Beyond the Prompt: Approaches to Knowledge Boundary Detection in Large Language Models - A Systematic Literature Review

Brandon Colelough[a, b,*,1], **Davis Bartels**[a,2], **Dina Demner-Fushman**[a,2], **Ishan Tamrakar**[b,2], **Kwesi Cobbina**[b,2], **Mike Ledford**[b,2], **Srividya Ponnada**[b,2], **Xinchen Yang**[b,2] **and Yuexi Chen**[b,3]

[a]National Library of Medicine, National Institutes of Health, HHS
[b]Department of Computer Science, University of Maryland, College Park, MD, USA

Abstract. Large Language Models (LLMs) have become increasingly prevalent in critical domains, such as clinical decision-making. Yet, these models frequently produce incorrect or fabricated outputs referred to as hallucinations, which pose a serious danger in high-risk applications such as healthcare. Determining the knowledge internally held by a generative model, such as an LLM, without adding external data remains a significant and unresolved challenge. This review systematically maps the current architectures and methods designed to detect and determine the knowledge boundary of a generative system without providing external sources of information to the system. An initial retrieval of 5,919 articles was independently screened by a team of nine reviewers through a dual independent screening process, narrowing the literature down to 199 relevant papers. Pilot data extraction sheets were then used to capture study data and further scope the relevant literature to 43 eligible studies. A convergent, segregated synthesis process was performed which revealed six main clusters of research. The included studies revealed significant gaps between what models internally know and how effectively they use that knowledge. Overall, we found that the effective internal detection of an LLM's knowledge boundary significantly depended on identifying reliable internal signals, applying thoughtful boundary-aware methods, and optimizing model prompts. This review provides the first detailed synthesis of introspective approaches to knowledge boundary detection, highlighting pathways toward more reliable LLMs in critical decision-making domains such as healthcare.

1 Introduction

Large language models now pervasively, though contentiously, sit at the center of what can be high-stakes decision pipelines, yet there is a lack of mechanisms that offer the ability to explicitly map what is genuinely "known" by these models. Humans, when asked to justify confidence, can appeal to metacognitive frameworks such as the Johari window to disclose "known knowns" while flagging blind spots, or tools such as the Rumsfeld's matrix, famous for the introduction of "unknown unknowns". These devices let us express the boundaries of our knowledge before mistakes we as humans make can solidify into negative outcomes. LLMs possess no such native mechanism, and as such, their confidence and often authoritative responses mask a serious issue of real uncertainty in their ability to provide truthful answers, as they may hallucinate responses (see [1] for more on this). To overcome the hallucination problem, there is a growing body of work seeking mechanisms by which an LLM can delineate its knowledge boundary using only signals already latent in the network (embeddings, activations, logits, etc.) rather than prompts, retrieval corpora, or other exogenous aids. Central to this body of work is the concept of introspective knowledge boundary identification, which is to say that the mechanism used to determine whether or not a model truly "knows" something should not utilize external knowledge from the model so as not to contaminate the testing environment for boundary identification. For this survey, we first define *unbounded knowledge*, *bounded knowledge*, and *domain knowledge* as follows. Unbounded knowledge, denoted as K, refers to the infinite set of all possible knowledge, including knowledge that may be fundamentally unknowable or unverifiable (i.e. the "unknown unknown"), bounded knowledge, represented as $K_{\text{bounded}} \subset K$, is the finite subset of this space that is, in principle, knowable (the "known known" to the "unknown known") and finally, domain knowledge, denoted $K_{\text{domain}} \subseteq K_{\text{bounded}}$, is the portion of bounded knowledge that pertains to a specific topic or field (e.g. the knowledge contained within a medical textbook etc.). A *knowledge boundary* can hence be defined as the set difference $K_{\text{bounded}} \setminus K_{\text{known}}$, representing the frontier between what is currently known by an agent or collective and what remains knowable but not yet acquired. A *model knowledge boundary* may be observed as the set difference $K_{\text{domain}} \setminus K^{\text{model}}_{\text{known}}$, capturing the gap between the domain-specific knowledge available to a large language model and the subset it has actually internalized through training. Since large language models are trained on curated domain-restricted corpora, their knowledge boundary exists strictly within the confines of $K_{\text{domain(s)}}$. Motivated by this gap, we conduct the first (to our knowledge) systematic literature review that collates, taxonomises, and appraises architectures, algorithms, and methodologies intended to locate an LLM's knowledge limits without providing external information. Our review starts by examining key architectures presently available that attack the problem of determining the knowledge boundary of a model whilst still requiring external information, and then extrapolating on current work that may

