Augmented Retrieval and Citation Integration using arXiv and GPT-4 Enhancements (ARCITE)

Dongheng Li, Zongxian Feng, Hang Yu, Qiannian Liang, Mohammad Shoaib Abbas
University of Illinois at Urbana-Champaign
{dl29, zfeng13, hangy6, ql17, mabbas22}@illinois.edu

Abstract

In the realm of academic writing, ensuring the accuracy and relevance of citations is a critical challenge, particularly when utilizing automated tools. This paper introduces ARCITE, a groundbreaking solution that integrates the advanced capabilities of GPT-4 with the extensive resources of the arXiv database to tackle the prevalent issue of citation hallucination in automated citation generation. By implementing a sophisticated approach to identify cite-worthy content and efficiently retrieve pertinent articles, ARCITE significantly outperforms GPT-4 in terms of citation precision and normalized entailment score. Our experimental results showcase ARCITE's enhanced reliability and contextual accuracy in generating citations, making it a valuable asset for researchers and academicians. We also made our code available on Github[1]. Our demo is available on Youtube[2].

1 Introduction

As Large Language Models (LLMs) like OpenAI started to gain popularity in the industry of academic writing and citation, related opportunities and challenges have also arisen. One such challenge has successfully caught our attention - the citation "hallucination". Specifically, we have a concern for the accuracy and credibility of automated citation generation. When using some of the most popular LLMs, such as GPT-4, to automatically generate citations for a specific term or text, we have noticed that the generated citations, while seemingly relevant, actually lack an authentic foundation or proper source context. Aiming to preserve the integrity and authority of academic works, we decided to implement approaches to address this hallucination issue and to improve the overall quality of automated citation generation.

Based on the testing, most of the existing LLMs have limitations to some extent, which may be related to their susceptibility to citation hallucination. This discovery further underscores the necessity of a comprehensive and innovative solution.

Motivated to refine the accuracy of the generated citations, we developed a system to call the GPT API to get the identification for citations. After acquiring the identification, we divided the provided input into two categories: term and text. Based on different categories, the system would generate queries for either term or Semantic. With the results from such queries, an API call would be made to a scholar database, like arXiv, to retrieve the academic paper that has a strong connection with the input. After a successful retrieval, the system would then return some of the best matches to the users as the output. Below is a graph showing how our system works:

As shown in the graph, our approach combines the text analysis techniques from GPT-4, integrates the citation retrieval methods with the arXiv and Semantic Scholar API, and includes code logic to enhance the relevance and citation matching features. The innovative techniques employed by our approach have provided our LLM with the ability to conduct citation exploration from multiple disciplines.

When compared with the result from the GPT-4 model, the citation generated by our model has demonstrated a great improvement in reliability and credibility. Since the GPT-4 model relies on a large number of datasets, issues like "Overreliance on Patterns" may arise, which indicates that if the model has been trained on large datasets without specific guidelines on generating accurate citations, it might learn to generate citations based on patterns it observed in the training data, rather than ensuring logical entailment. Thus, a fair comparison between our model and the GPT-4 model could be a necessary evaluation of the models' perfor-

[1] https://github.com/Carstin520/CS546-ARCITE
[2] https://youtu.be/N4T7YIt6N2g

mance.

As an overview of the results, we have noticed a significant increase in the number of ground truth references generated by our model. Based on our precision test and normalized entailment score test, the ability of our model to generate creditable citations has demonstrated a huge improvement over most of the existing automated citation generation frameworks.

As we delve deeper into the subsequent sections, more detailed information, like our specific Python functions and LLM components, will be provided. The reader will also gain insights into how our methodology contributes to resolving the concerns of citation accuracy in the context of LLMs.

2 Related Work

The emergence of LLMs as tools for augmenting scholarly writing has been notable, albeit fraught with challenges in citation accuracy. Gao et al., 2023a underscored the susceptibility of LLMs to generate "hallucinated" citations, drawing attention to the potential for LLMs to produce relevant yet sometimes baseless references when the source material is absent from their training data. This issue is critical for maintaining the integrity and utility of scholarly work, motivating our project to refine the process of citation generation using LLMs. We can summarize the related work under this topic into two categories based on Huang et al., 2021:

Pre-hoc citation: This proactive strategy requires the LLM to determine the need for a citation before generating content. When a citation need is identified, an information retrieval (IR) system is triggered to find the necessary data, which the LLM then includes as citations in its response. This method aligns with research on enhancing language models with retrieval capabilities, as indicated by Guu et al., 2020; Lewis et al., 2020; Wang et al., 2023

Post-hoc citation: This reactive approach has the LLM first generate a response and then review the content to decide if citations are needed. If required, an IR system finds and adds the relevant citations to the already generated text. This strategy is supported by research focused on attribution requirements in LLM outputs (Rashkin et al., 2023; Gao et al., 2022, 2023a).

