Review

Explainable Hallucination Mitigation in Large Language Models: A Survey

Wentao Deng , Jiao Li , Hong-Yu Zhang , Jiuyong Li , Zhenyun Deng * , Debo Cheng * , Zaiwen Feng *

Posted Date: 7 May 2025

doi: 10.20944/preprints202505.0456.v1

Keywords: Large language models; Hallucinations; Explainability; Interpretability; Prompt engineering; Factual consistency; Reasoning transparency

 doi:10.20944/preprints202505.0456.v1

Review

Explainable Hallucination Mitigation in Large Language Models: A Survey

Wentao Deng [1,†], Jiao Li [2,3,†], Hong-Yu Zhang [1,4], Jiuyong Li [5], Zhenyun Deng [6,*], Debo Cheng [5,*] and Zaiwen Feng [1,3,4,*]

[1] College of Informatics, Huazhong Agricultural University, Wuhan 430070, China
[2] Agricultural Information Institute, Chinese Academy of Agricultural Sciences, Beijing 100081, China
[3] Key Laboratory of Knowledge Mining and Knowledge Services in Agricultural Converging Publishing, National Press and Publication Administration, Agricultural Information Institute, Chinese Academy of Agricultural Sciences, Beijing 100081, China
[4] Hubei Key Laboratory of Agricultural Bioinformatics, College of Informatics, Huazhong Agricultural University, Wuhan 430070, China
[5] STEM, University of South Australia, Adelaide SA 5001, Australia
[6] Department of Computer Science and Technology, University of Cambridge, Cambridge CB3 0FD, UK
* Correspondence: zd302@cam.ac.uk (Z.D.); Debo.Cheng@unisa.edu.au (D.C.); zaiwen.Feng@mail.hzau.edu.cn (Z.F.)
† These authors contributed equally to this work.

Abstract: Hallucinations in large language models (LLMs) pose significant challenges to their reliability, especially in knowledge-intensive and reasoning-oriented tasks. While existing efforts have focused on detecting or correcting such errors, they often lack a unified interpretive framework for understanding the underlying causes and designing principled mitigation strategies. This survey investigates hallucination mitigation through the perspective of explainability, organizing a taxonomy that differentiates between internal and post-hoc interpretability methods. We analyze the role of techniques such as attribution tracing, reasoning path construction, and prompt-based verification in enabling both transparent diagnosis and structured intervention. Additionally, we reflect on the constructive role of hallucinations in creativity-driven and user-centered applications, and suggest that context-aware control may be more appropriate than universal suppression. By consolidating recent research, this survey advocates for the integration of explainability into the development of more transparent, controllable, and trustworthy language generation systems.

Keywords: large language models; hallucinations; explainability; interpretability; prompt engineering; factual consistency; reasoning transparency

1. Introduction

Large Language Models (LLMs), which are deep learning models pre-trained on massive corpora, have achieved remarkable success across a broad range of natural language processing (NLP) tasks, including text generation [1], machine translation [2], and conversational systems [3,4]. These models exhibit exceptional capabilities in both language understanding and generation [5,6]. However, a critical limitation increasingly observed in real-world applications is the phenomenon of *hallucinations*, which refers to the generation of content that is factually incorrect or logically inconsistent [7]. Such errors undermine the credibility of LLM outputs and pose serious risks in high-stakes domains like healthcare [8], law [9], and finance [10], where accuracy and reliability are essential [11].

A major contributing factor to hallucinations is the insufficient explainability in LLMs [12]. In this survey, we define the term *explainability* as a broad set of techniques that aim to make the internal decision-making processes of models more transparent. We distinguish this from *interpretability*, which pertains to the human-comprehensibility of model behavior or architecture. Interpretability is thus foundational to explainability, particularly for black-box models such as LLMs. Despite their fluency and contextual appropriateness, LLMs often operate as opaque systems, limiting users' and researchers'

 doi:10.20944/preprints202505.0456.v1

ability to trace the origins of errors or validate generated content [13]. Moreover, human cognition is itself vulnerable to illusions—such as the Moses illusion—where plausible but incorrect information goes undetected [14]. These parallels reinforce the need for enhanced explainability as a means of addressing hallucinations.

Recent studies have increasingly focused on mitigating hallucinations by improving the explainability of LLMs. These methods aim to enhance transparency, traceability, and verifiability of model behavior, thereby strengthening factual alignment and user trust [12,15]. A prominent line of work introduces external retrieval mechanisms, including Retrieval-Augmented Generation (RAG) [16] and knowledge-aware generation [17], which provide factual grounding by incorporating external knowledge sources. Other approaches involve uncertainty-aware generation and counterfactual reasoning, which make the model's confidence and alternative reasoning paths more explicit [18,19]. Additionally, prompt-based strategies such as Chain-of-Thought (CoT) prompting support step-by-step reasoning, offering inherently interpretable reasoning traces. These explainability-oriented techniques contribute to both mitigating hallucinations and improving the robustness of LLMs.

This survey centers on hallucinations as a core challenge and explainability as a key analytical framework for understanding and mitigating them. We offer a structured review of recent work that leverages explainability to diagnose and address hallucinations in LLMs. Our goal is to articulate a coherent conceptual landscape that clarifies the interplay between explainability and hallucination, and to highlight promising directions for future research in this rapidly evolving area.

The remainder of this survey is organized as follows. Section 2 introduces a taxonomy of hallucinations and foundational concepts related to explainability in the context of LLMs. Section 3 reviews explainability-guided techniques for hallucination detection and mitigation. Section 4 concludes the survey with a summary and discussion of future directions.

2. Explainability and Hallucinations in LLMs

The explainability of LLMs refers to the ability of users to understand, trace, and reason about the decision-making processes that underlie model outputs. To systematically investigate the role of explainability in hallucination-related challenges, we categorize relevant techniques into two primary dimensions. The first is *internal explainability*, which focuses on increasing transparency within the model's internal mechanisms and decision paths. The second is *post-hoc explainability*, which emphasizes interpreting model behavior after the output has been generated. In this survey, we define explainability as the transparency of internal reasoning processes and the traceability of generated content, excluding aspects such as data preprocessing and input provenance, which are considered part of data governance.

Figure 1 presents an overview of the conceptual framework adopted in this survey. It illustrates how explainability techniques intersect with hallucination detection and mitigation strategies, organizing these methods into a coherent taxonomy.

Given the multifaceted nature of hallucination causes, we adopt an inclusive classification strategy that encompasses not only traditional interpretability methods but also complementary techniques. These include RAG and logical consistency checks, which, although not originally designed for model interpretation, offer valuable support for transparency, evidence tracing, and output verification. Within our framework, such mechanisms serve as critical tools for hallucination detection and mitigation.

This section introduces the core categories of explainability methods in LLMs, laying the groundwork for subsequent sections that analyze hallucination types and corresponding mitigation strategies. It also establishes a conceptual connection between interpretability techniques and the broader challenge of hallucination control.

 doi:10.20944/preprints202505.0456.v1

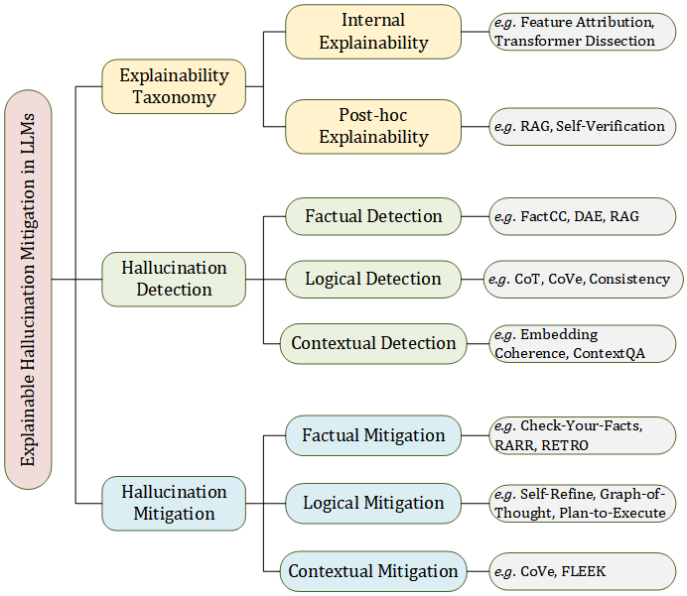

Figure 1. The main content flow and categorization of this survey.