* Corresponding author. Email: brandon.colelough@nih.gov.
[1] First author. Led the systematic literature review.
[2] Second author. Contributed to screening, extraction, and reporting.
[3] Third authors. Participated in screening and extraction.

prove useful to reaching the goal of introspective knowledge boundary identification within generative models. Our review maps where internal-only signals succeed, where they silently fail, and how these patterns inform safer model deployment.

2 Methods

This systematic literature review followed a protocol grounded in the JBI manual and reported in line with PRISMA. Eligibility criteria for this survey hinged on three positive requirements, which were (i) the study probes an LLM's internal knowledge without external retrieval, (ii) it formalizes or evaluates that knowledge, and (iii) it is peer-reviewed and English-language. A structured exclusion hierarchy was also utilized, which excluded papers if they were a literature review, dealt with non-generative models, external-knowledge pipelines, or were non-research formats. A PRESS-validated search string was executed across nine bibliographic databases (Scopus, Web of Science, PubMed, IEEE Xplore, ACL Anthology, ACM DL, Springer Link, arXiv, Google Scholar) and key grey-literature portals (OpenAI, DeepMind, Microsoft, conference workshops, etc), with no date restrictions and citations managed in Covidence. Two reviewers independently screened titles/abstracts, piloting on 100 records, with disputes resolved by a third assessor where necessary; full texts received a single review during the data extraction phase. The process scoped the 5,919 initial records to 199 papers for full-text extraction. Finally, post full-text extraction, a total of 43 eligible studies were selected to be reported on. The data extracted from the 199 eligible papers were charted using a piloted extraction form that captured publication details, model family, boundary signal, evaluation dataset, performance metrics, and declared limitations. Each paper was then tagged under an evidence-classification schema (looking at architecture characteristics such as embedding-space probing, neuron editing, decoding-dynamics confidence, graph-based extraction, refusal-grammar heuristics, or hybrid) and assigned credibility flags for components such as missing baselines, dataset leakage, or opaque hyperparameters, etc. Synthesis followed a convergent segregated design wherein quantitative results (e.g., metrics reported on within the literature) were summarised descriptively, with meta-analysis attempted when at least three homogeneous studies were available. Qualitative findings were reported on as authors' reflections on conceptual framing, failure modes, and deployment context, and this modality underwent JBI meta-aggregation, generating higher-order statements per evidence class. Finally, a thematic analysis was constructed to align the numerical and narrative strands into six research domains All search strategies, extraction sheets, and synthesis scripts are openly archived to facilitate replication and future updates.

3 Results

This section presents a consolidated view of the empirical findings from the studies included in this systematic review. The results are organized based on their primary task, and quantitative metrics are assessed within the six domain groupings explored below in the discussion. An overarching presentation of the empirical findings across all studies is presented in Table 1. The choice of quantitative metrics reported for the datasets and model architectures explored varied based on each study's research goal, with many studies gravitating on common metrics such as accuracy, precision, F1, ROUGE-L, BERTScore, QuestEval, Expected Calibration Error (ECE), Mean Rank, hit@n, and additional measures such as consistency, fluency, coverage, diversity, and human acceptance. The models presented throughout the accepted literature were assessed on numerous datasets, ranging from structured knowledge bases like ATOMIC, ConceptNet, TransOMCS, Wikidata, FB15k-237, and Wiki27K, to benchmarks for factual and commonsense reasoning (e.g., TruthfulQA, CommonsenseQA, StrategyQA), language understanding (e.g., MMLU, BoolQ, CoNLL-2003, OpenWebText), and text generation (e.g., XSum). The clusters 2, 5, and 6 focus on literature that gives insights into how LLMs internally store, manipulate, and assess factual knowledge, and provide strategies to better understand and regulate model behavior. Literature in Cluster 2 highlights that factual and relational associations are encoded in structured, linearly decodable forms within LLMs [15, 18]. The methods explore ways to trace, edit, or manipulate specific knowledge with minimal inference within attention heads, MLP Layers, and sparse feature spaces [14, 19]. These experiments reveal that knowledge circuits and propositional states are present and interpretable [16]. With literature in cluster 5, we see targeted edits to small neuron subsets can meaningfully change model behavior, showing neuron-level interpretability. PMET [34] highlights the feasibility of the technique with precise model-level interventions and improves factual edit success from 76% to 92%, preserving fluency. Literature in cluster 6 shows that internal signals such as uncertainty, consistency, and supportiveness can serve as indicators of knowledge boundaries. PGDC [41] and SKR [42] improve factual precision and reduce hallucinations, and LLMMaps [37] shows that misalignments between internal representations and final outputs can be effectively quantified using activation-based metrics and consistency cues.

