

Application of Large Language Models and ReAct Prompting in Policy Evidence Collection

Yang Zhang^a and James Pope^b

Intelligent Systems Lab, University of Bristol, Bristol, U.K.

Keywords: Natural Language Processing, Large Language Model, Prompting Engineering, Evidence-Based Policy-Making, Policy Analysis.

Abstract: Policy analysis or formulation often requires evidence-based support to ensure the scientific rigor and rationality of the policy, increase public trust, and reduce risks and uncertainties. However, manually collecting policy-related evidence is a time-consuming and tedious process, making some automated collection methods necessary. This paper presents a novel approach for automating policy evidence collection through large language models (LLMs) combined with Reasoning and Acting (ReAct) prompting. The advantages of our approach lie in its minimal data requirements, while ReAct prompting enables the LLM to call external tools, such as search engines, ensuring real-time evidence collection. Since this is a novel problem without existing methods for comparison, we relied on human experts for ground truth and baseline comparison. In 50 experiments, our method successfully collected correct policy evidence 36 times using GPT-3.5. Furthermore, with more advanced models such as GPT-4o, the improved understanding of prompts and context enhances our method's efficiency. Finally, our method using GPT-4o successfully gathered correct evidence 45 times in 50 experiments. Our results demonstrate that, using our method, policy researchers can effectively gather evidence to support policy-making.

1 INTRODUCTION

Policy analysis refers to the assessment and research conducted to improve economic and social public issues and the formulation of policy principles or guidelines. Policy analysis can provide strong support for policy decision-making, and this process often involves extensive evidence research. Evidence-based research provides reliable data and analytical support for policy-making, helping policymakers make more scientific and rational decisions based on actual evidence and enhancing citizens' trust in policies (Françoise et al., 2022).

This study is conducted in collaboration with PolicyBristol (University of Bristol, 2024), a policy research organization from the University of Bristol. Similarly, in order to support evidence-based policy-making, PolicyBristol aims to identify relevant policy content issued by various types of official organizations or institutions when exploring a particular policy, to provide evidence for subsequent policy analysis.

For example, when considering housing subsidy policy, London City Council is an official organization that would launch relevant policy. The purpose of this study is to identify the policy content issued by different types of organizations or institutions as evidence to support a given policy issue.

However, the policy research discussed above often involves evidence gathering, which can be a slow and tedious process. Collecting relevant policy documents from various organizations requires extensive time and effort, as it involves searching through numerous sources, filtering through the information, and ensuring the accuracy and relevance of the evidence. Some automated methods are needed.

Policy analysis often involves a large amount of textual content, so natural language processing (NLP) techniques are frequently used as tools for automating policy analysis. Previously, SBERT (Reimers and Gurevych, 2019) is used for clustering casual relationship between policies (Hooper et al., 2023). BERT (Devlin et al., 2019) is also used and achieves best F1-Score on answer sentences selection on questions about privacy policy content (Ravichander et al., 2019). This demonstrates that NLP techniques can

^a <https://orcid.org/0009-0003-0482-4867>

^b <https://orcid.org/0000-0003-2656-363X>

significantly enhance policy analysis by improving language understanding. However, these methods often require access to extensive datasets to train effectively for learning complex patterns and extracting meaningful insights. Additionally, the evidence support required for policy analysis is often the latest and real-time, which can cope with dynamically changing social problems, respond quickly to environmental changes, and maintain effectiveness. The above methods also cannot guarantee timeliness, as they rely entirely on the training dataset.

Large language model (LLM) has already demonstrated strong generalization capabilities on Few-shot (FS) (Brown et al., 2020). Therefore, this paper proposes a novel LLM-based evidence collection approach. The core idea is to use Reasoning and Acting (ReAct) prompting (Yao et al., 2023) to guide the LLM, allowing it to choose to utilize tools such as search engines, ensuring the timeliness of the results. Therefore, our study makes the following two main contributions:

- We propose a fully automated policy collection method that policy researchers can use to gather evidence for policy analysis, providing a basis for decision-making and significantly improving the efficiency of policy analysis.
- We are the first to propose using LLMs combined with prompt engineering for policy collection. Using LLMs eliminates the need for large amounts of high-quality data and extensive training time. The ReAct framework can integrate external tools, such as search engines, ensuring the factual accuracy and high reliability of the evidence gathered, which show our method are both effective and easy to implement.