2.1. Typology of Explainability Methods

We classify explainability methods into two broad categories: internal explainability and post-hoc explainability. Internal methods aim to elucidate the internal dynamics of LLMs during the process of output generation. In contrast, post-hoc methods interpret model behavior retrospectively, without modifying the model's architecture or inference process. Both categories encompass a range of techniques that enhance the transparency, reliability, and interpretability of LLM behavior.

2.1.1. Internal Explainability

Internal explainability aims to uncover how different components of LLMs, such as attention heads or neural sublayers, influence output generation. Two prominent classes of internal methods are feature attribution and transformer block dissection.

Feature Attribution

Feature attribution methods quantify the influence of each input feature (e.g., words, phrases, or text spans) on the model's output. Given an input sequence $x = \{x_1, x_2, ..., x_n\}$ and a model f, the output $f(x)$ is attributed to tokens x_i via relevance scores $R(x_i)$. These techniques fall into three categories: perturbation-based, gradient-based, and vector-based approaches.

- **Perturbation-based methods** assess feature importance by removing, masking, or altering input components and observing the resulting changes in the output [20,21]. Although intuitive, these methods often assume feature independence and may overlook interactions. They can also produce unreliable results for nonsensical inputs [22], prompting efforts to improve robustness and efficiency [23].
- **Gradient-based methods** compute derivatives of the model output with respect to input tokens, thereby measuring the sensitivity of outputs to specific inputs [24]. While widely adopted, these methods face concerns regarding computational cost and the reliability of attribution scores [25,26].
- **Vector-based methods** represent input features in high-dimensional embedding spaces and analyze their relationships to model outputs, using metrics such as cosine similarity or Euclidean distance [27,28]. These techniques account for inter-feature dependencies but are sensitive to embedding quality and representation space [29].

 doi:10.20944/preprints202505.0456.v1

Dissecting Transformer Blocks

This class of methods analyzes the internal operations of individual components within transformer layers, particularly in decoder-based LLMs. A standard transformer block includes two main sublayers: Multi-Head Self-Attention (MHSA) and a Multi-Layer Perceptron (MLP).

- **MHSA sublayers** capture dependencies between input tokens. Given input X, it is projected into queries (Q), keys (K), and values (V), with attention computed as $\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$. Outputs from all attention heads are concatenated. Interpretability is typically achieved through attention weight visualization [30] or gradient-based attribution methods [31].
- **MLP sublayers** consist of linear transformations and non-linear activation functions. A typical MLP computes $h = \text{Activation}(W_1 x + b_1)$ and $y = W_2 h + b_2$. These components are thought to function as memory units that retain semantic content across layers [32]. Notably, MLPs often contain more parameters than attention sublayers, underscoring their importance for feature abstraction [31].

2.1.2. Post-Hoc Explainability

Post-hoc explainability refers to methods that analyze and explain model outputs without altering internal model structures. These techniques aim to enhance transparency through auxiliary mechanisms such as retrieval, verification, and post-generation editing. Based on the timing of external intervention, we categorize post-hoc methods into three groups: before, during, and after generation.

Before Generation

This class of methods introduces factual guidance prior to model inference, typically through prompt augmentation, external retrieval, or knowledge grounding, to steer the generation process toward more accurate and verifiable outputs.

- **LLMs-Augmenter** integrates plug-and-play knowledge modules to build evidence chains that guide response generation, improving factuality and interpretability [33,34].
- **FreshPrompt** dynamically retrieves real-time information to construct adaptive prompts, addressing the limitations of static prompts in time-sensitive domains [35,36].

During Generation

This class of methods integrates real-time feedback and verification mechanisms directly into the generation process, enabling the model to monitor and adjust its outputs as they are being produced.

- **Knowledge retrieval with anomaly detection** identifies hallucination risks based on output logits and retrieves supporting evidence for in-process correction [17,34].
- **Decompose and Query (D&Q)** guides the model to decompose complex questions and retrieve intermediate knowledge at each step, offering verifiable checkpoints during inference [37,38].
- **EVER** introduces a generation–verification–revision loop that enables the model to validate each step and revise outputs when inconsistencies are detected [39,40].

After Generation

This class of methods operates post hoc, applying verification and revision procedures after the model has produced its output. The goal is to enhance factual accuracy, improve traceability, and reduce the risk of hallucinations by evaluating or refining the generated content.

- **RARR (Retrofit Attribution using Research and Revision)** aligns generated content with retrieved factual evidence while preserving narrative quality [41,42].
- **High entropy word spotting and replacement** identifies uncertain tokens in the output and replaces them with more reliable alternatives sourced from robust models [43,44].

 doi:10.20944/preprints202505.0456.v1

2.2. Typology of Hallucinations Under Limited Explainability

model explainability into three primary types: factual hallucinations, logical hallucinations, and contextual hallucinations. Table 1 provides representative real-world examples illustrating each category.

It is important to emphasize that this taxonomy is grounded in the deviation between model-generated content and either objective truth or contextual coherence, rather than in the underlying causes of hallucinations. For instance, certain factual hallucinations may result from inaccuracies or biases in the training corpus, which fall under the broader domain of data governance. However, from the standpoint of explainability, these errors signify a lack of transparency in the model's generative process, particularly its inability to offer traceable evidence or verifiable reasoning.

Consequently, we include such phenomena within an explainability-based framework, focusing on how the absence of interpretability mechanisms hinders the detection, diagnosis, and mitigation of hallucinations [7,12].

2.2.1. Factual Hallucinations

Factual hallucinations occur when a language model generates outputs that contradict objective facts or established knowledge. These errors are particularly concerning in domains requiring high factual accuracy, such as healthcare, legal reasoning, and scientific reporting . Factual hallucinations often arise when models lack access to accurate or up-to-date information, leading to content that cannot be externally verified.

Explainability Perspective. From an explainability standpoint, factual hallucinations underscore the model's inability to justify its outputs through identifiable knowledge sources or explicit reasoning paths. Key contributing factors include:

- **Erroneous Training Data:** Outputs influenced by misinformation or low-quality content present in the training data can result in factually incorrect generations [18].
- **Knowledge Boundaries:** The model may lack access to domain-specific or current knowledge, leading to outdated or incomplete responses [35].
- **Fabricated Content:** In the absence of verifiable knowledge, the model may generate plausible-sounding but entirely fabricated information, especially in response to ambiguous prompts [15].

2.2.2. Logical Hallucinations

Logical hallucinations arise when outputs violate principles of sound reasoning, often due to the model's inability to maintain logical consistency across inferential steps. These errors are common in tasks requiring multi-step reasoning, arithmetic operations, or deductive logic [43,45].

Explainability Perspective. The lack of transparent reasoning processes impedes both the interpretability and verifiability of outputs. Two main limitations are identified:

- **Insufficient Reasoning Ability:** The model fails to apply basic logical rules or computational procedures, resulting in conclusions that are either incorrect or unjustified [40].
- **Broken Reasoning Chains:** In multi-step tasks, the model frequently loses coherence across intermediate steps, generating outputs that lack continuity or internal justification [19,36].

 doi:10.20944/preprints202505.0456.v1

Table 1. Taxonomy of Hallucinations with Examples.

