

# Case Hallucinations and Steps Toward Repair

Kaitlynn Wilkerson, David Leake

*Luddy School of Informatics, Computing, and Engineering, Indiana University, Bloomington, IN, USA*

## Abstract

Hallucinations are a well-known problem for Large Language Models (LLMs). In past work, we hypothesized that providing LLMs with problem-solving cases and prompting them to solve problems by a case-based reasoning process might reduce the risk of hallucinated solutions and improve accuracy. However, we discovered that simply providing LLMs with cases is not enough, as they may hallucinate the contents of the provided cases themselves. In this paper, we explore the prevalence of this issue using Llama 3 70B and present an approach to identifying and repairing case hallucinations as they are produced. In initial tests, our method decreased final hallucination rates for provided cases by 70 to 100%. We consider these results promising for improving the ability of LLMs to reliably apply provided cases and explain their reasoning in terms of those cases.

## Keywords

ChatGPT, Llama, Large Language Models, Hallucinations, Case Based Reasoning, Retrieval Augmented Generation

## 1. Introduction

Hallucinations are a well-documented problem for Large Language Models (LLMs), catching the attention of scientists, regulators, and the public. Despite the attention and concern, they arise from fundamental characteristics of the functioning of LLMs [1, 2], attributable to the statistical nature of their processing as well as the lossy encoding of information as they generalize training data [1, 3]. Because hallucinations are intrinsic to LLMs, it is essential that any application relying on LLMs has a way to detect, and potentially reduce, their occurrences.


The use of external knowledge has become a popular and proven method for detecting and repairing hallucinations in LLM-generated responses [3, 4, 5]. The rationale is that external knowledge can provide a sanity check for a response as well as grounding knowledge for repairing hallucinations that have been detected; so if the response does not refer to that knowledge, the response should be checked for a hallucination and required to use that knowledge in the newest iteration of the response. External knowledge can be provided in multiple forms, including the fact-based information used in most work on Retrieval Augmented Generation (RAG), and episodic information in the form of cases [6]. In previous research [7] we proposed guiding LLMs to perform the case-based reasoning (CBR) cycle [8, 9] to increase reliability and explainability of LLM outputs, under the hypothesis that LLMs provided with cases would generate solutions based on them. Initial results were encouraging, but we observed that occa-

---

*2nd Workshop on Case-Based Reasoning and Large Language Model Synergies (CBR-LLM) at ICCBR 2025, June 30, 2025, Biarritz, France*

The authors contributed equally.

 0000-0002-8666-3416 (D. Leake)

 © 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

sionally LLMs could fall prey to hallucinations *within the cases themselves*—apparently basing reasoning on a partially or fully hallucinated case instead of the original. However, in tests with Llama 2 this occurred at a relatively low rate, and a simple protocol of rerunning queries that contained hallucinated cases was sufficient for removing them.

The hallucination rate for cases changed when we tested our method on Llama 3 70B. Hallucinations were a frequent problem and were resistant to change using our previous protocol of simply re-running the query, which was sufficient in practice for Llama 2. This was surprising because of the expectation that Llama 3, introduced as a major advancement over Llama 2, would perform better than Llama 2. This motivated us to investigate the prevalence and impact of hallucinations in cases used by LLMs, and methods for identifying and reducing hallucinations when an LLM uses the CBR cycle and cases for reasoning. This paper makes two contributions. First, it presents initial experiments assessing the prevalence of case hallucinations for Llama 3. Second, it introduces a hallucination feedback module that acts as a plug-and-play (PnP) module for our original system interface; it scans for hallucinations in a response and, if one is detected, constructs a feedback prompt instructing the LLM to fix the problem. This approach produced a 70 to 100% reduction in hallucination rates, depending on the CBR prompt being used. The paper begins with an outline of the prevalence of the hallucination problem, along with early attempts to fix it, and hallucination reduction strategies in the literature. It then describes our Hallucination Feedback module and presents experimental results on its effects.