2 RELATED WORK

2.1 Training Process of LLMs

Currently, large language models are typically pre-trained on vast amounts of textual data to develop a foundational language understanding. Subsequently, they undergo supervised fine-tuning (SFT) using conversational-format data to further enhance their ability to understand instructions and engage in dialogue. Finally, to align the model's responses with human preferences, reinforcement learning (Ouyang et al., 2022) or non-reinforcement learning methods (Rafailov et al., 2023) are used for further training. Therefore, LLMs have demonstrated powerful

instruction comprehension and generation capabilities.

2.2 Challenges in Applying LLMs

Using an LLM alone, without external tools, may encounter the following issues:

- **Timeliness Problem.** The content generated by an LLM largely depends on its training data. The date range of this data constrains the model's reasoning abilities. For instance, GPT-3.5 by OpenAI¹ is trained on data up until September 2021, leaving the model unaware of events beyond that point, and therefore unable to assist users with information on more recent developments.
- **Hallucination Issue.** Hallucination issue refers to LLMs attempting to fabricate answers that appear correct (Lin et al., 2022). The hallucination problem in LLMs is caused by factors such as low data quality, biases, and outdated information. (Lee et al., 2022; Narayanan Venkit et al., 2023; Onoe et al., 2022). For the timeliness issue, an LLM might respond with "I am unable to help". However, the hallucination problem can lead the LLM to deceive, confuse, or mislead users, resulting in more serious consequences.
- **Insufficient Model Capability.** LLMs might not perform well on tasks that involve reasoning or complex, multi-step processes (Wei et al., 2022), as model initially lacks the ability to generate intermediate reasoning steps. Instead, it tends to produce the final answers directly, which often results in low-quality responses.

In this study on policy evidence collection based on LLMs, using only a single LLM to generate evidence may lead to timeliness and hallucination issues, such as producing outdated information or fabricating content. On top of that, evidence collection involves multiple steps, such as first identifying official organizations related to the policy problem and then generating relevant policy content issued by these organizations. In such cases, LLMs may also face challenges related to insufficient capability.

2.3 Mitigation Strategies

To address the timeliness and hallucination issues in LLMs, retrieval-augmented techniques are commonly used to supplement the model's knowledge (Lewis et al., 2020). This approach involves preparing knowledge data in advance to mitigate the LLM's

¹<https://platform.openai.com/docs/models>

knowledge gaps, making it well-suited for applications within specific, specialized domains. To address the issue of insufficient capability in LLMs, some previous work has focused on training language models with datasets that include rationale processes, enabling the model to develop reasoning abilities (Ling et al., 2017; Cobbe et al., 2021).

However, the policy issues explored in this study span multiple domains, introducing an element of uncertainty to when using retrieval-augmented techniques. Furthermore, above methods all require large amounts of high-quality data, which increases task complexity.

To reduce dependency on data and to enhance the model's problem-solving capabilities, prompt engineering is used in this study. By designing the LLM's prompts (input to LLMs) in natural language, guiding LLM to generate specified content. For instance, including examples of reasoning steps in the prompts allows the model to imitate these examples, stimulating its reasoning ability and addressing the issue of insufficient capability (Wei et al., 2022). To address issues of timeliness and hallucination, we applied the Reasoning and Acting (ReAct) prompting approach, which not only enables the model to generate reasoning steps but also allows it to choose actions based on different reasoning thoughts (Yao et al., 2023). One of these actions can correspond using an external tool, such as a search engine, where the search results provide the LLM with real-time knowledge updates.

3 METHODOLOGY

For our methodology, key idea is to combine FS learning (Brown et al., 2020) with the ReAct framework (Yao et al., 2023). In our approach, the prompt given to the LLM includes a task description, available actions, and examples of how to collect policy evidence, containing both the reasoning and action-execution processes. Figure 1 shows our prompt design. Blue part is task description and instructions, orange part is example provided, for simplicity of demonstration, Figure 1 only includes one example, yellow part is new task that are about to be addressed where new policy issue and type of official organization are typed in. Each available action corresponds to an external tool. Observation represents result returned by each action. It is important to note that we aim for the collected evidence to be authoritative. Therefore, we search for policy content in official websites. The steps to complete the task involve first locating the official homepage and then, based on that, searching for relevant sub-pages including policy evidence.



Figure 1: ReAct prompt design.