4 Discussion

4.1 Strategies for Evaluating Knowledge Fidelity and Boundaries in LLMs

LLM knowledge-boundary evaluation asks whether a model can recognize and openly signal the edge of what it truly knows. Take an LLM trained only on a Wikipedia snapshot that ends in 2019, when asked, "Which city hosted the 2016 Summer Olympics?" the model should answer "Rio de Janeiro" with high confidence because that fact lies inside the frozen corpus. By contrast, when queried with "Who won the 2024 U.S. presidential election?" (knowledge that post-dates its training), the model should abstain or express uncertainty, doing so by drawing solely on epistemic signals such as elevated token-level entropy or shallow logit margins. For an effective evaluation method, we seek to ensure that external search is forbidden because it would collapse the very boundary we are trying to map: the space of all possible knowledge is effectively unbounded, and letting the model fetch fresh information would turn evaluation into an open-ended scrape of ground-truth sources. By isolating the model, we confine the task to the bounded subset K_{model} already encoded in its parameters; once that frontier is charted, users can trust when the model answers and when it confidently abstains. Figure 1 situates this challenge: the domain's fact set K_{domain} overlaps only partly with K_{model}, and our probes trace the line where confident responses must give way to principled abstention.

External QA benchmarks remain the dominant probe to elicit a knowledge boundary from a model, yet they leak information into the evaluation loop and blur the distinction between genuine self-assessment and plausible hallucinations. A benchmark-free protocol would avoid such leakage, forcing the model to rely solely on its own statistical indicators of confidence and ignorance, thereby reducing the risk of persuasive but incorrect answers or hallucinations.

