaluation of grounding across different knowledge areas.

3. Conterparametric compatibility
Questions are designed to facilitate the addition of a context allowing an answer that contradicts the parametric answer. An answer different to the parametric answer must be incorrect.

Existing datasets, such as the Natural Questions dataset [16] and the Countries' Capitals dataset [7], provided valuable insights but fell short of meeting all three criteria. For example, while the Natural Questions dataset offers a wide range of questions, its lack of systematic categorization hinders counterparametric experiments. The Countries' Capitals dataset, while well-suited for counterparametric evaluation, is limited in scope. These limitations motivated the creation of a custom dataset.

2) Dataset Design and Generation: The design of this dataset is inspired by the methodology designed by Yu et al. [7]. Several queries of the form "`What is the capital of {country}?`" are asked for a large list of countries. Later, these parametric answers from are used as counterparametric answers for questions relating to different countries.

This paper creates a similar but larger and more varied dataset of questions and answers from a wide range of topics. We can then emulate the approach used in that paper of reusing the answer from a certain question as the counterfactual context of another.

Our dataset consists of 9 different categories, each of which has a series of manually-written questions that can be answered with short and simple answers.

B. Model Selection

In order to understand the knowledge grounding of a wide variety of large language models, the queries in the dataset we previously generated are tested into models of various architectures and sizes, which are listed in Table I.

Model	Architecture	Params
`flan-t5-xl`	Encoder-Decoder	3B
`flan-t5-xxl`	Encoder-Decoder	11B
`Meta-Llama-3.1-8B-Instruct`	Decoder-Only	8B
`Meta-Llama-3.1-70B-Instruct`	Decoder-Only	70B

TABLE I
MODELS EVALUATED IN THIS STUDY.

All of the models used in this research leverage autoregressive attention using the transformer architecture [11], where each token attends to its preceding tokens, maintaining the temporal order of the sequence. This approach allows them to generate coherent and contextually relevant text by sampling from this learned distribution, while also capturing long-range dependencies and complex patterns in language.

Both Seq2Seq models are based on T5 models [17], which employ an encoder-decoder architecture: while an encoder processed the input sequence into a context vector, and an decoder generates an input sequence from this vector. The `Flan-T5` models are fine-tuned to follow instructions, and have improved zero-shot performance compared to the original T5 models [9]: `flan-t5-xl` contains approximately 3 billion parameters. This is considerably bigger than the base Flan-T5 model [9], which will generally improve the accuracy accuracy of its parametric answers; `flan-t5-xxl` contains 11 billion parameters, has higher accuracy on the parametric answers as the previous model [9].

Decoder-only models generate answers one token at a time from the input query. Given a sequence of tokens, they generate text one token at a time by attempting to solve the problem of predicting the following token [18].

This paper uses the `-Instruct` versions of the latest Llama models [19], which use this architecture and fine-tune it to tasks of instruction-following. These models are specially adept at complex prompts. Of the models used in this paper, `Meta-Llama-3.1-8B-Instruct` has 8 billion parameters, while `Meta-Llama-3.1-70B-Instruct` has 70 billion parameters.

C. Query Design

The first step to understanding the knowledge grounding of large language models is to create queries that contain data that contradicts its parametric knowledge as part of the context. By comparing the result to the existing answers it becomes possible to understand whether an answer came from the model's memory, the queries' context, or neither of these.

Initial Query	**Parametric Answer**	**Query with Counterparametric Context**
Q: What country is Cairo in? A: Cairo is in	**Egypt**	[Cairo is in **the United States**] Q: What country is Cairo in? A: Cairo is in
Q: What country is New York in? A: New York is in	**the United States**	[New York is in **Egypt**] Q: What country is New York in? A: New York is in
Q: What country is Bangkok in? A: Bangkok is in	**Thailand**	[Bangkok is in **the United States**] Q: What country is Bangkok in? A: Bangkok is in
Q: What country is San Francisco in? A: San Francisco is in	**the United States**	[San Francisco is in **Thailand**] Q: What country is San Francisco in? A: San Francisco is in
Q: What is the date of birth of Che Guevara? A: The date of birth of Che Guevara is	**June 14, 1928**	[Che Guevara was born in **245 CE**] Q: What is the date of birth of Che Guevara? A: The date of birth of Che Guevara is
Q: What is the date of birth of Emperor Diocletian? A: The date of birth of Emperor Diocletian is	**245 CE**	[Emperor Diocletian was born in **June 14, 1928**] Q: What is the date of birth of Emperor Diocletian? A: The date of birth of Emperor Diocletian is