3.1 Implementation of Each Action

- **Generate Action.** When given a policy issue and a particular type, the first step should be to generate a specific official organization that involves the policy issue and fits the type. The approach is to use **another** LLM with simple FS learning to have the LLM generate an organization based on the provided exemplifications. The LLM used is GPT3.5 by OpenAI. Figure 2 shows the design of our FS prompts, where the model is given several exemplifications in the yellow section so that the LLM understands the context and generates organizations for new policies and types (new policy and type can be inserted into green section).
- **Search Action.** This action is to search for the official homepage of specific organization. This step is necessary because we want the final policy evidence to come from the official website of this organization. If we don't know what the official homepage of this organization is, we have no basis when searching for websites containing

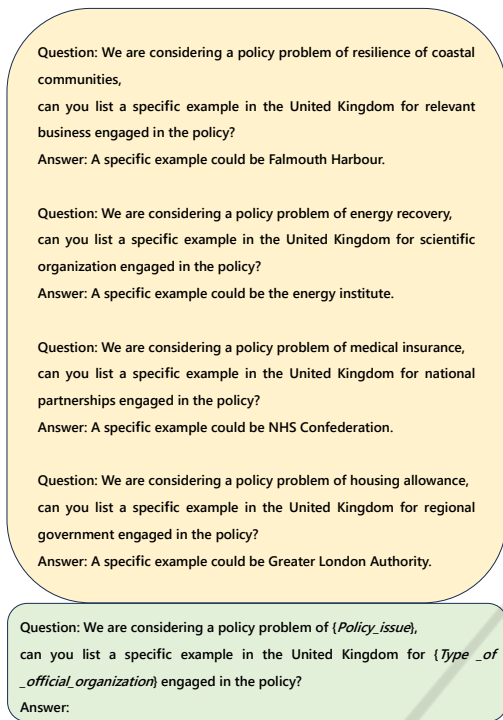


Figure 2: Few shot prompt design for **Generate action**.

policy evidence. Google search engine Application Programming Interface (API) is used with the search keyword: “official website of $\{specific\ organization\}$ ”. The official website often appears as the first search result. However, there are instances where the official website might be in the second result. To address this issue, we extract the URL information from the top two search results and input them into **another** GPT3.5 model, which determines which URL is the official website empirically. Figure 3 shows our prompt design.

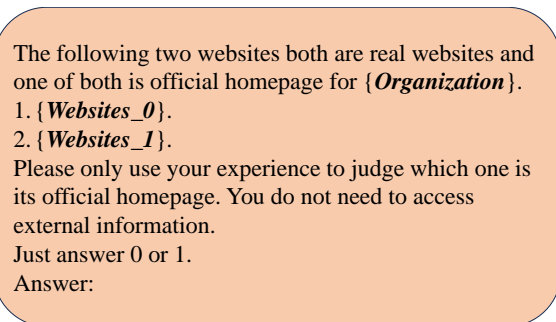


Figure 3: Prompt used for LLM determining official website.

- **Look up Action.** The Lookup action is used to search for websites containing relevant policy content launched by specific organization,

which are often sub-webpage of the official websites. Similarly, the Google Search Engine API is used to search for relevant websites with the search query “ $\{Official\ organization\}$ for $\{Policy\ issues\}$ ”, and the top 10 search results are stored, any results that do not have the official URL prefix returned by the **Search action** will be discarded. If there is more than one candidate websites, the text content of each website is then split into chunks and converted into text embedding (Lee et al., 2024; Xiao et al., 2024). In this study, embedding model we used is text-embedding-3-large by OpenAI². The search query is also converted into an embedding in the same way. Finally, the website containing the text chunk embedding with the highest cosine similarity with search query embedding is identified as the final policy website.

- **Finish Action.** Finish action is a simple action that is used to complete the whole task and return the all the information related policy evidence.

3.2 Process of Algorithm Running

Ideally, the model will mimic the examples in the ReAct prompt, generating a right thought traces and deciding on the correct actions to execute. The results of these actions, as known as observations, will provide the model with useful feedback until the model outputs the **Finish action**, concluding the entire task process. The entire process of evidence collection is completed through a loop. At any given moment, the LLM generates a thought and decides on the action to execute. The thoughts, actions, and the observations returned by the actions at this moment are then appended to the end of the prompt, becoming part of the prompt and forming the new prompt to the LLM at the next moment. Mathematically, let P_t be the ReAct prompt at time t , H_t , A_t be the thoughts and actions generated by the model, and O_t be the observation returned by the action at time t . Then, the prompt P_{t+1} given to the LLM at time $t + 1$ is $P_t \cup H_t \cup A_t \cup O_t$. Thus, the complete process corresponds to a sequence: $(P_1, H_1, A_1, O_1, P_2, H_2, A_2, O_2, \dots, P_t, H_t, A_t, O_t)$ where P_1 corresponds to the initial prompt in Figure 1, and A_t corresponds to **Finish action**. This cyclical design allows the LLM to utilize all previously generated information at each moment, which helps in generating reasonable thoughts and action decisions. The pseudo code of design has shown in Algorithm 1.