Ref.	Task Focus	Representative Methods	Models / Architectures	Evaluation Metrics	Benchmarks / Datasets	Key Quantitative Findings
Cluster 1: Bridging Knowledge Graph Completion and LLM-Driven Extraction						
[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]	• Symbolic knowledge graph construction or tuple completion from LLMs using probing, alignment, or structured prompting	• contrastive info bottleneck, ridge regression • ROME, MEMIT, • KG crawling and editing • Tuple or ontology extraction, • contextual semantics, cloze statements	• BERT, RoBERTa, DistilBERT, mBERT • GPT-2/3/3.5, T5, Mistral, LLama • PanGu-Alpha, BLOOM, VKGC, Zephyr • Feedforward Neural Network, Graph Neural Network, RoBERTa-Net • Pythia, COMET, PKGC, XLM-R	• MRR; Hits@k, EM, SR(l1,l2) • Graph2Vec, GED, P@k, MAP, Spearman's rho • F1, P@k, Precision@1, Accuracy, • CKE, CKC, POS, RSA, AUC-PR • E[NCL], E[NTL]	• Wiki27K, FB15K-237-N, Wikidata • CLIKE, LAMA (SQuAD, Google-RE), MultiSimLex • BATS, WiQueen, ConceptNet, E-KAR, • MTurk, KGT5, WN18RR • MCGM, LSIM, BLI, CLEF	• VKGC improves link prediction under OWA/CWA • LLMs align with KG embeddings at scale (e.g. GPT-3 P@50 > 60%) • 150% EM boost with script alignment; • BertNet extracts 400+ novel relations at high accuracy
Cluster 2: Interpreting & Modulating Relational Knowledge						
[13, 14, 15, 16, 17, 18, 19]	• Factual and relational knowledge representation • Linear decoding and probing • Circuit and neuron-level interpretability • Knowledge editing	• Linear probing • Attention/MLP circuit tracing • Remedi vector editing • Codebook quantization • ACDC pruning	• BERT, RoBERTa • GPT-2, GPT-J • LLaMA, Mistral	• Attribution % • Fact edit success % • Code purity % • Circuit fidelity • Probe AUC	• LAMA, Google-RE, SQuAD • PaRaRel, MedNLI • Synthetic logic tasks	• 30–70% facts decodable via neurons/circuits • Remedi edits +15% accuracy • Probes yield +40% QA improvement • Codebooks: 97.1% purity • Circuits: 100% edge fidelity
Cluster 3: Detecting & Mitigating Hallucinations						
[20, 21, 22, 23, 24, 25, 26]	• Detecting & mitigating hallucinations via probing, contrastive decoding & attention-head interventions	• Gradient attribution • Attention-head constraining • Probing classifiers • Entropy-tuned decoding • Contrastive masking	• LLaMa 2/3 (7B-70B) • Mistral-7B • Qwen2 • GPT-J • BERT	• AUC, F1, macro/micro accuracy • EM, BERTScore, ECE • factKB, RMSE, MRR • NLI, QuestEval • Perplexity, Exposure	• TruthfulQA, TriviaQA, StrategyQA, BoolQA • PopQA, NQ, FACTOR, MMLU • HaluEva, FAITH, GSM8K, CommonsenseQA • MemoTrap, MuSiQue, HotpotQA, OpenBookQA • IFEval, SuperNI, Movies, Winogrande	• DeCoRe +5.5% EM (NQ-Swap) & +18.6% (XSum) • DoLa +12–17% on TruthfulQA • FAITH constrains 1% heads → +20% false-premise accuracy • InternalInspector +20.4% AUC, +8.9% ECE
Cluster 4: Commonsense KGs & Concept Extraction						
[27, 28, 29, 30, 31]	• Constructing/adapting symbolic & neural commonsense KGs for multilingual & concept-aware reasoning	• COMET, CN-COMET • CALM, C2S • COR, concept-probing • masked-LM filtering • bootstrapped generation	• T5, mT5 • GPT2-XL, GPT-3 • BART • BERT, RoBERTa	• Accuracy • BLEU, ROUGE • Meteor, CIDEr • BERTScore • Human Judgments	• ATOMIC20^{20}, KBP37 • CN-AutoMIC, SemEval2010, • CommonsenseQA, OpenBookQA • PIQA, ANLI, FewRel80 • CommonGen, C3KG	• COMET(BART) outperforms GPT-3 despite 400× fewer parameters; • CN-AutoMIC achieves 87.2% acceptance vs 38.7 for translated ATOMIC-zh% • CALM +2.9% PIQA and +1.5 BLEU on CommonGen with only 20k parameters
Cluster 5: Probing & Editing Knowledge Neurons						
[32, 33, 34, 35, 36]	• Interpret and rank knowledge neurons • Edit neuron activations to update facts • Understand layer-wise attribution	• Neuron ablation, PMET • Gradient localization, activation edits • Neuron importance scoring	• BERT, RoBERTa • GPT-2, GPT-J • LLaMA, T5	• Fact edit success rate • Hallucination rate • Retention %, edit locality • Neuron recall / precision	• LAMA, SQuAD, MedNLI • PaRaRel, Google-RE • Synthetic counterfactuals	• PMET: 76% → 92% local edit success • 85% of outputs altered with neuron edits • Salient neurons >70% consistency • Minimal performance drop
Cluster 6: Evaluating Knowledge Fidelity & Boundaries						
[37, 38, 39, 40, 41, 42]	• Detect model uncertainty and boundaries • Improve calibration and refusal behavior • Evaluate internal supportiveness	• PGDC prompt search • CoKE, SKR, ECE probing • Consistency-based signal filtering	• GPT-2, GPT-J, ChatGPT • InstructGPT, Med-PaLM • Vicuna, LLaMA-2	• Accuracy, ECE, Saware • Supportiveness score • Hallucination and refusal rate	• TriviaQA, NQ, PopQA • MedQA, PubMedQA • PARAREL, CFACT, KAssess	• PGDC: +4–5% recall, 0% hallucinations • CoKE: +15–20% precision on boundary detection • SKR: +3–10% factual EM gain • LlamaCare: +19% MedQA accuracy • ECE gaps exceed 20% for verbal confidence

Table 1. Quantitative summary of interpretability and knowledge-representation work on LLMs.