## 2. Background

### 2.1. Motivations for Implementing CBR with LLMs

In previous work [7], we performed initial studies on prompting LLMs to perform a case-based reasoning process. The general goals of this approach included:

1. Reducing the hallucination risk and increasing LLM accuracy by grounding LLM reasoning in cases; and
2. Increasing the trustworthiness and convincingness of explanations as explanations generated by cases are found to be more intuitive [10, 11].

We conducted experiments to assess the accuracy effects of prompting LLMs to perform CBR with ChatGPT and Llama 2, and results were encouraging. However, tests with Llama 3 revealed an unexpected issue: That hallucinations frequently changed the features of cases provided to the system (see Table 1). After the first round of tests, we re-ran the queries to assess changes in the hallucination rate, resulting in a second set of results. Table 1 displays the hallucination rates for our first and second rounds of testing for Llama 3 and compares them to Llama 2.

### 2.2. Initial Test Results

Our initial experiments compared two different prompt types, Implicit CBR (ICBR) and Explicit CBR (ECBR), against an LLM’s baseline solution on a medical triage task [7]. The ICBR prompt provided the LLM with KNN-selected cases and asked the LLM to adapt the prior triage scores based on the current patient information. The ECBR prompt included a list of cases for different

Prompt Types		Llama 2	Llama 3 Attempt 1	Llama 3 Attempt 2
ECBR 1NN		0%	32%	16%
ECBR 2NN	NN	0%	32%	16%
	2NN	4%	68%	56%
ECBR NUN	NN	0%	32%	24%
	NUN	0%	44%	40%

**Table 1**

Rates of hallucinations affecting provided cases for Llama 2, and Llama 3 on two different, consecutive attempts. The ECBR 2NN and NUN rows have two different values because two selections (i.e., two cases need to be selected for this prompt) are being made by the LLM: NN (first selection) and 2NN or NUN (second selection). For more details, see Section 4.2

patients and the LLM was asked to perform similarity assessment followed by adaptation. The major difference between the prompts boiled down to whether the LLM was provided information to reason with (ICBR) or asked to select information to reason with (ECBR). These prompts were tested on three different input conditions, 1-NN (providing the LLM with only the nearest case to the query), 2-NN (providing the two nearest cases), and Nearest Neighbor + Nearest Unlike Neighbor (NUN).

The cases provided to the LLMs via prompt were sourced from a primary triage dataset on Kaggle.<sup>1</sup> Cases contained seven features commonly discussed in triage literature: sex and age of the patient, heart rate, respiratory rate, mental state, blood pressure and the patient’s chief complaint. The target value listed in the case was the Korean Triage Acuity Score (KTAS) provided by the dataset. The KTAS score ranks a patient’s need for medical attention on a scale of 1 to 5, with 1 indicating the need for immediate treatment. A random selection of 102 cases from the original dataset were used in the case base; cases were selected via weighted k-NN prior to being added to the LLM prompt.

To obtain a baseline for LLM performance, we prompted ChatGPT and Llama 2 to provide a triage score for each test case without any additional knowledge or instruction. The LLM baseline prediction accuracy for both systems was 28%. Using both ChatGPT and Llama 2, all our prompts and information setups yielded results at, above, or significantly above the LLM baseline levels. The highest performing prompt for both systems was ICBR 1NN, for which the accuracy was 60% and 56% for ChatGPT and Llama 2, respectively. (We note in contrast to the Llama 2 results, the results with ChatGPT are not reproducible, because of variant uncertainties, etc., so that accuracy figure is only illustrative for the performance of one commercial system at the time of the tests.)

We found that the ICBR prompts tended to perform better than the ECBR prompts. We attributed this to the poor similarity assessment ability of the LLMs used, because the only difference between the two prompts was that ECBR had to choose which case was most similar to the problem case before it could perform adaptation. Comparing how often the LLM chose the same cases as the weighted KNN system, which we treated as gold standard, we found that the overlap rate in cases chosen was between 4 and 32% [7].