Consequently, our study naturally falls into the second category, post-hoc citation.

The method by Huang et al., 2021 focuses on automated citation recommendation in the legal domain (Domain-Specific), using machine learning techniques such as collaborative filtering, BiLSTM, and RoBERTa. These models analyze the textual context within legal documents to predict relevant citations, emphasizing the understanding of legal texts' intricate context for appropriate citation suggestions.

In contrast, our method diverges significantly by emphasizing semantic analysis and term-specific searches. While the approach by Huang et al., 2021 is context-driven, leveraging machine learning for citation prediction, our method identifies citation-worthy content through semantic matching of domain-specific and non-original ideas. Our approach involves direct content analysis and determining the semantic relevance of citations, showcasing a distinct methodology in the realm of automated citation recommendation.

3 Dataset

We use the Multi-XScience dataset [1], a large-scale dataset for extreme multi-document summarization of scientific articles. The dataset consists of 30,369, 5,066, and 5,093 samples for the train, validation, and test split respectively[2]. Each sample contains the abstract of a query paper and the abstracts of the papers it references, as well as the related work section of the query paper as the summary. The average document length is 778.08 words, and the average summary length is 116.44 words. The dataset covers various scientific domains, such as computer science, physics, mathematics, and biology.

The dataset is formatted in JSON, with the following structure:

```
{
 "aid": string,
 "mid": string,
 "abstract": string,
 "related_work": string,
 "ref_abstract": {
   "@cite_N": {
     "mid": string,
     "abstract": string
   },
   ...
 }
}
```

The related work section of the query paper includes background knowledge or recent studies of the topic, with in-text citations to the reference papers. The in-text citations are standardized as `@cite_N`, where `N` is the index of the cited paper in the reference list. The reference papers are stored in the `ref_abstract` field, with their corresponding citation labels, Microsoft Academic Graph (MAG) ids, and abstracts.

To access the original paper information, such as title, DOI, and publishing date, we use the OpenAlex API[3], which is a successor of the Microsoft Academic Graph (MAG)[4]. The OpenAlex id is obtained by prepending a `W` to the MAG id.

To use this dataset in our project, we performed some preprocessing steps, such as:

- Filtering out samples that had missing or incomplete fields, such as empty abstracts or related work sections.
- Removing duplicated papers that appeared in multiple samples.
- Excluding papers that were published before 2010 or had no proper title or in-text citations.

After preprocessing, we obtained a clean and consistent dataset that is suitable for our research purposes. The preprocessing was applied to the `test.json` file, which initially contained 5,093 samples but was reduced to 246 high-quality samples after filtering. Since our methods do not require any supervised training, the evaluation on 246 samples is sufficient to demonstrate their performance. In particular, we can calculate the precision of the predicted citation positions by comparing them with the ground truth positions of the `@cite_N` tags in the related work section. We can also determine the precision of the paper identification by comparing the paper name and DOI of the predicted citation with the actual citation.

4 Methodology

4.1 Generation Methods

In this section, we describe the architecture of our proposed system, ARCITE, which integrates the identification of citation-worthy content with scholarly database retrieval. Figure 2 provides an overview of the workflow.

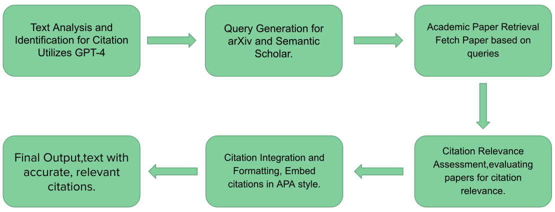

Text Analysis and Identification for Citation Utilizes GPT-4
Query Generation for arXiv and Semantic Scholar.
Academic Paper Retrieval Fetch Paper based on queries
Citation Relevance Assessment,evaluating papers for citation relevance.
Citation Integration and Formatting, Embed citations in APA style.
Final Output,text with accurate, relevant citations.