TABLE II
EXAMPLE OF COUNTERPARAMETRIC CONTEXT BEING ADDED TO A QUERY ON CITIES. COUNTERPARAMETRIC ANSWERS ONLY GET ADDED TO QUESTIONS OF THE SAME CATEGORY. ARROWS REPRESENT THE RANDOM SHUFFLING OF PARAMETRIC ANSWERS TO ADD TO THE COUNTERPARAMETRIC CONTEXT OF A NEW QUERY. WE ENSURE THAT THE ANSWERS TO THE NEW YORK QUESTION ISN'T SHUFFLED INTO THE CONTEXT TO THE SAN FRANCISCO QUESTION AND VICEVERSA AS THEY ARE IDENTICAL.

We follow the approach by Yu et al. [7]: to test the knowledge grounding of each large language model, for every question generated with the procedure described in the previous subsection, we randomly sample an answer from the set of answers of the same base question for answers that are different to the parametric answer which is given by the original query. This ensures that this answer is different to the parametric answer to the question.

We refer to this answer as the *counterparametric answer*.

This later is later concatenated to new prompt which uses the same question to form a new query and query the same model again with the added counterparametric context. This process is exemplified in Table II.

D. Query execution and categorisation of answers

To ensure that the results are simple to interpret and minimise the effect of randomness, we follow the example of Hsia et al. [8] and use greedy decoding to generate the answer. While beam search with tends to produce more accurate results for long answers [20, 21] and there are many other sampling methods that tend to produce better results [22], this is likely to not have an effect on experiments shorter answers [17].

We compare the generated answer with the context to the previously generated parametric answer, and we categorise the answer into one of three categories depending on its equality to the possible answers.

Parametric answers are equal to the answer given by the model when queried without context. This answer would come from the parametric memory of the model, and could potentially indicate an hallucination not present in the context.

Contextual answers are equal to the context given in the query. When using a context generated by RAG, this answer would be retrieved from the index.

Other answers are neither of these, and this answer comes from a mis-interpretation of the input by the model or from some other source.

To minimise the amount of problems caused by large language models generating extra information, we truncate answers on the first period or `<EOS>` token and remove punctuation and stop words.

IV. RESULTS

A. Creating a representative dataset of questions

As described in Section III-A, we require a new and diverse dataset in order to run this data and answer the research question.

We chose 9 different categories to ensure that the questions are varied and to study how the knowledge grounding of each model changes depending on the topic. From those categories we manually wrote a set of 100 base questions and 411 objects, resulting in a total of 4760 initial queries.

1) **Person** Historical people living from early antiquity to the present day from all around the globe. The questions have short, unambiguous answers, such as date of birth or most famous invention.
2) **City** Cities from all over the globe. Questions may include population, founding date, notable landmarks, or geographical features.
3) **Principle** Scientific principles, discovered from the 16th century forward. Questions about their discovery, use, and others.
4) **Element** Elements from the periodic table. Questions may cover discovery, atomic number, chemical properties, or common uses.
5) **Book** Literary works from various genres, time periods, and cultures. Questions may involve authors, publication dates, plot summaries, or literary significance.
6) **Painting** Famous artworks from different art movements and periods. Questions may cover artists, creation dates, styles, or current locations.
7) **Historical Event** Significant occurrences that shaped world history, from ancient times to the modern era. Questions may involve dates, key figures, causes, or their historical consequences.
8) **Building** Notable structures from around the world, including ancient monuments, modern skyscrapers, and architectural wonders. Questions may cover location, architect, construction date, or architectural style.
9) **Composition** Musical works from various genres and time periods. Questions may involve composers, premiere dates, musical style, or cultural significance.

Each one of these categories has a number of questions that are assigned one of the objects, following and enhancing the question-building approach used by Yu et al. [7].

The total amount of these and composition of the 4760 questions can be found in Table III. The full set of questions can be found in the repository with the code used in this paper.