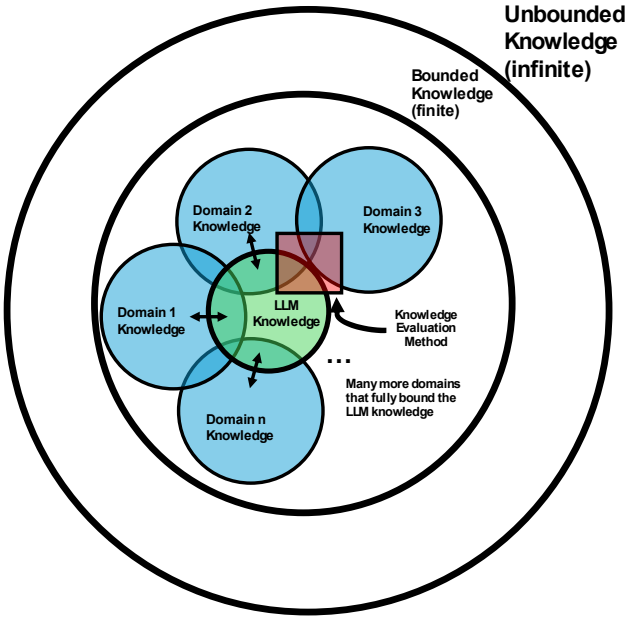

Figure 1.
Conceptual visualization of the knowledge boundary problem in LLMs. The green core represents the LLM's knowledge, which overlaps with various bounded knowledge domains (blue). Evaluation methods (red square, e.g., PGDC, COKE, etc) operate at the intersection of LLM knowledge and these domains to classify what the model knows from within a bounded domain of bounded knowledge, does not know, or mistakenly believes it knows. The larger circle boundaries indicate the finite space of bounded knowledge (known knowns to unknown knowns) to the infinite space of unbounded knowledge (unknown unknowns).

Recent work shows that purely introspective evaluation is achievable in principle, yet every current method still leans (sometimes only slightly) on external information in some form. Gradient-based techniques illustrate the promise. PGDC [41] treats boundary detection as a prompt-embedding optimization task: by measuring how far a prompt must be nudged for the model to move from an incorrect to a correct answer, it assigns each query to one of three epistemic zones (Prompt-Agnostic, Prompt-Sensitive, or Unanswerable). Because the signal is a gradient taken inside the model, the approach is architecture-agnostic and largely self-contained, yet it still requires a labeled question set to judge when the "correct" region is reached. Probability-based methods push the same idea with token-level statistics. COKE [42] relies on logit gaps and entropy to map responses onto the Rumsfeld matrix (known knowns, known unknowns, etc.), while Ni et al. [39] show that such numeric uncertainty is more honest than the model's own verbal confidence claims. Both papers, however, calibrate their thresholds on external QA benchmarks, so the evaluation loop is not fully sealed. Other lines of work convert uncertainty into action. SKR [40] first asks the model to predict, from its internal scores, whether it already knows an answer; only when the self-assessment is negative does the system trigger retrieval, which operationalizes the boundary; however, the classifier is still trained on labeled known/unknown examples. LLMMaps [37] organizes thousands of questions into a textbook-style hierarchy and then visualizes the model's accuracy at every node, exposing fine-grained blind spots. The hierarchy itself, however, is produced by ChatGPT, adding a mild external dependency and the architecture requires a QA benchmark for the questions required to achieve the hierarchy. Finally, XTEVAL [43] quantifies the "knowledge utilization gap" by checking whether facts a model can recall zero-shot also remain useful in downstream reasoning tasks; its diagnostic fact list, drawn from Wikipedia, again imports outside content.