Figure 1: The ARCITE system overview. The process begins with text analysis using GPT-4 and ends with the final output text, which includes accurate and relevant citations formatted in APA style. The workflow emphasizes citation relevance assessment to ensure the quality of the citations.

4.1.1 Citeworthy Content Identification

Our approach for the initial step in the citation process involves leveraging the powerful language capabilities of GPT-4. The function `analyze_text_for_citations` has been designed to guide GPT-4 in identifying crucial components within the given text that necessitate citations.

Incorporating clear instructions ensures GPT-4 better focuses upon distinct terminologies and sentences in turn providing a granular analysis. The same techniques are also used to consider acronyms. Clear explanations are recognized as being critical to enhance clarity.

The output will consist of the input message that has been altered with the addition of index anchors appropriately placed; the index anchors will appear regarding whether the text was identified as a 'Term' or 'Sentence.' Also, A section of the output will contain an assessment of what citation is required, the explanation allowing us to understand the thought process-production chain.

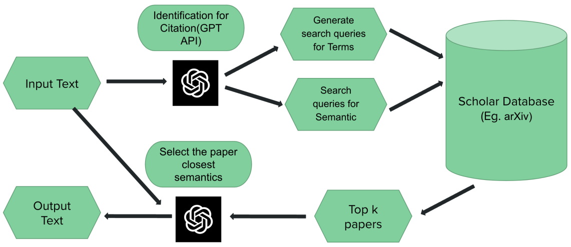

Identification for Citation(GPT API)
Input Text
Generate search queries for Terms
Search queries for Semantic
Scholar Database (Eg. arXiv)
Select the paper closest semantics
Output Text
Top k papers

Figure 2: The ARCITE system architecture. The workflow begins with the input text and ends with the output text, integrating various components such as the GPT API for citation identification and query generation, and the scholarly database for paper retrieval.

Instructions to GPT-4: The function constructs a message for GPT-4, prompting it to:

- Identify terminologies and sentences requiring citations.
- Treat each term separately and avoid mixing different terms.
- Include explanations for acronyms.
- Assign unique indices to each identified term or sentence, distinguishing between 'Term' and 'Sentence' (e.g., [Term 0], [Sent 0]).
- For each identified term or sentence, provide a brief explanation of why a citation is necessary to validate or support the information or claim.

The identification of content to cite is an essential first step which is intellectually driven and the consequence of a structured understanding of any domain topic. The subsequent sections will add depth to this analysis with a specific focus on any areas of weakness for automatic creation adding strength to the overall process of automating the citation of content.

4.1.2 Search Query Generation

After we have obtained the analysis result in Step 1, we will proceed to Step 2 to generate search queries tailored for the arXiv repository. The purpose of this step is to create accurate queries that fit the subjects mentioned in the text. This is to make the retrieval of relevant scholarly articles more effective.

The GPT-4 prompt instructs the model to prioritize aligning the arXiv.search categories and keywords for the subjects mentioned in the response. The response is then structured as a dictionary, where each term or sentence with an index to arXiv.search method. The response is parsed with a function with robust logic that can handle unexpected formats with ease. varargin of the arXiv.search method consists of properly constructed search queries with arXiv.search categories and correlating and associated keywords to obtain a comprehensive and precise search functionality.

Instructions to GPT-4: The function constructs a prompt for GPT-4, instructing it to:

- Analyze the provided text and determine the appropriate arXiv.search categories for citations.
- Specify arXiv.search categories and keywords relevant to the subjects mentioned in the text.
- For each term or sentence requiring a citation, list the corresponding arXiv.search categories (e.g., cs.AI, cs.CC, eess.AS).
- Also, list the keywords associated with each term or sentence requiring a citation.

4.1.3 arXiv Search and Retrieval

Building upon the meticulously generated search queries from Step 2, the third step in our methodology involves the execution of arXiv searches using the `arxiv` library. Instructing the library to use the specified search queries, the function retrieves a predetermined number of results (5 in our demo) for each query. The sorting criterion is set to relevance, ensuring that the most pertinent scholarly articles are prioritized in the results.

The output is organized into a dictionary, where each key corresponds to a term or sentence index, and the associated value is a list of scholarly articles retrieved from the arXiv repository based on the respective search query.