Type	Sub-Causes	User Input	Model Output	Explanation
Factual Hallucination	Data Source Error	What is the capital city of China?	The capital city of China is **Shanghai**.	The model incorrectly states Shanghai as the capital of China, likely due to exposure to inaccurate or biased training data.
	Knowledge Boundary	What is the primary treatment for CJD?	The primary treatment for CJD is **antibiotic therapy**.	The model generates medically incorrect content, reflecting limitations in domain-specific or up-to-date knowledge acquisition.
	Factual Fabrication	When did Japan surrender in WWII?	Japan surrendered in **December 1944**.	The model fabricates a historical fact in the absence of verifiable knowledge or source attribution.
Logical Hallucination	Insufficient Reasoning	Solve $2x - 3 = 7$	$2x = 10 \Rightarrow x = \mathbf{2}$	The model demonstrates a failure in symbolic reasoning by producing a mathematically incorrect solution.
	Broken Reasoning Chain	Mary is 4 years older than John (age 12). How old is Mary?	Mary is **16**. Next year, she will be **5 years older** than John.	The model fails to preserve logical continuity, producing a temporally inconsistent and unjustified conclusion.
Contextual Hallucination	Insufficient Context Tracking	(round-1): I watched The Lord of the Rings yesterday. It was amazing! (round-2): Yes, The Lord of the Rings is a classic. Who's your favorite character?	(round-1): I like Gandalf the most. His wisdom and power really impressed me. Do you know who played him? (round-2): The **director** of The Lord of the Rings is Peter Jackson.	The model fails to maintain topic continuity across dialogue turns, resulting in a contextually misaligned response.
	Poor Information Integration	(round-1): I love Iron Man movies, he's my favorite superhero. (round-2): Yes, I really admire his intelligence. When did he first appear? (round-3): I also like Captain America. Have Iron Man and Captain America ever worked together?	(round-1): Iron Man is indeed a key figure in the Marvel Universe. His real name is Tony Stark, a wealthy inventor. (round-2): Iron Man first appeared in 1963, and the Iron Man movie was released in 2008. Do you like any other superheroes? (round-3): Iron Man and Captain America have **never appeared in the same movie**.	The model fails to integrate prior contextual cues, resulting in a factually inaccurate and inconsistent conclusion across turns.

2.2.3. Contextual Hallucinations

Contextual hallucinations typically occur in long-form generation or multi-turn dialogues, where the model produces outputs that contradict or ignore prior context. These hallucinations reflect difficulties in retaining and integrating previous content across extended sequences [18].

Explainability Perspective. The absence of mechanisms to trace context-dependent reasoning contributes to two major explainability gaps:

- **Insufficient Context Tracking:** The model often fails to recall and incorporate relevant historical information, especially when the discourse spans multiple topics or temporal references [14,16].
- **Poor Information Integration:** Even when prior context is partially retained, the model may fail to integrate it coherently into new outputs. This leads to contradictions or incoherence across turns, particularly in extended dialogues [19,44].

3. Analyzing Hallucinations via Explainability

This chapter builds upon the three types of hallucinations introduced in Section 2.2 and systematically examines the detection, identification, and mitigation methods for each. It further analyzes how various explainability techniques are applied within these processes and highlights their effectiveness in supporting hallucination understanding and control.

 doi:10.20944/preprints202505.0456.v1

3.1. Explainability-Oriented Hallucination Detection Methods

3.1.1. Factual Hallucinations Detection

Knowledge-Aligned Detection of Data Source Errors.

Factual hallucinations often stem from training data noise or unreliable sources. A widely adopted strategy involves aligning generated outputs with structured knowledge bases (e.g., Wikidata, Google Knowledge Graph) to evaluate semantic consistency and factual grounding. These methods assess whether outputs are logically entailed by verifiable knowledge, enhancing interpretability and accountability.

A foundational approach is the RAG framework, introduced by Lewis et al. [16], which incorporates external documents during inference to support evidence-grounded generation. For factuality evaluation, semantic alignment tools such as BERTScore [46] and QAGS [47] are widely used, while entailment-based scorers like FactCC [48] and DAE [49] examine logical consistency.

As shown in [17], hallucination trajectories can be traced by combining logit behavior analysis with retrieval-consistency features. This not only supports early hallucination detection but also enhances transparency in long-form generation.

Building upon these ideas, Check-Your-Facts [33] and EVER [39] implement multi-stage pipelines that combine retrieval, consistency verification, and scoring. These systems emulate human fact-checking workflows and produce interpretable evidence paths with verifiable factual scores, providing strong post-hoc explainability.

Detecting Hallucinations in Knowledge-Blind Regions.

LLMs frequently operate under incomplete or outdated knowledge coverage. In [50], the RETRO model introduces a retrieval-based approach that supplements missing knowledge in real time. When outputs diverge from retrieved evidence, these inconsistencies signal hallucination and improve traceability.

According to [17], factual drift during generation can be modeled by analyzing variations in logit distributions across retrieved contexts. In parallel, structured fact-verification pipelines such as FEVER [51], VERIFI [52], and TrueTeacher [53] decompose outputs into claims, retrieve supporting evidence, and apply natural language inference to classify support status (e.g., “Supported”, “Refuted”, “Not Enough Info”).

These pipelines provide not only high detection accuracy but also strong interpretability through causal chains between input, evidence, and conclusion. Benchmark datasets such as TruthfulQA [54] further support the field by distinguishing misleading model completions from genuine knowledge gaps and enabling intermediate-state visualization.

Self-Consistency-Based Identification of Fabricated Hallucinations.

Fabricated hallucinations are confident but unsupported statements generated by LLMs, often in speculative or ambiguous contexts [15]. Two prominent detection strategies exist: (1) intra-model output self-consistency, and (2) entailment assessment relative to input context.

The first approach prompts the model to generate multiple candidate outputs for the same query and compares factual consistency among them. Inconsistencies reveal unreliable or speculative reasoning. SelfCheckGPT, proposed by Manakul et al. [55], identifies contradictions via sampling-based self-evaluation. This technique is especially effective in open-ended generation tasks, such as question answering and summarization. Subsequent work [44] integrates attention weights and uncertainty modeling to refine this process.

The second approach leverages natural language inference (NLI) to assess entailment relationships between generated claims and their source contexts. In [19], generated text is segmented into claims, paired with context, and labeled as “Entailment,” “Neutral,” or “Contradiction.” Atanasova et al. [23] further refine this by aligning fine-grained claim segments with their evidential counterparts to improve hallucination resolution.

 doi:10.20944/preprints202505.0456.v1

Together, these techniques support explainable, self-supervised hallucination detection without requiring external fact-checking resources. By identifying logical inconsistencies and unsupported assertions, they form a robust foundation for fabrication-aware model evaluation.

3.1.2. Logical Hallucinations Detection

Chain Construction and Reasoning Consistency.

Logical hallucinations often emerge from flawed or unsupported reasoning chains, even when individual steps appear locally coherent. To detect such issues, a prominent line of research focuses on eliciting and analyzing explicit reasoning traces.

CoT prompting is a foundational method in this direction. By encouraging models to articulate step-by-step reasoning, it enables both transparency and error localization [56]. A notable enhancement is the "Let's think step by step" prompt introduced in [57], which improves zero-shot reasoning by making intermediate steps observable. Building on this, the self-consistency decoding strategy proposed in [45] samples multiple CoT paths for the same question and compares their outputs to evaluate inference stability and expose inconsistencies.

Beyond reasoning articulation, recent approaches integrate verification mechanisms to validate each step. In [40], the Chain-of-Verification (CoVe) framework augments CoT by introducing an entailment-based verifier that evaluates the logical validity of each intermediate step. This design enables fine-grained identification of inconsistencies along the reasoning trajectory.

Other efforts model reasoning as structured inference trees. For example, the belief tree framework in [58] decomposes complex claims into hierarchically related sub-claims, allowing inference over proposition chains. Logical hallucinations are detected by identifying internal contradictions or weak entailments within the tree. This not only enhances interpretability but also enables probabilistic error attribution across reasoning nodes.

These methods collectively demonstrate that the explicit modeling and validation of reasoning chains—through CoT, entailment verification, or tree-based expansion—are essential to detecting and mitigating logical hallucinations in LLMs.

Structural Modeling and Coherence Evaluation in Multi-Step Inference.

Another manifestation of logical hallucinations arises from structural incoherence in multi-step reasoning, where intermediate transitions deviate from logical progression or violate previously established subgoals. Two complementary strategies have emerged to address this: plan-then-execute generation and graph-based coherence modeling.