<sup>1</sup><https://www.kaggle.com/datasets/ilkeryildiz/emergency-service-triage-application>

### 2.3. Test Results with Llama 3

Our first follow-up test on this method used Llama 3 70B-Instruct and the same internal parameters as the Llama 2 model. We expected Llama 3 to outperform Llama 2 given that it was the latest model released by Meta. However, we were unable to verify this hypothesis due to Llama 3’s high hallucination rates on ECCR tasks (see Table 1). In these hallucinations, the LLM changed one or more features of the provided cases. Interestingly, the hallucination problem did not extend to the ICCR task. The reason for this discrepancy is unclear, but could be related to the ECCR prompt containing a set of ten cases and asking for similarity assessment instead of only having one or two cases provided. To produce a fair comparison of Llama 3 to Llama 2’s performance on the triage task, Llama 3 needed to have no hallucinations present in its responses for the response to be accepted. This policy was originally instituted to ensure the model’s reasoning was based on the case(s) being provided to or selected by it. When hallucinations occurred with Llama 2, the test prompt was resubmitted without change to the LLM until a hallucination was no longer detected. Our first attempt to reduce hallucinations in Llama 3 used this approach, but it, at best, reduced hallucinations by 50%. The continued presence of hallucinations meant that we needed to produce a better method for reducing them before a fair comparison could be made between models. This paper takes steps towards such a method; we intend to return to the question of relative performance of Llama 2 and 3 after addressing the hallucination problem more deeply.

## 3. Hallucination Reduction Strategies

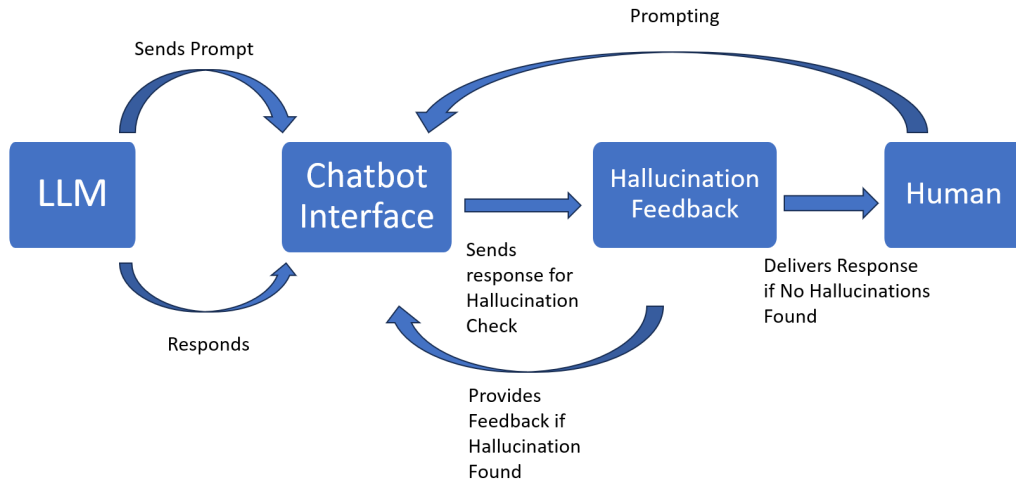
As discussed in the introduction, external knowledge can support a "sanity check" for hallucinations in LLM responses [7]. Using such knowledge is a common approach in the literature on hallucination reduction strategies, but strategies may differ in two ways:

1. Whether hallucinations are detected in real time [5] or post-hoc [3, 12] and,
2. Whether LLM responses are edited to align with ground truth information [12, 4] or whether LLMs are re-queried with feedback messages [3, 4] (thus allowing the LLM to fix its mistakes on its own).

Feedback messages tend to contain:

1. External knowledge to the LLM [3, 4, 5],
2. Either information about why the initial response was not useful (e.g. a utility score on the response) [3] or instructions on how to fix the hallucination [4], and
3. Occasionally, examples of the task and response (i.e., few-shot prompting) [13].