4.1.4 Filter Retrieved Papers and Analyze Content

We refine the retrieved scholarly papers to identify the most relevant support for each term or sentence identified in the initial analysis. The function `find_citations` orchestrates this process, leveraging the capabilities of GPT-4 to analyze content and filter papers. The prompt constructed for GPT-4 emphasizes the need to focus on abstracts for terms and measure semantic similarity for sentences. The desired output format includes details such as title, PDF URL, authors, and published date, ensuring consistency in the presentation of results.

This step ensures that the filtered papers provide the best support for each cite-worthy term or sentence, setting the stage for the final integration and citation steps in our automated system.

Instructions to GPT-4: The function constructs a prompt for GPT-4, instructing it to:

- For each term or sentence identified in the analysis result, find the paper from the retrieved set that provides the best support.
- Focus on finding the term in the provided abstract for terms and measure semantic simi-

larity between the sentence and the provided documents for sentences.

- Maintain a consistent format for the results, including title, PDF URL, authors, and published date.

4.1.5 Generate Citations in Required Style

In the concluding step of our automated citation system, the `match_style` function utilizes the GPT-3.5 Turbo model to generate citations in a specified style for identified cite-worthy content. This crucial step aims to seamlessly integrate the outcomes from prior analysis into well-formatted in-text citations and a reference list, ensuring adherence to the specified citation style.

The function prepares informative messages for GPT-3.5 Turbo, including the original text, Step 1 results, and details about selected papers, guiding the model to produce accurate and professionally formatted citations.

The output from GPT-3.5 Turbo encapsulates completed in-text citations and a reference list, representing the final polished product of our automated system. This process not only enhances the document's academic rigor but also streamlines the presentation of citations, providing a cohesive and standardized scholarly output.

Instructions to GPT-3.5 Turbo: The function constructs a series of messages for GPT-3.5 Turbo, instructing it to:

- Provide the original text, Step 1 result, and selected paper information.
- Generate in-text citations in the specified citation style for the corresponding terms or sentences in the Step 1 result.
- Include a request to generate a reference list.

4.2 Evaluation Method

4.2.1 Precision of In-Text Citation Position

Our evaluation focuses on assessing the precision of predicted in-text citation positions by comparing them with ground truth positions of @cite_N tags in the related_work section. Precision is calculated using the formula:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

True Positive (TP): The number of correctly predicted in-text citation positions.

False Positive (FP): The number of inaccurately predicted in-text citation positions.

4.2.2 Entailment Score of Matched Paper

The entailment score evaluates the quality of the entailment relationship between the abstract of the matched paper and the cited content. We leverage the BERT-NLI model by Laurer et al. to compute the entailment score. The formula for normalization, treating the ground truth as 100%, is given by:

$$Entailment = \frac{1}{1 + e^{-BERT-NLI\ Score}}$$

This normalization ensures that the entailment score is presented on a consistent scale, aligning it with the ground truth.

4.2.3 Evaluation Procedure

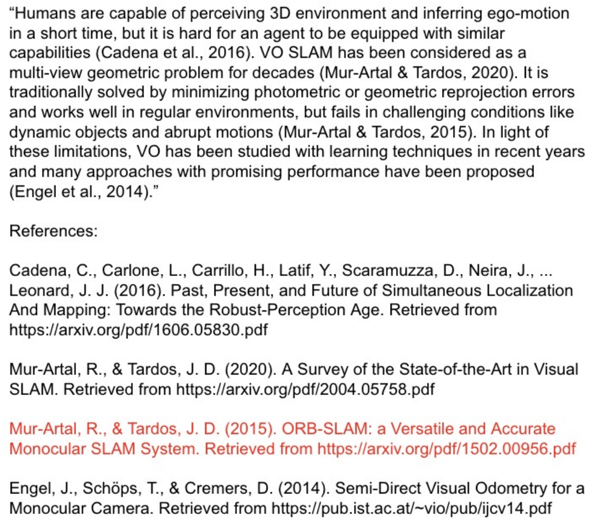
One related previous work of ours is Gao et al.(Gao et al., 2023b). In their evaluation, entailment score of the passage as the premise to the content of the cited paper as the hypothesis is used as a scorer to decide whether the passage entails the cited paper's content or not. Positive citations are marked if the entailment score is large enough and the precision metric is defined for the ratio of the true positives divided by all the citations generated. In our study, we have a new interpretation of the entailment score and it is directly used as one of the evaluation metrics under our context.