The first strategy includes frameworks such as Plan-to-Execute [59] and ReAct [60], where models are guided to first generate an explicit reasoning plan before executing it sequentially. This separation allows for interpretable tracking and exposes hallucinations when execution deviates semantically from the plan. In [61], Self-Refine is introduced as an internal review mechanism that allows the model to self-edit prior steps based on consistency checks. This supports proactive correction and embeds structural validation within the generation process itself.

The second strategy focuses on transforming reasoning steps into structured representations for logic-based validation. For instance, [61] presents a consistency-aware framework that converts stepwise outputs into logic graphs and identifies contradictions or missing edges to localize reasoning failures. As shown in [62], the Graph-of-Thoughts (GoT) framework constructs dynamic reasoning graphs, where each node corresponds to an intermediate inference and must be coherently integrated into the evolving structure. This enables both backward traceability and forward consistency evaluation, supporting model-level reasoning diagnostics.

Together, these structured reasoning methods offer rich interpretability through path-level visualization, intermediate verification, and causal error tracing. They form a robust foundation for mitigating logical hallucinations through explicit modeling of reasoning coherence.

 doi:10.20944/preprints202505.0456.v1

3.1.3. Contextual Hallucinations Detection

Consistency Modeling in Multi-Turn Dialogue.

Contextual hallucinations arise when language models fail to retain or integrate prior dialogue context, leading to contradictions, omissions, or topical drift. Two main strategies have been proposed to address this: explicit consistency tracking and memory-augmented generation.

The first line of work focuses on modeling discourse continuity by analyzing topic flow, entity references, and semantic alignment across turns. In [63], a consistency-aware dialogue model is introduced, which computes alignment scores between dialogue history and candidate responses to detect context drift. Similarly, ContextQA [64] formulates context-sensitive questions based on previous content and validates generated segments using QA-style reasoning. These methods enable traceable, fine-grained localization of hallucinations while enhancing transparency in dialogue modeling.

The second strategy enhances the model's memory capacity to improve long-range dependency tracking. In [65], the kNN-LM retrieves nearest-neighbor embeddings from historical token distributions, creating an external memory to guide generation. The Memory-Augmented Transformer proposed in [66] incorporates a dynamic memory module that stores and updates latent contextual representations during generation. These architectural augmentations not only improve cross-turn coherence but also support explainability by enabling backward tracing of hallucination origins.

Together, these approaches provide a robust framework for modeling conversational consistency. By combining context-aware validation with memory-enhanced architectures, they improve the accuracy and interpretability of contextual hallucination detection.

Cross-Turn Coherence Evaluation and Structured Hallucination Assessment.

Another important detection strategy targets the evaluation of how effectively a model reuses historical content across turns or paragraphs. To this end, two complementary approaches have emerged: embedding-based coherence scoring and toolkit-assisted structured evaluation.

The first approach compares the semantic proximity between generated responses and their preceding context using embedding similarity. In [67], the Embedding Coherence Analysis method computes vector distances between dialogue history and candidate responses to identify abrupt semantic shifts. Similarly, a sliding window BERT-based similarity method proposed in [68] dynamically tracks paragraph-level coherence in document generation. These methods enable interpretable detection of semantic drift by aligning outputs with temporal context embeddings.

The second approach relies on structured toolkits that perform multi-perspective hallucination analysis. HaluEval [69], for instance, defines a multi-dimensional evaluation framework incorporating entity tracking, factual alignment, and logical coherence. It automatically identifies context-related hallucinations such as topic drift or redundant inferences. In [70], ConsistencyBench combines NLI models with multi-hop citation chain analysis to identify weakly grounded responses and contextual misalignment in dialogue. These tools produce annotated outputs and diagnostic traces, supporting systematic and explainable evaluation of contextual hallucinations.

In summary, these methods strengthen both the reliability and interpretability of contextual hallucination detection. By modeling information flow across turns and providing structured evaluation outputs, they enable fine-grained analysis of how models manage long-range dependencies while also highlighting potential points of failure.

3.2. Explainability-Driven Hallucination Mitigation Strategies

While Section 3.1 focused on explainability-oriented detection methods—functioning primarily as diagnostic tools that identify when and where hallucinations occur—this section turns to mitigation, emphasizing proactive intervention in the generative process.

Unlike detection methods, which typically rely on post-hoc analysis, mitigation approaches aim to prevent hallucinations from arising in the first place. Here, explainability is not merely an evaluative framework but a guiding principle that informs the design of generative constraints. These

 doi:10.20944/preprints202505.0456.v1

include reasoning path control, integration of external knowledge, and consistency-aware generation mechanisms.

Accordingly, this section presents a systematic review of mitigation strategies that embed explainability into model behavior. The focus is on reducing hallucination frequency, enhancing output stability, and improving the transparency and controllability of language model generation.

3.2.1. Factual Hallucination Mitigation

Factual hallucinations refer to discrepancies between generated outputs and objective reality, often resulting from biased training data, incomplete knowledge coverage, or speculative generation. Although these issues are not always rooted in explainability deficiencies, they frequently coincide with a lack of transparent attribution, unverifiable content, or inaccessible evidence chains. Accordingly, most mitigation strategies emphasize explainability-driven mechanisms, including knowledge grounding, fact verification, and attribution-aware intervention, to reduce hallucination risk during generation.

Mitigating Hallucinations from Flawed Training Data. When hallucinations arise from erroneous or biased training data, the model tends to reproduce flawed patterns without the capacity to assess their factual accuracy. To address this, verification-based correction frameworks have been proposed that compare model outputs with external references. In [39], the EVER framework introduces real-time fact validation and correction at each decoding step, enabling early detection of factual mismatches. Similarly, RARR [42] decomposes generated content into atomic factual claims, validates them via retrieval and natural language inference, and revises hallucinated segments accordingly. These systems enhance explainability by constructing traceable claim-level verification chains. Further, Doc-level Attribution Tracing [71] identifies specific clusters of training data that contributed to hallucinated outputs, offering a novel introspective lens for hallucination origin tracing.

Mitigating Hallucinations from Knowledge Blind Spots. When factual hallucinations result from knowledge limitations, such as content beyond the model's pretraining distribution, external augmentation becomes a key solution. RAG [16] has become foundational in this context, dynamically incorporating relevant documents during generation. LLMs-Augmenter [33] expands on this paradigm by retrieving supporting evidence pre-generation to structure the generation context. In [50], the RETRO model retrieves nearest-neighbor token sequences from a large corpus, enhancing factual grounding without model retraining. These approaches strengthen generation accuracy through retrieval-evidence alignment, which also improves post-hoc explainability via traceable knowledge support chains.

Suppressing Fabricated Content via Attribution-Aware Control. Fabricated hallucinations emerge when models speculate beyond the bounds of verified input, especially in open-ended contexts. Attribution-based methods aim to mitigate such behavior by identifying and controlling hallucination-prone signals during generation. In [28], a global attribution framework quantifies token-level influence across transformer layers, providing interpretability into semantic contributions. An attribution-aware gating mechanism proposed in [27] selectively suppresses misleading activation patterns before they affect output. As an extension, [72] combines attribution with uncertainty estimation to construct an explainable hallucination risk estimator. These methods embed explainability into the generation pipeline, supporting both interpretation and intervention in real time.

3.2.2. Logical Hallucination Mitigation

Logical hallucinations arise when language models fail to maintain inferential consistency or produce valid reasoning chains, particularly in complex multi-step tasks. These failures typically manifest as either insufficient reasoning ability or fractured logical sequences. To mitigate such errors, explainability-driven methods have focused on three primary strategies: explicit reasoning path construction, iterative self-correction, and real-time verification during inference.

Enhancing Reasoning Stability through Prompt Engineering and Iterative Self-Revision. CoT prompting has emerged as a foundational approach to improving reasoning interpretability [45,57].

 doi:10.20944/preprints202505.0456.v1

By encouraging models to articulate intermediate reasoning steps, CoT increases transparency and facilitates stepwise validation. To further stabilize outputs, self-consistency decoding [45] samples multiple reasoning traces and selects the majority answer, thereby mitigating stochastic variability in inference. In [61], the Self-Refine framework enables models to iteratively revisit and revise prior reasoning steps, allowing for internal correction of logical errors without external feedback. More recently, hybrid prompting with symbolic constraints [73] has been proposed to combine CoT with logic templates that guide models through structurally valid reasoning paths. This integration significantly enhances inferential robustness and reduces illogical transitions.