As an illustration, Lei [4] provides the following template for feedback messages: "Refer to the knowledge: "final knowledge" and answer the question: XXX with one paragraph." This simple prompt clearly delineates the information provided as either knowledge to be used or task to be completed, with the goal of facilitating the LLM applying the knowledge to the task.



**Figure 1:** Hallucination Feedback Module Workflow

## 4. Hallucination Feedback Module

### 4.1. Module Design

Given that one of our motivations for implementing CBR in LLM models is improving the explainability of LLM decisions, real-time hallucination detection and feedback are important. Our detection system works as a PnP method, with hallucination identification and correction included in the system's reasoning log in the order in which the events occurred, increasing transparency of the roles of detection and repair in system processing.

The Hallucination Feedback Module (Figure 1) scans for hallucinations in LLM responses (i.e., case selection responses in which the "case" returned by the LLM does not align with a case provided to the LLM in the prompt) as they are received, and provides feedback prompts to the LLM in response if needed. Feedback prompt responses are also scanned for hallucinations in cases; If a hallucination is still detected, the LLM is provided with a new feedback prompt. This cycle can be repeated up to three times (to prevent infinite looping); when a problem is still detected after three attempts, the system attaches a message to the LLM response reporting that human review is needed for the case and proceeds to the next prompt. The module was integrated into our original experimental workflow [7]; the only necessary change was that previous responses were received and recorded to a text file for future verification, and now are subject to this detection and feedback cycle before being recorded to a text file.

## 4.2. Hallucination Types and Feedback Prompts

As discussed in the introduction, we classified hallucinations into two categories: partially hallucinated and fully hallucinated. A partially hallucinated case is one which has no exact match to any case in the case base provided to the LLM, but for which it is possible to identify a corresponding case in the case base. For particular domains, different criteria would be appropriate for determining correspondence. Here we adopt a simple criterion: We consider a case to correspond if a majority of its features match those of at least one of the provided cases. If it corresponds but has no exact match, we consider it partially hallucinated; if it does not correspond to any case in the case base we consider it fully hallucinated. In practice, we only encountered partial hallucinations with up to two perturbed feature values. For partially hallucinated cases, the feedback prompt refers to the reference case to guide correction; for fully hallucinated cases, it directs the LLM to select a case from the case base. These prompts can be seen in Table 2; square brackets indicate where information is provided to the LLM and 'Assistant' is used to indicate when knowledge provided to the LLM was from the previous response.

Feedback prompts for partial hallucinations provide the LLM with the hallucinated response, the reference case and information regarding the differences between the two. Then it is instructed to correct the hallucinated case using the provided information. During pre-experiment testing, Llama 3 struggled to correctly complete this task and would return the hallucinated case with no changes. A static example of a good and bad response was added to the bottom of the prompt, which helped correct this problem.

Prompts for full hallucinations are divided into two subcategories: first and second choice. This refers to whether the prompt is using ECBR 2NN or NUN, and if the LLM should be selecting a 2NN or NUN instead of a NN. The only difference between the two is that second choice prompts provide the LLM with the first choice and instruct it to pick a case different from that. Otherwise, both prompts inform the LLM of its mistake, provide it with the case list again, and instruct it to choose a new case from the list.

Hallucination Type	Feedback Prompt
Fully Hallucinated:	<p><b>First Choice:</b></p> <p><i>The patient that you chose is not one of the previous patients provided for you to choose from. Refer to the following: [] and select a patient from this list that best matches the current patient. Please format your response as 'Selection: [Patient Information]'.</i></p> <p><b>Second Choice:</b></p> <p><i>The patient that you chose is not one of the previous patients provided for you to choose from. Refer to the following: [] and select a next most similar patient from this list that best matches the current patient. Do not choose [] as you have already picked this as the most similar patient. Please only list the new selection in your response and format your response as 'Selection: [Patient Information]'.</i></p>
Continues on Next Page	