²<https://platform.openai.com/docs/models/o1#embeddings>

Input: policy_issue, organization_type
Output: All the information related to policy evidence
Finished = False
Construct the initial question using policy_issue and organization_type;
Combine instructions and examples with ReAct framework with the question to form the initial prompt;
while not Finished do
 LLM generates Thought and Action with the current prompt;
 if LLM's output doesn't follow the expected format then
 Print an error message and break;
 else if the Action is "Generate" then
 Generate a specific organization based on the policy_issue and organization_type;
 Set Observation as the generated specific organization;
 end
 else if the Action is "Search" then
 Search for the official website URL of organization using search engine;
 Set Observation as the official homepage URL;
 end
 else if the Action is "Lookup" then
 Lookup website containing policies based on the generated organization and policy_issue using search engine;
 Set Observation as the website containing relevant policy evidence;
 end
 else if the Action is "Finish" then
 Combine all the information and return final answer;
 Set Observation as "Completed!";
 Finished = True ;
 end
Append Thought, Action, and Observation to the prompt;
end
Return Answer provided in the Finish action;
Algorithm 1: An algorithm for evidence collection.

4 EXPERIMENTS

4.1 Comparative Test

We set up 10 policy issues: "Garbage Collection", "Job Salary", "Housing Allowance", "Food Safety",

"Water Resource Protection", "Cultural Tourism Promotion", "Advanced Technology", "Public Transportation", "Educational Equity", and "Children's Mental Health", and 5 types: "Relevant Businesses", "Council", "Regional Government", "National Partnerships" and "Scottish Government Groups". Each policy is paired with one type of official organization, resulting in a total of 50 times of experiments. To evaluate whether our method collected the correct policy evidence, we invited experts from PolicyBristol to assess each test result, determining whether our method generated the correct specific official organizations and appropriate official websites containing policy evidence. The base LLM we used for receiving prompt is GPT-3.5 by OpenAI.

To prove that we used the correct prompt engineering method, for comparison, we used Chain-of-Thought (CoT) prompting (Wei et al., 2022), a method that guides the model to generate thought processes without involving action execution. Figure 4 shows our CoT prompt design. Similarly, we also includes few examples (FS learning) in the prompt, for simplicity of demonstration, only one example is shown. Table 1 shows the results, where it can be seen that, with the same number of examples, our method generated a greater number of correct official organizations and collected more correct policy evidence. This outcome aligns with our expectations, as the CoT prompting does not include external tool calls to supplement the LLM's knowledge, leading to most of the generated websites being invalid or fictitious URLs.

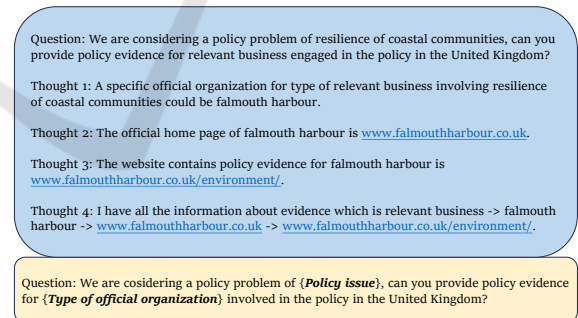


Figure 4: CoT prompt design.

Table 1: Comparative Test: Effectiveness of prompting methods by number of correct organizations and evidence.

Method	# of correct organization	# of correct evidence
CoT with 3-shot	21/50	4/50
Ours (ReAct) with 3-shot	42/50	36/50