1. Data Preparation: Utilize the Multi-XScience dataset, incorporating the related work section that includes standardized in-text citations marked as @cite_N tags.

2. In-Text Citation Position Evaluation: Evaluate the model's precision in predicting the correct positions of in-text citations by comparing them with ground truth positions. Calculate precision using the provided formula.

3. Entailment Score Evaluation: Employ the BERT-NLI model by Laurer et al. to compute the entailment score between the abstract of the matched paper and the cited content. Normalize the entailment score using the provided formula, treating the ground truth as 100

4. Performance Assessment: Analyze the precision results to gauge the accuracy of in-text citation position predictions. Examine the normalized entailment scores to understand the quality of the entailment relationship between the matched paper and the cited content.



“Humans are capable of perceiving 3D environment and inferring ego-motion in a short time, but it is hard for an agent to be equipped with similar capabilities (Cadena et al., 2016). VO SLAM has been considered as a multi-view geometric problem for decades (Mur-Artal & Tardos, 2020). It is traditionally solved by minimizing photometric or geometric reprojection errors and works well in regular environments, but fails in challenging conditions like dynamic objects and abrupt motions (Mur-Artal & Tardos, 2015). In light of these limitations, VO has been studied with learning techniques in recent years and many approaches with promising performance have been proposed (Engel et al., 2014).”

References:

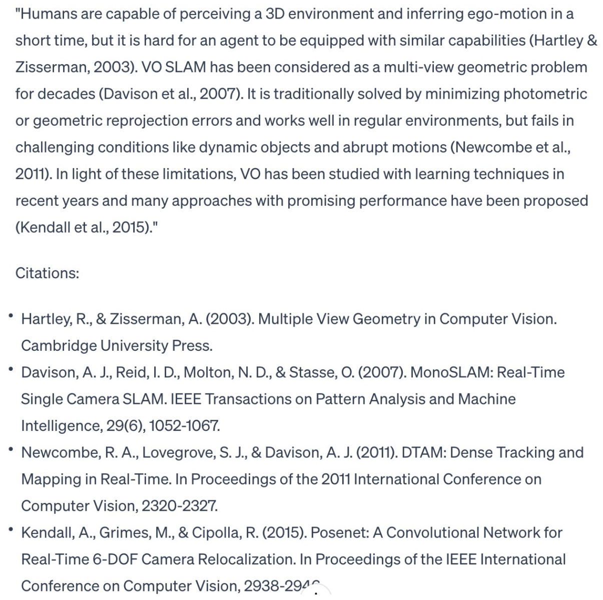
Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., ... Leonard, J. J. (2016). Past, Present, and Future of Simultaneous Localization And Mapping: Towards the Robust-Perception Age. Retrieved from https://arxiv.org/pdf/1606.05830.pdf

Mur-Artal, R., & Tardos, J. D. (2020). A Survey of the State-of-the-Art in Visual SLAM. Retrieved from https://arxiv.org/pdf/2004.05758.pdf

Mur-Artal, R., & Tardos, J. D. (2015). ORB-SLAM: a Versatile and Accurate Monocular SLAM System. Retrieved from https://arxiv.org/pdf/1502.00956.pdf

Engel, J., Schöps, T., & Cremers, D. (2014). Semi-Direct Visual Odometry for a Monocular Camera. Retrieved from https://pub.ist.ac.at/~vio/pub/ijcv14.pdf

Figure 3: Example of citations generated by GPT-4. As shown in the figure, there is one overlap between our model auto-generated citations and the ground truth citations. In addition, many of the citations are plausible.



"Humans are capable of perceiving a 3D environment and inferring ego-motion in a short time, but it is hard for an agent to be equipped with similar capabilities (Hartley & Zisserman, 2003). VO SLAM has been considered as a multi-view geometric problem for decades (Davison et al., 2007). It is traditionally solved by minimizing photometric or geometric reprojection errors and works well in regular environments, but fails in challenging conditions like dynamic objects and abrupt motions (Newcombe et al., 2011). In light of these limitations, VO has been studied with learning techniques in recent years and many approaches with promising performance have been proposed (Kendall et al., 2015)."