Category	Base Questions	Objects	Total Questions
Person	17	57	969
City	17	70	1190
Principle	5	37	185
Element	15	43	645
Book	11	49	539
Painting	12	44	528
Historical Event	4	64	256
Building	9	22	198
Composition	10	25	250
Total	100	411	4760

TABLE III
THE AMOUNT OF BASE QUESTIONS, OBJECTS, AND THE TOTAL AMOUNT OF QUESTIONS IN EACH CATEGORY ON THE FINAL DATASET AFTER MERGING THE BASE QUESTIONS WITH THE OBJECTS OF EACH RESPECTIVE CATEGORY.

B. Framework Results

The results of running the queries created in Section IV-A with added counterparametric context on each of the four models.

Table IV shows the total type of answers for each one of the models; this information is plotted in Figure 1.

Additionally, Table V contains the information separated by question category. How the category of each answer affects the amount of **Parametric**, **Contextual**, and **Other** answers is discussed in the next sections.

Model	P'tric	C'tual	Other
`flan-t5-xl`	248	4284	228
`flan-t5-xxl`	242	4304	214
`Meta-Llama-3.1-8B-Instruct`	745	3662	353
`Meta-Llama-3.1-70B-Instruct`	1070	3303	387

TABLE IV
AMOUNT OF ANSWERS OF EACH CATEGORY WHEN RUNNING OUR DATASET ON EACH OF THE FOUR MODELS.

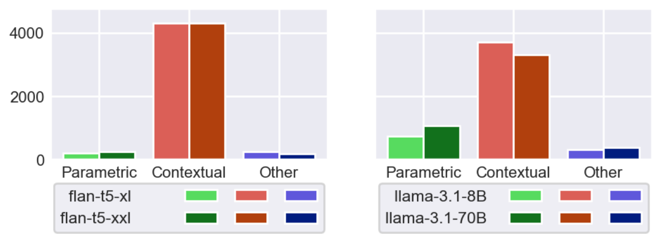

Fig. 1. Amount of each answers of each category when running a context with counterparamteric information for Seq2Seq and Decoder-only models of different sizes.

V. Discussion and Analysis

Section IV-B presented results from generating the question dataset and running the framework to understand the role of knowledge grounding in a variety of models and their parametric knowledge in question-answering. This section explains these results, and discusses what they mean for our research question.

A. Model architecture and memorised knowledge

When taking into account model architecture, the results are clear: Seq2Seq models tend to ground their knowledge from the query's context rather than from their parametric knowledge more often than Decoder-only models. These results persist across different question categories and are consistent regardless of answer types and lengths

In the framework of question-answering when using RAG to fetch contextual data from an index, Seq2Seq models might tend to have fewer hallucinations that contradict this index than Decoder-only models. We propose two hypotheses that could explain these differences.

1) Inherent Advantages of the Encoder-Decoder Architecture: Seq2Seq models such as `Flan-T5` are encoder-decoder models that process the entire context of the query in the encoder component before passing it to the decoder, which could increase the weight given to the context itself [9].

	flan-t5-xl			flan-t5-xxl			llama-8B			llama-70B		
	P'tric	C'tual	Other	P'tric	C'tual	Other	P'tric	C'tual	Other	P'tric	C'tual	Other
Person	32	900	37	23	890	56	40	833	96	209	614	146
City	120	1030	40	78	1093	19	117	1007	66	166	966	58
Principle	13	164	8	9	168	8	44	118	23	44	117	24
Element	6	637	2	102	515	28	218	385	42	275	347	23
Book	26	488	25	18	457	64	135	344	60	154	318	67
Painting	26	446	56	4	498	26	47	458	23	49	445	34
Historical Event	11	217	28	1	254	1	81	154	21	117	118	21
Building	14	174	10	0	189	9	27	163	8	31	159	8
Composition	0	228	22	7	240	3	36	200	14	25	219	6

TABLE V
RESULTS FOR EACH MODEL TESTED ON QUERIES WITH COUNTERPARAMETRIC CONTEXT IN EACH ONE OF THE 10 GIVEN CATEGORIES.

2) Different training data and fine-tuning: It's possible that these result doesn't come from the model architecture, but from the bias caused by their training methodology.

The `Flan-T5` models were trained on masked token generation and later fine-tuned on question-answering about passages [9]. This requires strong alignment between query and answer, which encourages the model to focus on the input context and makes it more likely to take the answer from the RAG-provided context.

Llama models were trained mainly on open-ended text generation, which relies more on parametric data.

It's possible that the deficiencies of knowledge grounding in Llama models might come simply to not being trained on related tasks.