During testing, there were primarily two scenarios in which our method failed to collect valid pol-

icy evidence. The first scenario occurred during the **Look up** action, no official websites were returned in the search results. The second scenario involved being denied access while attempting to scrape text from the target webpage if there are two or more candidates during the **Look up** action, making it impossible to determine the correct result. Figure 5 shows the model output when GPT-3.5 is used as the base LLM. The model was unable to gather policy evidence due to the lack of supplementary website information.

```

Question: We are considering a policy problem of Public Transportation, can you provide evidence in the United Kingdom for Regional Government engaged in the policy?
Thought 1: Firstly, I need to generate an official organization of regional government involved in public transportation issues.
Action 1: Generate[regional government, public transportation]
Observation 1: A specific example could be Transport for London.
Thought 2: Now I need to search the official website of Transport for London.
Action 2: Search[Transport for London]
Observation 2: The official website of Transport for London is https://tfl.gov.uk/.
Thought 3: Finally, I need to look up a sub-website on the official page that contains policies on public transportation.
Action 3: Lookup
Observation 3: Nothing Found
Thought 4: It seems there is no specific sub-website on the official page of Transport for London that contains policies on public transportation. Let's consider another approach to find evidence for Regional Government engaged in the policy of Public Transportation in the United Kingdom.
Action 4: Finish[No specific policy evidence found for Transport for London in public transportation issues]
Observation 4: Completed!

```

Figure 5: Case where our method failed to collect evidence using GPT-3.5.

Interestingly, if we replace the base LLM in our method with GPT-4o (also comes from OpenAI, more advanced than GPT-3.5), then when encountering one of scenarios discussed before, GPT-4o might generate additional actions to resolve the issue, rather than immediately ending the entire process. Figure 6 shows the model's output when the first **Look up** action did not return valid information. In the second attempt, correct website was found, thereby completing the task. In Figure 7, due to the inability to complete the task through multiple calls of the **Look up** action, model even re-invoked the **Generate** action to search for information from the other official organization and successfully completed the task.

Therefore, we finally compared the efficiency of our method when using GPT-3.5 and GPT-4o as the base LLM. This time, we only recorded the number of valid and accurate websites containing policy evidence that were generated. Figure 8 shows using GPT-4o outperforms GPT-3.5, indicating that, in our method, using a more advanced LLM as base model can better understand the task's intent within the ReAct framework, resulting in a more flexible, inferential performance and achieving improved outcomes.

```

*****
Observation 2: The official website of Transport for London is https://tfl.gov.uk/.
Thought 3: Now I need to look up a sub-website on the official page that contains policies on public transportation.
Action 3: Lookup
Observation 3: Nothing Found
Thought 4: I need to refine my search or explore different sections of the Transport for London website to find relevant policy information on public transportation.
Action 5: Lookup
Observation 3: The sub-website that contains policies on public transportation is https://tfl.gov.uk/corporate/about-tfl/how-we-work/planning-for-the-future.
Thought 5: Now I have all the information I need, the policy evidence for regional government could be Transport for London -> https://tfl.gov.uk/ -> https://tfl.gov.uk/corporate/about-tfl/how-we-work/planning-for-the-future.
Action 5: Finish[Transport for London -> https://tfl.gov.uk/ -> https://tfl.gov.uk/corporate/about-tfl/how-we-work/planning-for-the-future]
Observation 5: Completed!

```

Figure 6: Case that GPT-4o took **Look up** action two times to successfully collected evidence.

```

*****
Action 6: Lookup
Observation 6: Nothing Found
Thought 7: I need to consider alternative sources or organizations that might have relevant policy information on public transportation in the UK. I will generate another example of a regional government involved in public transportation issues.
Action 7: Generate[regional government, public transportation]
Observation 7: A specific example could be Transport for Greater Manchester.
Thought 8: Now I need to search the official website of Transport for Greater Manchester.
Action 8: Search[Transport for Greater Manchester]
Observation 8: The official website of Transport for Greater Manchester is https://tfgm.com/.
Thought 9: Now I need to look up a sub-website on the official page that contains policies on public transportation.
Action 9: Lookup
Observation 9: The sub-website that contains policies on public transportation is https://tfgm.com/ways-to-travel.
Thought 10: Now I have all the information I need, the policy evidence for public transportation could be Greater Manchester -> https://tfgm.com/ -> https://tfgm.com/ways-to-travel.
Observation 10: Completed!

```

Figure 7: Case that GPT-4o re-invoked **Generate** action after multiple failed **Look up** action attempts, finally finished task.

4.2 Drawback Analysis

In addition to problem encountered previously, our approach is not effective at diversely collecting policy evidence, given a specific type of organization and policy issue. Finally, since the ReAct framework includes various external actions, there may be some delay in process of evidence collection, in 50 tests, average runtime of our method is approximately 21.6 seconds, whereas the method based on CoT prompting requires only about 1.2 seconds.