Repairing Broken Reasoning Chains via Structural Alignment and Intermediate Verification. When hallucinations stem from fractured reasoning chains, maintaining semantic and logical coherence across steps becomes essential. Structured reasoning paradigms such as Plan-and-Solve [59] and ReAct [60] address this by decomposing reasoning into interpretable subgoals, each validated before proceeding. These frameworks enforce alignment between inference steps, enabling early detection of contradictions. In [39], the EVER system embeds retrieval-based verification directly into the generation loop, correcting hallucinated steps through real-time consistency checks.

For more holistic modeling, the GoT framework [62] represents reasoning processes as evolving graph structures. In this setting, each node corresponds to a sub-inference and each edge encodes logical dependencies. Such graph-based reasoning enables fault localization, backward tracing, and chain-level diagnosis of hallucinated logic. Additionally, logic-guided prompting [74] integrates verifier models that preemptively filter logically inconsistent continuations before output finalization. This approach further reduces invalid reasoning by enforcing coherence constraints during generation.

Together, these methods advance the mitigation of logical hallucinations by embedding explainability into both the planning and execution phases of reasoning. Through path-level modeling, intermediate verification, and structural introspection, they improve both reasoning fidelity and model transparency.

3.2.3. Contextual Hallucination Mitigation

Contextual hallucinations arise when language models fail to preserve coherence across dialogue turns or extended text segments, resulting in semantic inconsistencies, omissions, or contradictory references. These errors typically stem from insufficient context tracking and ineffective information integration. From an explainability perspective, recent mitigation strategies focus on proactively modeling contextual continuity and embedding validation mechanisms into the generation process.

Improving Context Tracking via Memory-Augmented and Knowledge-Guided Generation. To address the challenge of maintaining long-range coherence, memory-augmented architectures and knowledge-aware representations have been proposed. In [72], the RHO framework reduces hallucinations in open-domain dialogue by encoding contextual cues through knowledge graph entities and aligning utterances with the global dialogue context. This approach enhances both local and global semantic consistency. Similarly, the Memory-Augmented Transformer in [41] incorporates latent memory modules that retrieve and update contextual states during generation, supporting continuity and interpretability through memory traceability. In [64], a decompositional prompting strategy is introduced for long-context QA, wherein inputs are broken into position-agnostic segments to improve long-range dependency modeling and traceability.

Enhancing Information Integration through Stepwise Validation and Semantic Drift Correction. When models fail to effectively integrate accessible context, the result is often semantic drift or logical conflict. To mitigate this, stepwise verification frameworks intervene during generation to ensure contextual alignment. In [40], the CoVe framework introduces validation sub-questions during generation, each linked to relevant dialogue history for structured consistency checking. FLEEK [75] compares generated facts with retrieved external knowledge in real time and applies corrective edits when inconsistencies are identified. Additionally, [63] presents a large-scale benchmark for detecting contradiction in multi-turn dialogue and proposes contrastive fine-tuning to enforce turn-level logical coherence.

 doi:10.20944/preprints202505.0456.v1

Collectively, these approaches embed explainability into the generation pipeline through explicit memory representation, intermediate verification, and semantic consistency modeling. They enhance both the accuracy and transparency of multi-turn reasoning, forming a foundation for mitigating contextual hallucinations in dialogue and long-form generation tasks.

4. Methodological Outlook, and Theoretical Reflections

4.1. Methodological Outlook

This survey has systematically reviewed techniques for detecting and mitigating hallucinations in LLMs, placing explainability at the core of both analytical and practical frameworks. By establishing an explicit conceptual link between explainability mechanisms and hallucination governance, the review categorizes existing approaches according to three primary hallucination types: factual, logical, and contextual. These approaches are analyzed through the unified lens of interpretability, encompassing both internal interpretability (e.g., feature attribution and transformer-layer analysis) and post-hoc interpretability (e.g., external knowledge integration and causal inference modeling) [76,77]. For detection methods, we emphasize the importance of maintaining observable mappings between model behaviors and hallucination phenomena. For mitigation strategies, we highlight explainability as a critical intermediary that facilitates structured interventions throughout the generation process.

From a methodological perspective, integrating explainability with structured reasoning emerges as a promising direction for future research. Techniques such as CoT, ToT, GoT, and Plan-to-Execute are increasingly recognized for their dual roles: not only improving model performance but also providing transparent inference pathways and verifiable reasoning structures. These methods facilitate both the identification and proactive mitigation of logical inconsistencies. Similarly, memory-augmented architectures, context-tracking modules, and stepwise verification methods such as CoVe contribute to progressively more interpretable and controllable representations for managing contextual coherence. Collectively, these developments signify a critical paradigm shift in hallucination mitigation, moving from outcome-based corrections toward proactive and mechanism-level transparency.

Several challenges and opportunities remain for further exploration. Despite rapid advancements, current explainability methods still require standardized evaluation metrics, increased objectivity, and improved usability to ensure practical deployment. In addition, hallucinations arising in multimodal models, domain-specific applications, and adversarial contexts currently lack comprehensive explanatory frameworks. Future research avenues include integrating multimodal explanations, developing user-centric hallucination control interfaces, and providing explicit interpretability assurances in high-stakes decision-making scenarios, thereby advancing the trustworthiness and controllability of generative systems.

It is noteworthy that many hallucination-oriented methods, although not initially designed with interpretability as a primary goal, exhibit substantial explanatory potential in practice. Future efforts could focus on further integrating task-level and behavior-level explainability, with the aim of establishing a comprehensive hallucination governance framework characterized by enhanced traceability, verifiability, and proactive intervention capabilities.

4.2. Explainability-Driven Prompt Optimization

Prompt engineering has become a critical technique for steering the behavior of LLMs. However, traditional prompt optimization often follows an empirical, heuristic-driven process, lacking transparency, interpretability, and systematic control over model reasoning. In the context of hallucination mitigation, it is increasingly recognized that explainability must serve not only as a diagnostic tool, but also as a foundational principle guiding prompt design. Embedding explainability mechanisms into prompts enables observable reasoning paths, verifiable decision processes, and controllable output behaviors, thus establishing a principled approach to reducing hallucination risks.

Recent advances in explainability-driven prompt optimization reveal three prominent strategies:

 doi:10.20944/preprints202505.0456.v1

(1) Structured Reasoning Traces for Transparent Inference. Prompts that explicitly elicit structured reasoning processes enhance the observability of model decision-making and expose potential hallucinations early. CoT prompting [56] guides models to produce step-by-step intermediate reasoning, making inferential gaps and logical inconsistencies visible. Building on this, ToT prompting [78] expands reasoning into multiple paths organized hierarchically, enabling backtracking and error localization. More recently, GoT [62] prompting explores graph-based reasoning scaffolds, further enhancing the structural interpretability of inference chains. These methods collectively demonstrate that making the reasoning process explainable within prompts significantly mitigates hallucination risks associated with skipped or unstable logic.

(2) Embedded Verification for Real-Time Self-Assessment. Another key trend involves embedding verification and feedback loops into prompt designs to proactively detect and correct hallucinations during generation. The Self-Refine framework [61] introduces iterative self-review within prompts, encouraging the model to assess and revise its own reasoning. Similarly, the EVER framework [39] integrates explicit verification steps and retrieval-based correction into the generation loop, enhancing factual consistency and traceability. These designs instantiate internal checkpoints within the prompt itself, transforming generation from a one-shot process into an explainable, self-monitoring workflow. Embedding verification thus operationalizes explainability at the prompt level, improving both model robustness and interpretability.

(3) Memory-Augmented and Context-Aware Prompting. Long-form generation and multi-turn dialogue tasks often suffer from contextual hallucinations due to inadequate historical tracking. Recent methods address this by designing prompts that integrate memory structures and context-awareness into generation. CoPrompter [79], for instance, incorporates semantic alignment mechanisms within prompts to preserve coherence across dialogue turns. In parallel, memory-augmented prompting frameworks, inspired by dynamic retrieval architectures, maintain latent representations of prior discourse to guide future responses. These techniques enhance explainability by maintaining transparent, traceable chains of information flow throughout multi-step interactions, thereby reducing hallucination arising from context drift or forgetting.