B. Model size and memorised knowledge

Section IV-B also shows differences in how models of different sizes process information in queries with counterparametric context.

1) Seq2Seq Models: While the average results are very similar, which is likely due to the properties of Seq2Seq models discussed in Section V-A, there seems to be a significantly lower amount of parametric answers in the larger Flan model for the categories of *Element* and *Historical Event*. This is likely the case of the short questions answers: these categories have more questions that can be answered with answers that are 1- or 2-tokens long.

However, we can conjecture that overall the size of a Seq2Seq model has little overall impact on its knowledge grounding.

2) Decoder-only Models: Section IV-B shows a very different result for Decoder-only models. The smaller model `Meta-Llama-3.1-8B-Instruct` has better knowledge grounding than the larger model `Meta-Llama-3.1-70B-Instruct`.

We already established that decoder-only models rely on parametric knowledge to a greater degree than Seq2Seq models. Larger models have a vast internalised knowledge base accumulated from expensive training data, which can lead to increased confidence in their parametric knowledge.

It's possible that larger Decoder-only models are able to use their parametric knowledge to interpret the answer to the question in more ways that contradict the contextual knowledge. The extra information encoded on the model's weights can produce more varied evidence against the contextual answer.

With this information, we can conclude that the size of Decoder-only models has a significant effect on its knowledge grounding, and when enhancing queries with RAG it might be preferable to use a smaller model. This is consistent with similar results found for other Decoder-only models, such as Pythia and GPT-2 [7].

C. Investigating the source of *Other* answers

By manually checking the minority of answers which do not come either from the query's context nor from the model's parametric knowledge, we can understand the reason why the model chose them down to one of the following seven cases.

1. Different phrasing of a parametric answer
There are many answers where the model provides the parametric answer phrased with the format of the counterparametric context given in the query.

2. Plain incorrect answers
Sometimes, adding counterfactual context to the query causes the model to produce an incorrect answer, which is different the answers from both the parametric knowledge of the model and the given context.

3. Question misinterpretation due to the context
Some questions can be ambiguous or have a low probability of another answer. By adding a context with a counterfactual answer, the model can misinterpret the question and answer that's correct different to both the context and the parametric answer.

4. Negating the context
If the model has an answer in its parametric knowledge that contradicts the data on its context, then it interpret the context as part of the question and adds its negation as part of the answer.

5. Different phrasing of the context
Models sometimes give the same answer as provided in the query's context but in the format of the parametric answer.

6. Correct answer, just different than the parametric answer
Some questions have multiple correct answers, and adding counterfactual context can cause the model simply choose different one from its parametric memory.

7. Mixing elements of both parametric answer and context The final answer contains elements of the parametric answer combined with elements of the given. This produces an answer that's different to both the parametric and contextual answer, but with parts of both of them.

Does the architecture and size of the model affect the distribution of each type of **Other** answer? Table VI contains the amount of answers for each model.

Type	flan-t5-xl	flan-t5-xxl	llama-8B	llama-70B
(P'tric)	248	242	745	1070
(C'tual)	4284	4304	3662	3303
1.	0	0	116	234
2.	6	3	50	15
3.	0	0	13	8
4.	0	0	20	61
5.	241	170	33	38
6.	7	16	63	23
7.	6	3	17	8

TABLE VI
DIFFERENT TYPES OF OTHER ANSWERS PER MODEL, WITH AMOUNT OF PARAMETRIC AND CONTEXTUAL ADDED FOR COMPARISON.

There is a large difference in the distribution of answers that don't come either from the model or from the given context.

In the case of Seq2Seq models, the majority of **Other** answers are **Contextual** answers with different phrasing. This is consistent with the previous result, where the vast majority of their answers came from the query's context; most **Other** answers have this source.

The reasons for **Other** answers in Decoder-only models are more varied, and an interesting topic for future research.

VI. CONCLUSIONS

We presented an empirical study on knowledge grounding in LLMs, probing how models respond when provided with contradictory context. We showed that Seq2Seq architectures and smaller models better integrate new evidence, while large decoder-only models often revert to their **Parametric** knowledge. Additionally, model size does not have a large impact on Seq2Seq models, while it tends to push answers in Decoder-only models towards answers that use more **Parametric** knowledge.

Answers that are neither **Parametric** nor **Contextual** tend to have a source that follows a similar distribution for these models.