Type and Formulation	Prompt(s)
Partially Hallucinated	<p>Assistant: []. Feedback from helper: It appears that you incorrectly listed some of the patient information in your response. Refer to the following: [] and fix the patient information with the corrected information. The difference between the selected information and the correct information is: []. TASK INSTRUCTIONS: You are given a patient's incorrect information. Your task is to find the correct patient data based on the provided corrections. The incorrect patient information will be labeled 'Assistant:'. The correct patient information will be in the 'Feedback from helper:' section. Your job is to correct only the incorrect details (heart rate, blood pressure, etc.) while keeping all other information the same. Do not use any information from examples — only use the correction data from the 'Feedback from helper:' section. Please only list the new selection in your response and format your response as 'Selection: [Patient Information]'. EXAMPLE: Bad Response: Selection: [Sex: Female, Age: 55, Chief complaint: ocular pain. Lt., Mental state: Alert, Heart rate: 80, Respirations: 20, Blood pressure: 110/70, Triage number: 4] Good Response: Selection:[Sex: Female, Age: 55, Chief complaint: ocular pain. Lt., Mental state: Alert, Heart Rate: 103, Respirations: 20, Blood Pressure: 135.0/101.0 Triage Number: 4]</p>

Table 2: Prompt(s) used for each hallucination type.

## 5. Experimental Setup

### 5.1. Hallucination Reduction

To examine the performance of the Hallucination Feedback module, we retested our prior experimental setup [7] with the module integrated into the test environment. We reused the same parameters, case base, and ECBR prompts from [7], with the exception of temperature, which will be discussed shortly. The system was tested on the same 25 test cases and two metrics were recorded: system accuracy and total percentage of hallucinations detected for each ECBR prompt and case set up pair. These results will then be directly compared to the original experimental results in terms of accuracy and hallucination rate.

### 5.2. Temperature Changes

Additionally, we wanted to explore the role of temperature on case hallucination reduction. Temperature is an internal parameter for LLMs in the range [0, 1] that controls for the amount of “creativity” allowed, with 0 making choices deterministic. Our initial experiments used a temperature of 0 for two reasons: it provided the best performance on the tasks and allowed our results to be reproducible. However, it is possible that the low temperature could hamstring efforts to eliminate hallucinations from responses, and thus, that temperature adjustment might provide a simple approach to reducing hallucinations. To examine this possibility, we tested

Prompt Type and Temperature		Original	Hallucination Feedback
ECBR 1NN	0	32%	8%
	0.1	32%	8%
	0.2	32%	4%
	0.5	28%	8%
	1.0	20%	4%
ECBR 2NN (NN)	0	32%	8%
	0.1	32%	8%
	0.2	36%	8%
	0.5	36%	0%
	1.0	32%	0%
ECBR 2NN (2NN)	0	68%	12%
	0.1	64%	12%
	0.2	72%	12%
	0.5	80%	12%
	1.0	80%	0%
ECBR NUN (NN)	0	32%	0%
	0.1	34%	0%
	0.2	32%	0%
	0.5	40%	4%
	1.0	48%	0%
ECBR NUN (NUN)	0	44%	8%
	0.1	52%	8%
	0.2	48%	4%
	0.5	56%	8%
	1.0	72%	8%

**Table 3**

Hallucination Rates for the Original Experiment and the Hallucination Feedback Experiment, organized by prompt type and temperature.

both the original experiment and the Hallucination Feedback module on temperatures of 0, 0.1, 0.2, 0.5, and 1.0. Temperature values are clustered on the lower end of the spectrum because deterministic results are better for reproducibility, and we wanted to examine whether a small tradeoff in reproducibility had any effect on hallucination rates.