(4) Future Directions: Toward Explainable Prompt Systems. Looking forward, integrating explainability more deeply into prompt design presents several promising research trajectories:

- *Multimodal Explainable Prompting:* Developing structured, interpretable prompts that maintain logical coherence across text, image, and audio modalities.
- *Interactive User-Driven Prompt Refinement:* Building interfaces that provide real-time, interpretable feedback during prompt crafting, enabling users to dynamically adjust prompts to minimize hallucination risks.
- *Attribution-Guided Prompt Search and Optimization:* Leveraging attribution techniques to automatically identify and refine critical prompt components, creating self-explaining and hallucination-resilient prompts at scale.

In summary, explainability-driven prompt optimization shifts prompt engineering from a heuristic art toward a principled, transparent methodology for hallucination mitigation. By structuring reasoning traces, embedding verification feedback, and maintaining contextual coherence, explainable prompts offer a powerful pathway toward building more controllable, reliable, and interpretable language models.

4.3. The Value of Hallucination: A Theoretical Reflection from the Perspective of Explainability

Hallucinations in LLMs have conventionally been viewed as negative phenomena requiring complete elimination. This section proposes a dialectical theoretical reflection, reconsidering hallucinations through the lens of explainability and highlighting that they may not be unequivocally detrimental but instead possess potentially constructive aspects worthy of deeper exploration.

Hallucinations as Indicators of Explorative Creativity. Fundamentally, hallucinations represent the model's generative attempts under uncertainty or incomplete information, reflecting a form of

 doi:10.20944/preprints202505.0456.v1

explorative creativity intrinsic to imaginative and associative cognitive processes [80]. Rather than purely undesirable errors, these phenomena indicate the model's latent capacity for conceptual recombination, analogical reasoning, and knowledge transfer across contexts. Hallucinations thus serve as potential entry points into cultivating and evaluating genuine creativity in artificial intelligence, offering insights into how LLMs might generate novel ideas, formulate interdisciplinary connections, and contribute to open-ended innovation tasks.

Diagnostic Value of Hallucinations for Model Transparency. From an explainability perspective, hallucinations provide unique insights into the internal reasoning mechanisms and knowledge integration processes within language models. Rather than random errors, hallucinations reveal adaptive responses to ambiguous prompts or uncertain information scenarios [7], effectively acting as diagnostic windows into model cognition. Through systematic analysis of hallucinatory outputs, researchers can identify inherent cognitive biases, data-related deficiencies, and structural limitations in reasoning and context awareness. Consequently, hallucination analysis emerges not only as an error-detection strategy but as a powerful explanatory tool for model refinement, interpretability, and introspection.

Constructive Ambiguity in Creative and Human-Centric Tasks. In certain domains, such as creative writing, artistic generation, and imaginative storytelling, strict adherence to factual accuracy may conflict with creative objectives and user expectations. Controlled or contextually appropriate hallucinations can actively enhance user engagement, aesthetic value, and narrative coherence [81]. From this viewpoint, explainability should shift from purely suppressive strategies toward enabling selective governance and purposeful steering of hallucinatory behavior. This paradigm emphasizes the importance of user intent, contextual relevance, and interactive control, promoting a more flexible and context-aware approach to hallucination management in human-centered AI applications.

Hallucinations and the Theoretical Boundaries of Language Intelligence. The persistence of hallucinations in LLMs challenges prevailing notions of machine language understanding, underscoring fundamental distinctions between human semantic comprehension and the statistical approximations achieved by current models. Explainability, therefore, extends beyond its practical role in debugging to serve as a foundational theoretical framework, delineating the inherent limitations and capabilities of artificial cognition. Hallucinations illuminate critical cognitive, informational, and representational boundaries, offering valuable insights for foundational AI research.

This perspective advocates for a nuanced approach to hallucinations, recognizing both their detrimental effects and potential constructive applications. Future research should aim to develop integrated frameworks that balance effective mitigation strategies with context-dependent utilization of hallucinations. Navigating this duality represents a key frontier in advancing explainable and responsible artificial intelligence.

5. Conclusion

This survey highlights the pivotal role of explainability in addressing hallucinations within LLMs. By categorizing hallucinations into factual, logical, and contextual types, and analyzing detection and mitigation strategies through the lens of interpretability, we underscore the necessity of transparent reasoning pathways. Techniques such as CoT, ToT, and memory-augmented prompting not only enhance model performance but also facilitate proactive identification and correction of inconsistencies. While challenges persist—particularly in standardizing evaluation metrics and extending frameworks to multimodal and domain-specific contexts—integrating explainability into prompt design and model architecture offers a promising avenue for developing more trustworthy and controllable generative systems. Future research should continue to explore this integration, aiming to balance effective hallucination mitigation with the preservation of creative and contextually appropriate model outputs.

Author Contributions: Conceptualization, Wentao Deng, Jiao Li, Hong-Yu Zhang, Jiuyong Li, Zhenyun Deng, Debo Cheng and Zaiwen Feng; methodology, Wentao Deng and Jiao Li; writing—original draft preparation, Wentao Deng and Jiao Li; writing—review and editing, Hong-Yu Zhang, Jiuyong Li, Zhenyun Deng, Debo Cheng

 doi:10.20944/preprints202505.0456.v1

and Zaiwen Feng; visualization, Wentao Deng; supervision, Zaiwen Feng; project administration, Zaiwen Feng; funding acquisition, Zaiwen Feng. All authors have read and agreed to the published version of the manuscript.

Funding: This research project was supported in part by the Open Research Fund Program of Key Laboratory of Knowledge Mining and Knowledge Services in Agricultural Converging Publishing, National Press and Publication Administration under Grant 2023KMKS02, and the Hubei Key Research and Development Program of China under Grant 2024BBB055, 2024BAA008.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Yu, W.; Zhu, C.; Li, Z.; Hu, Z.; Wang, Q.; Ji, H.; Jiang, M. A survey of knowledge-enhanced text generation. *ACM Computing Surveys* **2022**, *54*, 1–38.
2. Poibeau, T. *Machine translation*; MIT Press, 2017.
3. Zhao, W.X.; Zhou, K.; Li, J.; Tang, T.; Wang, X.; Hou, Y.; Min, Y.; Zhang, B.; Zhang, J.; Dong, Z.; et al. A survey of large language models. *arXiv preprint arXiv:2303.18223* **2023**, *1*.
4. Zhou, G.; Kwashie, S.; et al. FASTAGEDS: fast approximate graph entity dependency discovery. In Proceedings of the International Conference on Web Information Systems Engineering. Springer, 2023, pp. 451–465.
5. Wu, T.; He, S.; Liu, J.; Sun, S.; Liu, K.; Han, Q.L.; Tang, Y. A brief overview of ChatGPT: The history, status quo and potential future development. *IEEE/CAA Journal of Automatica Sinica* **2023**, *10*, 1122–1136.
6. Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* **2023**.
7. Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; et al. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems* **2025**, *43*, 1–55.
8. Thirunavukarasu, A.J.; Ting, D.S.J.; Elangovan, K.; Gutierrez, L.; Tan, T.F.; Ting, D.S.W. Large language models in medicine. *Nature medicine* **2023**, *29*, 1930–1940.
9. Almeida, G.F.; Nunes, J.L.; Engelmann, N.; Wiegmann, A.; de Araújo, M. Exploring the psychology of LLMs' moral and legal reasoning. *Artificial Intelligence* **2024**, *333*, 104145.
10. Li, Y.; Wang, S.; Ding, H.; Chen, H. Large language models in finance: A survey. In Proceedings of the Proceedings of the fourth ACM international conference on AI in finance, 2023, pp. 374–382.
11. Kasneci, E.; Seßler, K.; Küchemann, S.; Bannert, M.; Dementieva, D.; Fischer, F.; Gasser, U.; Groh, G.; Günnemann, S.; Hüllermeier, E.; et al. ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and individual differences* **2023**, *103*, 102274.
12. Zhao, H.; Chen, H.; Yang, F.; Liu, N.; Deng, H.; Cai, H.; Wang, S.; Yin, D.; Du, M. Explainability for large language models: A survey. *ACM Transactions on Intelligent Systems and Technology* **2024**, *15*, 1–38.
13. Hassija, V.; Chamola, V.; Mahapatra, A.; Singal, A.; Goel, D.; Huang, K.; Scardapane, S.; Spinelli, I.; Mahmud, M.; Hussain, A. Interpreting black-box models: a review on explainable artificial intelligence. *Cognitive Computation* **2024**, *16*, 45–74.
14. Hagendorff, T.; Fabi, S.; Kosinski, M. Human-like intuitive behavior and reasoning biases emerged in large language models but disappeared in ChatGPT. *Nature Computational Science* **2023**, *3*, 833–838.
15. Zhang, Y.; Li, Y.; Cui, L.; Cai, D.; Liu, L.; Fu, T.; Huang, X.; Zhao, E.; Zhang, Y.; Chen, Y.; et al. Siren's song in the AI ocean: a survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219* **2023**.
16. Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.t.; Rocktäschel, T.; et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems* **2020**, *33*, 9459–9474.
17. Varshney, N.; Yao, W.; Zhang, H.; Chen, J.; Yu, D. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation. *arXiv preprint arXiv:2307.03987* **2023**.
18. McKenna, N.; Li, T.; Cheng, L.; Hosseini, M.J.; Johnson, M.; Steedman, M. Sources of hallucination by large language models on inference tasks. *arXiv preprint arXiv:2305.14552* **2023**.
19. Lei, D.; Li, Y.; Hu, M.; Wang, M.; Yun, V.; Ching, E.; Kamal, E. Chain of natural language inference for reducing large language model ungrounded hallucinations. *arXiv preprint arXiv:2310.03951* **2023**.