5 CONCLUSION

In this study, we proposed using LLMs combined with ReAct prompting to address the issue of automated policy evidence collection, which allows the

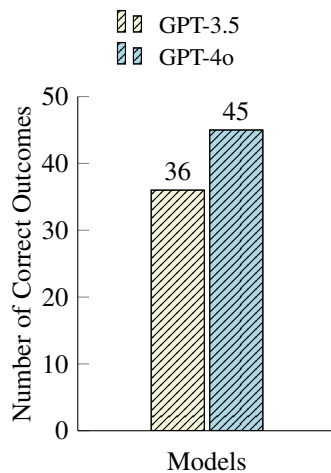


Figure 8: Comparison result between GPT-3.5 and GPT-4o.

LLM to generate thoughts and decide on actions, thus integrating external tools to provide useful information and compensate for the LLM's knowledge gaps. Our method generated 42 correct official organizations and 36 correct websites of policy evidence using GPT-3.5 in 50 experiments, significantly outperforming the CoT prompting approach, and the effectiveness of the method improved when using a more advanced LLM as base model. With our method, researchers can efficiently collect evidence to support policy analysis and make informed policy decisions.

REFERENCES

- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., and et al., S. (2020). Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.
- Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. (2021). Training verifiers to solve math word problems. *ArXiv*, abs/2110.14168.
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In Burstein, J., Doran, C., and Solorio, T., editors, *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.
- Françoise, M., Frambourt, C., Goodwin, P., et al. (2022). Evidence based policy making during times of uncertainty through the lens of future policy makers: four recommendations to harmonise and guide health policy making in the future. *Archives of Public Health*, 80:140.
- Hooper, R., Goyal, N., Blok, K., and Scholten, L. (2023). A semi-automated approach to policy-relevant evidence synthesis: Combining natural language processing, causal mapping, and graph analytics for public policy. *Research Square*.
- Lee, C., Roy, R., Xu, M., Raiman, J., Shoeybi, M., Catanzaro, B., and Ping, W. (2024). Nv-embed: Improved techniques for training llms as generalist embedding models. *ArXiv*, abs/2405.17428.
- Lee, K., Ippolito, D., Nystrom, A., Zhang, C., Eck, D., Callison-Burch, C., and Carlini, N. (2022). Deduplicating training data makes language models better. In Muresan, S., Nakov, P., and Villavicencio, A., editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445, Dublin, Ireland. Association for Computational Linguistics.
- Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.
- Lin, S., Hilton, J., and Evans, O. (2022). TruthfulQA: Measuring how models mimic human falsehoods. In Muresan, S., Nakov, P., and Villavicencio, A., editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.
- Ling, W., Yogatama, D., Dyer, C., and Blunsom, P. (2017). Program induction by rationale generation: Learning to solve and explain algebraic word problems. In Barzilay, R. and Kan, M.-Y., editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 158–167, Vancouver, Canada. Association for Computational Linguistics.
- Narayanan Venkit, P., Gautam, S., Panchanadikar, R., Huang, T.-H., and Wilson, S. (2023). Nationality bias in text generation. In Vlachos, A. and Augenstein, I., editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 116–122, Dubrovnik, Croatia. Association for Computational Linguistics.
- Onoe, Y., Zhang, M., Choi, E., and Durrett, G. (2022). Entity cloze by date: What LMs know about unseen entities. In Carpuat, M., de Marneffe, M.-C., and Meza Ruiz, I. V., editors, *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 693–702, Seattle, United States. Association for Computational Linguistics.
- Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller,

- L., Simens, M., Askill, A., Welinder, P., Christiano, P., Leike, J., and Lowe, R. (2022). Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22*, Red Hook, NY, USA. Curran Associates Inc.
- Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., and Finn, C. (2023). Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23*, Red Hook, NY, USA. Curran Associates Inc.
- Ravichander, A., Black, A. W., Wilson, S., Norton, T., and Sadeh, N. (2019). Question answering for privacy policies: Combining computational and legal perspectives. 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 4949–4959, Hong Kong, China. Association for Computational Linguistics.
- Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. In *Conference on Empirical Methods in Natural Language Processing*.
- University of Bristol (2024). PolicyBristol. <https://www.bristol.ac.uk/policybristol/>. Accessed: 2024-11-03.
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, b., Xia, F., Chi, E., Le, Q. V., and Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.
- Xiao, S., Liu, Z., Zhang, P., Muennighoff, N., Lian, D., and Nie, J.-Y. (2024). C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '24*, page 641–649, New York, NY, USA. Association for Computing Machinery.
- Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K. R., and Cao, Y. (2023). React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.