Citations:

- Hartley, R., & Zisserman, A. (2003). Multiple View Geometry in Computer Vision. Cambridge University Press.
- Davison, A. J., Reid, I. D., Molton, N. D., & Stasse, O. (2007). MonoSLAM: Real-Time Single Camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(6), 1052-1067.
- Newcombe, R. A., Lovegrove, S. J., & Davison, A. J. (2011). DTAM: Dense Tracking and Mapping in Real-Time. In Proceedings of the 2011 International Conference on Computer Vision, 2320-2327.
- Kendall, A., Grimes, M., & Cipolla, R. (2015). Posenet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization. In Proceedings of the IEEE International Conference on Computer Vision, 2938-294

Figure 4: Example of citations generated by our model. As shown in the figure, there is no overlap between the GPT-4 auto-generated citations and the ground truth citations, although some of the citations are plausible.

5 Experiments

In our study, we applied our novel automated citation method to our repurposed Multi-XScience dataset. Our problem formulation is unique, and there is no existing related work that precisely matches it. The primary motivation behind our approach was to address the issue of hallucination in citations generated by ChatGPT when producing citations end-to-end. Consequently, our natural baseline for evaluation is to compare our model with ChatGPT, specifically GPT-4, on our Multi-XScience dataset.

To assess the performance of our model and GPT-4, we measured the precision of in-text citation positions and the normalized entailment score, as detailed in the "Evaluation Method" section. The results are presented in the table below:

	Precision (%)	Entailment Score (%)
GPT-4	0	4.04
Ours	**16**	**44.5**
GT	100	100

Table 1: Results of precision of the in-text citation position and the normalized entailment score of the citations generated by GPT-4 and our model. "GT" stands for the ground truth

5.1 Precision of the In-Text Citation Position

In the "Precision" column, it is evident that GPT-4 achieved a precision of 0, indicating that it consistently fails to provide an exact match for citations compared to the ground truth text segments. In contrast, our model, while still relatively low in precision, managed to generate some citations that closely matched the ground truth citation, as indicated by Figure 2 (marked in red). This suggests that our approach improves the citation matching ability compared to the GPT-4 model. Notably, the low precision scores for both models stem from the fact that both tend to produce numerous reasonable citations based on context, while the ground truth contains fewer citations. Some citations might be omitted because of sufficient context or the use of common terminology and sentences in that research area. Therefore, the precision metric may not be perfect for evaluating citation generation quality.

5.2 Normalized Entailment Score

In the "Normalized Entailment Score" column, our model significantly outperforms GPT-4. This suggests that the papers cited by our model closely resemble the content of the ground truth citations compared to those generated by GPT-4. This is a strong indicator that our model captures context more effectively and suggests more relevant citations for positions requiring references. Moreover, this metric compensates for the shortcomings of the precision metric.

6 Conclusion

In conclusion, our experiments demonstrate that our proposed method surpasses the GPT-4 model in terms of auto-citation precision and normalized entailment score. This indicates that our model excels in the task of generating citations automatically. Our work introduces a novel NLP task enabled by large language models. Many intriguing aspects in this direction remain unexplored, such as the development of more comprehensive evaluation metrics and addressing issues related to controlling the number of generated citations, as mentioned in the "Experiment" section. These areas provide fertile ground for future research extensions.

References

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, et al. 2022. Rarr: Researching and revising what language models say, using language models. *ArXiv preprint, abs/2210.08726*.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023a. Enabling large language models to generate text with citations. *arXiv preprint arXiv:2305.14627*.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023b. Enabling large language models to generate text with citations. *arXiv preprint arXiv:2305.14627*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020*, pages 3929–3938. PMLR.

Z. Huang, C. Low, M. Teng, H. Zhang, D. E. Ho, M. S. Krass, and M. Grabmair. 2021. Context-aware legal citation recommendation using deep learning. pages 79–88.

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*.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. 2023. Measuring attribution in natural language generation models. *Computational Linguistics*, pages 1–66.

Boxin Wang, Wei Ping, Peng Xu, Lawrence McAfee, Zihan Liu, Mohammad Shoeybi, Yi Dong, Oleksii Kuchaiev, Bo Li, Chaowei Xiao, et al. 2023. Shall we pretrain autoregressive language models with retrieval? a comprehensive study. *ArXiv preprint, abs/2304.06762*.