 doi:10.20944/preprints202505.0456.v1

20. Li, J.; Monroe, W.; Jurafsky, D. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220* **2016**.
21. Lundberg, S.M.; Lee, S.I. A unified approach to interpreting model predictions. *Advances in neural information processing systems* **2017**, *30*.
22. Feng, S.; Wallace, E.; Grissom II, A.; Iyyer, M.; Rodriguez, P.; Boyd-Graber, J. Pathologies of neural models make interpretations difficult. *arXiv preprint arXiv:1804.07781* **2018**.
23. Atanasova, P. A diagnostic study of explainability techniques for text classification. In *Accountable and Explainable Methods for Complex Reasoning over Text*; Springer, 2024; pp. 155–187.
24. Kindermans, P.J.; Hooker, S.; Adebayo, J.; Alber, M.; Schütt, K.T.; Dähne, S.; Erhan, D.; Kim, B. The (un) reliability of saliency methods. *Explainable AI: Interpreting, explaining and visualizing deep learning* **2019**, pp. 267–280.
25. Sikdar, S.; Bhattacharya, P.; Heese, K. Integrated directional gradients: Feature interaction attribution for neural NLP models. In Proceedings of the Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 865–878.
26. Enguehard, J. Sequential integrated gradients: a simple but effective method for explaining language models. *arXiv preprint arXiv:2305.15853* **2023**.
27. Ferrando, J.; Gállego, G.I.; Costa-Jussà, M.R. Measuring the mixing of contextual information in the transformer. *arXiv preprint arXiv:2203.04212* **2022**.
28. Modarressi, A.; Fayyaz, M.; Yaghoobzadeh, Y.; Pilehvar, M.T. GlobEnc: Quantifying global token attribution by incorporating the whole encoder layer in transformers. *arXiv preprint arXiv:2205.03286* **2022**.
29. Kobayashi, G.; Kuribayashi, T.; Yokoi, S.; Inui, K. Analyzing feed-forward blocks in transformers through the lens of attention maps. *arXiv preprint arXiv:2302.00456* **2023**.
30. Jaunet, T.; Kervadec, C.; Vuillemot, R.; Antipov, G.; Baccouche, M.; Wolf, C. Visqa: X-raying vision and language reasoning in transformers. *IEEE Transactions on Visualization and Computer Graphics* **2021**, *28*, 976–986.
31. Luo, H.; Specia, L. From understanding to utilization: A survey on explainability for large language models. *arXiv preprint arXiv:2401.12874* **2024**.
32. Makantasis, K.; Georgogiannis, A.; Voulodimos, A.; Georgoulas, I.; Doulamis, A.; Doulamis, N. Rank-r fnn: A tensor-based learning model for high-order data classification. *IEEE Access* **2021**, *9*, 58609–58620.
33. Peng, B.; Galley, M.; He, P.; Cheng, H.; Xie, Y.; Hu, Y.; Huang, Q.; Liden, L.; Yu, Z.; Chen, W.; et al. Check your facts and try again: Improving large language models with external knowledge and automated feedback. *arXiv preprint arXiv:2302.12813* **2023**.
34. Li, X.; Zhao, R.; Chia, Y.K.; Ding, B.; Joty, S.; Poria, S.; Bing, L. Chain-of-knowledge: Grounding large language models via dynamic knowledge adapting over heterogeneous sources. *arXiv preprint arXiv:2305.13269* **2023**.
35. Vu, T.; Iyyer, M.; Wang, X.; Constant, N.; Wei, J.; Wei, J.; Tar, C.; Sung, Y.H.; Zhou, D.; Le, Q.; et al. Freshllms: Refreshing large language models with search engine augmentation. *arXiv preprint arXiv:2310.03214* **2023**.
36. Kamesh, R. Think Beyond Size: Dynamic Prompting for More Effective Reasoning. *arXiv preprint arXiv:2410.08130* **2024**.
37. Fu, R.; Wang, H.; Zhang, X.; Zhou, J.; Yan, Y. Decomposing complex questions makes multi-hop QA easier and more interpretable. *arXiv preprint arXiv:2110.13472* **2021**.
38. Shi, Z.; Sun, W.; Gao, S.; Ren, P.; Chen, Z.; Ren, Z. Generate-then-ground in retrieval-augmented generation for multi-hop question answering. *arXiv preprint arXiv:2406.14891* **2024**.
39. Kang, H.; Ni, J.; Yao, H. Ever: Mitigating hallucination in large language models through real-time verification and rectification. *arXiv preprint arXiv:2311.09114* **2023**.
40. Dhuliawala, S.; Komeili, M.; Xu, J.; Raileanu, R.; Li, X.; Celikyilmaz, A.; Weston, J. Chain-of-verification reduces hallucination in large language models. *arXiv preprint arXiv:2309.11495* **2023**.
41. Gao, L.; Dai, Z.; Pasupat, P.; Chen, A.; Chaganty, A.T.; Fan, Y.; Zhao, V.Y.; Lao, N.; Lee, H.; Juan, D.C.; et al. Rarr: Researching and revising what language models say, using language models. *arXiv preprint arXiv:2210.08726* **2022**.
42. Pan, L.; Saxon, M.; Xu, W.; Nathani, D.; Wang, X.; Wang, W.Y. Automatically correcting large language models: Surveying the landscape of diverse self-correction strategies. *arXiv preprint arXiv:2308.03188* **2023**.
43. Rawte, V.; Chakraborty, S.; Pathak, A.; Sarkar, A.; Tonmoy, S.; Chadha, A.; Sheth, A.; Das, A. The troubling emergence of hallucination in large language models-an extensive definition, quantification, and prescriptive remediations. 2023.