## 6. Results and Discussion

### 6.1. Hallucination Feedback Module

The Hallucination Feedback module reduced hallucination rates by 70 to 100% depending on prompt type and temperature (Table 3). At best, our original attempts to reduce hallucinations



Prompt Type and Temperature		Original	Hallucination Feedback
ECBR 1NN	0	40%	28%
	0.1	40%	28%
	0.2	36%	20%
	0.5	36%	24%
	1.0	32%	20%
ECBR 2NN	0	36%	32%
	0.1	32%	32%
	0.2	44%	32%
	0.5	44%	28%
	1.0	48%	32%
ECBR NUN	0	36%	32%
	0.1	36%	32%
	0.2	36%	32%
	0.5	24%	28%
	1.0	28%	20%

**Table 4**

Accuracy Rates for the Original Experiment and the Hallucination Feedback Experiment, organized by prompt type and temperature.

were 50%, so this module improved reduction rates by 20 to 50% over the original method. However, the module did not successfully remove all hallucinations for most of the prompts and temperatures tested. All of the remaining hallucinations were partial hallucinations. Given that the LLM was resistant to changing partial hallucinations in the pre-experiment tests, it is unsurprising that any remaining hallucinations were partial and indicates that further tweaking or experimentation needs to be done on this prompt. For example, we believe that minor tweaks to our system, such as using dynamic examples that are better related to the provided case(s), might help reduce the number to zero across the board.

Even though our module was successful at its intended purpose, there was an unintended consequence of removing hallucinations: reduced accuracy (Table 4). Drops in accuracy rates ranged from 0 to 44%, implying that some partially hallucinated cases were helpful to the LLM CBR process. It suggests that in some instances, the hallucinated cases generated by the LLM were a useful supplement to the case base; we can see this as a process of generating useful “ghost cases,” artificial cases which have proven useful in other contexts [14], with the LLM itself generating the cases. Further research will need to be done on whether helpful hallucinations can be identified and selectively targeted for *inclusion* to improve accuracy rates. This has potentially interesting ramifications for using LLMs for case-base augmentation more generally, whether or not the CBR process is carried out by an LLM. The effects of hallucinated cases, especially helpful ones, on trustworthiness should be examined as well.

## 6.2. Temperature Change

Temperature had a mild affect on hallucination rates for the original experimental setup, where greater temperatures lead to more hallucinations on average. However, this was not always the case; ECBR 1NN actually shows an inverse effect, where hallucination rates drop with increasing temperature. Neither of these trends were detected for the Hallucination Feedback module, implying that it is good at regulating hallucination reductions, even if increased temperature produces more of them. This could make the Hallucination Feedback module potentially useful for commercial LLMs (such as ChatGPT), where there is no control over the temperature parameter.

Both the original experiment and the Hallucination Feedback module experienced decreases in accuracy. This seems to further imply that at least some portion of the hallucinations generated, regardless of temperature, led to accurate triage assessments and that indiscriminate removal of hallucinations is not helpful.

The only exception to this trend was ECBR 2NN, which benefited from increased temperatures in the original experiment and saw stagnant accuracy with the Hallucination Feedback module. The reason for this deviation is currently unclear, but may be related to the percentage of hallucinations that were helpful to the LLM and which ones the module was unable to remove.

## 7. Conclusion and Future Work

There has recently been great interest in integrating CBR with LLMs, to leverage the strengths of both [15]. One promising direction is to use LLMs to implement the CBR cycle, with the goal of increasing LLM accuracy, due to reasoning from specific cases rather than generalizations from training, and explainability, due to being able to present those cases as explanations. However, a surprising observation is that when cases are presented to an LLM, the LLM may hallucinate parts of those specific cases. This paper presents first steps on understanding the prevalence of such hallucinations and initial methods for their detection and repair. In future work we plan to perform more extensive evaluations to assess this problem, and to develop deeper strategies for detection and repair.

## Acknowledgments

This work was funded by the US Department of Defense (Contract W52P1J2093009). This research was supported in part by Lilly Endowment, Inc., through its support for the Indiana University Pervasive Technology Institute. We thank Zachary Wilkerson for his helpful comments on a draft of this paper.