These insights can inform the selection of models and inference strategies for tasks where factual accuracy is crucial, and will be helpful on models which add a RAG to ensure that the model's parametric knowledge does not override it.

By deepening our understanding of knowledge grounding, we take a step closer to building more trustworthy and reliable language models.

REFERENCES

[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020.

[2] Z. Jiang, J. Araki, H. Ding, and G. Neubig, "How Can We Know When Language Models Know? On the Calibration of Language Models for Question Answering," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2021, pp. 1974–1991. [Online]. Available: https://aclanthology.org/2021.emnlp-main.150

[3] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," 2023. [Online]. Available: https://arxiv.org/abs/2305.10601

[4] Y. Li, D. R. Nguyen, J. Gullis, J. Li, D. Dohan, A. Shaw, B. Lakshminarayanan, H. Pham, I. Sutskever, O. Vinyals *et al.*, "Competition-level code generation with alphacode," *arXiv preprint arXiv:2203.07814*, 2022.

[5] Z. Ji, T. Yu, Y. Xu, N. Lee, E. Ishii, and P. Fung, "Towards mitigating hallucination in large language models via self-reflection," *arXiv preprint arXiv:2310.06271*, 2023.

[6] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. K"uttler, M. Lewis, W.-t. Yih, T. Rockt"aschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.

[7] Q. Yu, J. Merullo, and E. Pavlick, "Characterizing Mechanisms for Factual Recall in Language Models," 2023. [Online]. Available: https://arxiv.org/abs/2310.15910

[8] J. Hsia, A. Shaikh, Z. Wang, and G. Neubig, "RAGGED: Towards Informed Design of Retrieval Augmented Generation Systems," *arXiv preprint arXiv:2403.09040*, 2024.

[9] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, A. Castro-Ros, M. Pellat, K. Robinson, D. Valter, S. Narang, G. Mishra, A. Yu, V. Zhao, Y. Huang, A. Dai, H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei, "Scaling instruction-finetuned language models," 2022. [Online]. Available: https://arxiv.org/abs/2210.11416

[10] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "Llama: Open and efficient foundation language models," 2023. [Online]. Available: https://arxiv.org/abs/2302.13971

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17. Red Hook, NY, USA: Curran Associates Inc., 2017, p. 6000–6010.

[12] P. B. Ghader, S. Miret, and S. Reddy, "Can Retriever-Augmented Language Models Reason? The Blame Game Between the Retriever and the Language Model," 2023. [Online]. Available: https://arxiv.org/abs/2212.09146

[13] S. Cheng, L. Pan, X. Yin, X. Wang, and W. Y. Wang, "Understanding the Interplay Between Parametric and Contextual Knowledge for Large Language Models," *Preprint*, 2024, available at https://github.com/sitaocheng/Knowledge_Interplay.

[14] C. Whitehouse, E. Chamoun, and R. Aly, "Knowledge Grounding in Retrieval-Augmented LM: An Empirical Study," *arXiv preprint*, 2023.

[15] P. Beytía, "The positioning matters. estimating geographical bias in the multilingual record of biographies on wikipedia," *SSRN Electronic Journal*, 03 2020.

[16] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, M. Kelcey, J. Devlin, K. Lee, K. N. Toutanova, L. Jones, M.-W. Chang, A. Dai, J. Uszkoreit, Q. Le, and S. Petrov, "Natural Questions: a Benchmark for Question Answering Research," *Transactions of the Association of Computational Linguistics*, 2019.

[17] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," *Journal of Machine Learning Research*, vol. 21, pp. 1–67, 2020.

[18] A. Radford and K. Narasimhan, "Improving language understanding by generative pre-training," 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:49313245

[19] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Sravankumar, A. Korenev, A. Hinsvark, A. Rao, A. Zhang, A. Rodriguez, A. Gregerson, A. Spataru, B. Roziere, B. Biron,

B. Tang, and B. C. et al., “The Llama 3 Herd of Models,” 2024. [Online]. Available: https://arxiv.org/abs/2407.21783
[20] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to Sequence Learning with Neural Networks,” 2014. [Online]. Available: https://arxiv.org/abs/1409.3215
[21] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, Łukasz Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean, “Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation,” 2016. [Online]. Available: https://arxiv.org/abs/1609.08144
[22] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, “The curious case of neural text degeneration,” *arXiv preprint arXiv:1904.09751*, 2020.