 doi:10.20944/preprints202505.0456.v1

44. Zhang, T.; Qiu, L.; Guo, Q.; Deng, C.; Zhang, Y.; Zhang, Z.; Zhou, C.; Wang, X.; Fu, L. Enhancing uncertainty-based hallucination detection with stronger focus. *arXiv preprint arXiv:2311.13230* **2023**.
45. Wang, X.; Wei, J.; Schuurmans, D.; Le, Q.; Chi, E.; Narang, S.; Chowdhery, A.; Zhou, D. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171* **2022**.
46. Zhang, T.; Kishore, V.; Wu, F.; Weinberger, K.Q.; Artzi, Y. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675* **2019**.
47. Wang, A.; Cho, K.; Lewis, M. Asking and answering questions to evaluate the factual consistency of summaries. *arXiv preprint arXiv:2004.04228* **2020**.
48. Kryściński, W.; McCann, B.; Xiong, C.; Socher, R. Evaluating the factual consistency of abstractive text summarization. *arXiv preprint arXiv:1910.12840* **2019**.
49. Goyal, T.; Durrett, G. Evaluating factuality in generation with dependency-level entailment. *arXiv preprint arXiv:2010.05478* **2020**.
50. Borgeaud, S.; Mensch, A.; Hoffmann, J.; Cai, T.; Rutherford, E.; Millican, K.; Van Den Driessche, G.B.; Lespiau, J.B.; Damoc, B.; Clark, A.; et al. Improving language models by retrieving from trillions of tokens. In Proceedings of the International conference on machine learning. PMLR, 2022, pp. 2206–2240.
51. Thorne, J.; Vlachos, A.; Christodoulopoulos, C.; Mittal, A. FEVER: a large-scale dataset for fact extraction and VERification. *arXiv preprint arXiv:1803.05355* **2018**.
52. Rashkin, H.; Nikolaev, V.; Lamm, M.; Aroyo, L.; Collins, M.; Das, D.; Petrov, S.; Tomar, G.S.; Turc, I.; Reitter, D. Measuring attribution in natural language generation models. *Computational Linguistics* **2023**, *49*, 777–840.
53. Gekhman, Z.; Herzig, J.; Aharoni, R.; Elkind, C.; Szpektor, I. Trueteacher: Learning factual consistency evaluation with large language models. *arXiv preprint arXiv:2305.11171* **2023**.
54. Lin, S.; Hilton, J.; Evans, O. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958* **2021**.
55. Manakul, P.; Liusie, A.; Gales, M.J. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896* **2023**.
56. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D.; et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* **2022**, *35*, 24824–24837.
57. Kojima, T.; Gu, S.S.; Reid, M.; Matsuo, Y.; Iwasawa, Y. Large language models are zero-shot reasoners. *Advances in neural information processing systems* **2022**, *35*, 22199–22213.
58. Dziri, N.; Kamalloo, E.; Milton, S.; Zaiane, O.; Yu, M.; Ponti, E.M.; Reddy, S. Faithdial: A faithful benchmark for information-seeking dialogue. *Transactions of the Association for Computational Linguistics* **2022**, *10*, 1473–1490.
59. Wang, L.; Xu, W.; Lan, Y.; Hu, Z.; Lan, Y.; Lee, R.K.W.; Lim, E.P. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. *arXiv preprint arXiv:2305.04091* **2023**.
60. Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; Cao, Y. React: Synergizing reasoning and acting in language models. In Proceedings of the International Conference on Learning Representations (ICLR), 2023.
61. Madaan, A.; Tandon, N.; Gupta, P.; Hallinan, S.; Gao, L.; Wiegreffe, S.; Alon, U.; Dziri, N.; Prabhumoye, S.; Yang, Y.; et al. Self-refine: Iterative refinement with self-feedback. *Advances in Neural Information Processing Systems* **2023**, *36*, 46534–46594.
62. Besta, M.; Blach, N.; Kubicek, A.; Gerstenberger, R.; Podstawski, M.; Gianinazzi, L.; Gajda, J.; Lehmann, T.; Niewiadomski, H.; Nyczyk, P.; et al. Graph of thoughts: Solving elaborate problems with large language models. In Proceedings of the Proceedings of the AAAI Conference on Artificial Intelligence, 2024, Vol. 38, pp. 17682–17690.
63. Sato, S.; Akama, R.; Suzuki, J.; Inui, K. A Large Collection of Model-generated Contradictory Responses for Consistency-aware Dialogue Systems. *arXiv preprint arXiv:2403.12500* **2024**.
64. He, J.; Pan, K.; Dong, X.; Song, Z.; Liu, Y.; Sun, Q.; Liang, Y.; Wang, H.; Zhang, E.; Zhang, J. Never Lost in the Middle: Mastering Long-Context Question Answering with Position-Agnostic Decompositional Training. *arXiv preprint arXiv:2311.09198* **2023**.
65. Khandelwal, U.; Levy, O.; Jurafsky, D.; Zettlemoyer, L.; Lewis, M. Generalization through memorization: Nearest neighbor language models. *arXiv preprint arXiv:1911.00172* **2019**.
66. Wu, Q.; Lan, Z.; Qian, K.; Gu, J.; Geramifard, A.; Yu, Z. Memformer: A memory-augmented transformer for sequence modeling. *arXiv preprint arXiv:2010.06891* **2020**.

 doi:10.20944/preprints202505.0456.v1

67. Lee, S.; Park, S.H.; Jo, Y.; Seo, M. Volcano: mitigating multimodal hallucination through self-feedback guided revision. *arXiv preprint arXiv:2311.07362* **2023**.
68. Cui, P.; Hu, L. Sliding selector network with dynamic memory for extractive summarization of long documents. In Proceedings of the Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2021, pp. 5881–5891.
69. Li, J.; Cheng, X.; Zhao, W.X.; Nie, J.Y.; Wen, J.R. Halueval: A large-scale hallucination evaluation benchmark for large language models. *arXiv preprint arXiv:2305.11747* **2023**.
70. Cui, L.; Wu, Y.; Liu, S.; Zhang, Y.; Zhou, M. MuTual: A dataset for multi-turn dialogue reasoning. *arXiv preprint arXiv:2004.04494* **2020**.
71. Tonmoy, S.; Zaman, S.; Jain, V.; Rani, A.; Rawte, V.; Chadha, A.; Das, A. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313* **2024**, *6*.
72. Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y.J.; Madotto, A.; Fung, P. Survey of hallucination in natural language generation. *ACM computing surveys* **2023**, *55*, 1–38.
73. He, Q.; Zeng, J.; He, Q.; Liang, J.; Xiao, Y. From complex to simple: Enhancing multi-constraint complex instruction following ability of large language models. *arXiv preprint arXiv:2404.15846* **2024**.
74. Ling, Z.; Fang, Y.; Li, X.; Huang, Z.; Lee, M.; Memisevic, R.; Su, H. Deductive verification of chain-of-thought reasoning. *Advances in Neural Information Processing Systems* **2023**, *36*, 36407–36433.
75. Bayat, F.F.; Qian, K.; Han, B.; Sang, Y.; Belyi, A.; Khorshidi, S.; Wu, F.; Ilyas, I.F.; Li, Y. Fleek: Factual error detection and correction with evidence retrieved from external knowledge. *arXiv preprint arXiv:2310.17119* **2023**.
76. Pearl, J. *Causality*; Cambridge university press, 2009.
77. Cheng, D.; Li, J.; Liu, L.; Liu, J.; Le, T.D. Data-driven causal effect estimation based on graphical causal modelling: A survey. *ACM Computing Surveys* **2024**, *56*, 1–37.
78. Yao, S.; Yu, D.; Zhao, J.; Shafran, I.; Griffiths, T.; Cao, Y.; Narasimhan, K. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems* **2023**, *36*, 11809–11822.
79. Joshi, I.; Shahid, S.; Venneti, S.M.; Vasu, M.; Zheng, Y.; Li, Y.; Krishnamurthy, B.; Chan, G.Y.Y. CoPrompter: User-Centric Evaluation of LLM Instruction Alignment for Improved Prompt Engineering. In Proceedings of the Proceedings of the 30th International Conference on Intelligent User Interfaces, 2025, pp. 341–365.
80. Sriramanan, G.; Bharti, S.; Sadasivan, V.S.; Saha, S.; Kattakinda, P.; Feizi, S. Llm-check: Investigating detection of hallucinations in large language models. *Advances in Neural Information Processing Systems* **2024**, *37*, 34188–34216.
81. Rudolph, J.; Tan, S.; Tan, S. ChatGPT: Bullshit spewer or the end of traditional assessments in higher education? *Journal of applied learning and teaching* **2023**, *6*, 342–363.