## Declaration on Generative AI

The author(s) did not use any generative AI during the preparation of this work.

## References

- [1] K. Hammond, D. Leake, Large language models need symbolic AI, in: Proceedings of the 17th International Workshop on Neural-Symbolic Learning and Reasoning, La Certosa di Pontignano, Siena, Italy, volume 3432, 2023, pp. 204–209.
- [2] Z. Xu, S. Jain, M. Kankanhalli, Hallucination is inevitable: An innate limitation of large language models, 2024. [arXiv:2401.11817](https://arxiv.org/abs/2401.11817).
- [3] B. Peng, M. Galley, P. He, H. Cheng, Y. Xie, Y. Hu, Q. Huang, L. Liden, Z. Yu, W. Chen, et al., Check your facts and try again: Improving large language models with external knowledge and automated feedback, *arXiv preprint arXiv:2302.12813* (2023).
- [4] D. Lei, Y. Li, M. Hu, M. Wang, V. Yun, E. Ching, E. Kamal, Chain of natural language inference for reducing large language model ungrounded hallucinations, *arXiv preprint arXiv:2310.03951* (2023).
- [5] H. Kang, J. Ni, H. Yao, Ever: Mitigating hallucination in large language models through real-time verification and rectification, *arXiv preprint arXiv:2311.09114* (2023).
- [6] N. Wiratunga, R. Abeyratne, L. Jayawardena, K. Martin, S. Massie, I. Nkisi-Orji, R. Weerasinghe, A. Liret, B. Fleisch, CBR-RAG: case-based reasoning for retrieval augmented generation in llms for legal question answering, in: International Conference on Case-Based Reasoning, Springer, 2024, pp. 445–460.
- [7] K. Wilkerson, D. Leake, On implementing case-based reasoning with large language models, in: International Conference on Case-Based Reasoning, Springer, 2024, pp. 404–417.
- [8] A. Aamodt, E. Plaza, Case-based reasoning: Foundational issues, methodological variations, and system approaches, *AI Communications* 7 (1994) 39–52.
- [9] R. López de Mántaras, D. McSherry, D. Bridge, D. Leake, B. Smyth, S. Craw, B. Faltings, M. Maher, M. Cox, K. Forbus, M. Keane, A. Aamodt, I. Watson, Retrieval, reuse, revision, and retention in CBR, *Knowledge Engineering Review* 20 (2005).
- [10] L. Gates, D. Leake, K. Wilkerson, Cases are king: A user study of case presentation to explain cbr decisions, in: International Conference on Case-Based Reasoning, Springer, 2023, pp. 153–168.
- [11] P. Cunningham, D. Doyle, J. Loughrey, An evaluation of the usefulness of case-based explanation, in: Case-Based Reasoning Research and Development: Proceedings of the Fifth International Conference on Case-Based Reasoning, ICCBR-03, Springer, Berlin, 2003, pp. 122–130.
- [12] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, H. Wang, Retrieval-augmented generation for large language models: A survey, *arXiv preprint arXiv:2312.10997* (2023).
- [13] S. Min, X. Lyu, A. Holtzman, M. Artetxe, M. Lewis, H. Hajishirzi, L. Zettlemoyer, Rethinking the role of demonstrations: What makes in-context learning work?, *ArXiv abs/2202.12837* (2022). URL: <https://arxiv.org/pdf/2202.12837>.
- [14] D. Leake, B. Schack, Exploration vs. exploitation in case-base maintenance: Leveraging competence-based deletion with ghost cases, in: Case-Based Reasoning Research and Development, ICCBR 2018, Springer, Berlin, 2018, pp. 202–218.
- [15] K. Bach, R. Bergmann, F. Brand, M. Caro-Martínez, V. Eisenstadt, M. W. Floyd, L. Jayawardena, D. Leake, M. Lenz, L. Malburg, et al., Case-based reasoning meets large language models: A research manifesto for open challenges and research directions (2025). URL:

<https://hal.science/hal-05006761>.