The open problems within this domain are twofold: (i) design probes that use no external text yet still stress every corner of the K_{model}, and (ii) develop calibration techniques that let models construct their own Rumsfeld matrix—known knowns, known unknowns, unknown knowns, unknown unknowns—directly from epistemic signals. Progress here would offer a principled route to mitigating real-world failures such as unchecked hallucinations.

4.2 Bridging Knowledge Graph Completion and LLM-Driven Knowledge Extraction

There has recently been a great focus on methods, architectures, and frameworks designed to unify the symbolic reasoning of knowledge graphs (KGs) with the latent knowledge capacities of LLMs. A key motivation in this area is to overcome the brittleness of traditional KG completion models by leveraging the generalization abilities and contextual richness of LLMs. At the same time, it seeks to constrain and guide LLM outputs to maintain symbolic consistency, avoid hallucinations, and preserve ontological integrity. Figure 2 visualizes this domain as a sequential pipeline, where model generations are gated through a series of symbolic checks before incorporation into the graph.

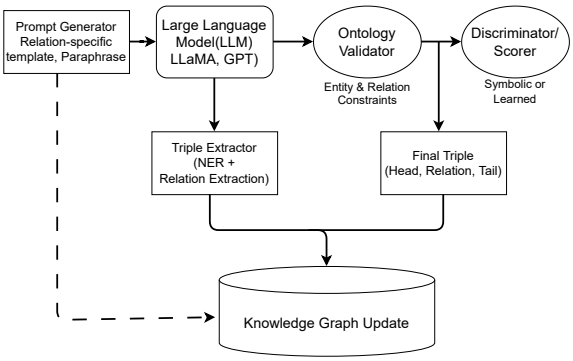

Figure 2.
A unified pipeline illustrating how LLMs can assist knowledge graph construction. Prompts are used to elicit candidate triples from the model. These triples are processed via extraction and validation modules that ensure alignment with ontology constraints. Only verified triples are used to update the symbolic KG.

Several hybrid techniques have been proposed by researchers with each of these works, attempting to align neural and symbolic representations of knowledge. Prompt-based pipelines such as LM-CRAWL [11] recursively elicit triple-style completions from language models, incorporating paraphrasing routines and "don't know" gates to control depth and precision. BERTNet [3] and Karmakar et al. [10] construct relational templates for knowledge extraction from domain texts such as building codes. However, all such systems are prompt-sensitive and require strong post-processing heuristics.

Other work has focused on representational alignment through co-trained or fused embeddings. VKGC [2] introduces a dual contrastive learning approach that is based on a variational information bottleneck to improve open-world Knowledge Graph Completion generalization. PLCS [6] combines prompt-generated triples with subgraph-level context representations for improved inductive relation prediction. Swamy et al. [8] show that symbolic triples can be extracted

from LLMs using cloze statements across training checkpoints to construct time-sliced KGs. Cross-lingual studies from Ifergan et al. [12] also reveal that even when LLMs produce consistent multilingual outputs, they may not encode unified internal representations, complicating multilingual KG alignment. Symbolic Knowledge Distillation (SKD) [7] applies discriminator-based filtering to the LLM-generated facts. This ensures that only high-quality schema-aligned triples are retained. In contrast, prompt consistency frameworks like the ICL-based KG generation approach by Khorashadizadeh et al. [5] use a multi-step generation pipeline where LLM outputs are scored and filtered based on consistency across multiple prompts to achieve the same goal.

A central research problem in this space is designing an LLM-to-KG interface that allows for high-recall, low-hallucination extraction of structured relational data while enabling downstream reasoning on symbolic graphs. Strategies span from prompt design to adversarial loss functions, but there is no consensus yet on a universal solution. Some trade-off flexibility for precision; others focus on inductive generalization under data sparsity. To further conceptualize this, we consider an example scenario, where we consider a KG of historical figures missing the triple (`Ada Lovelace`, `contributed_to`, `?`). An LLM is prompted with "What is Ada Lovelace known for?" and replies "The Analytical Engine." A good system would normalize this into a KG-compatible triple and validate its fit with the ontology. However, the LLM might hallucinate or respond with inconsistent tail entities. Solving this requires the pipeline shown in Figure 2 (wherein the system conducts prompt generation, triple extraction, relation typing, and final graph integration) supported by an alignment mechanism that can effectively mitigate against LLM generated hallucinations progressing into the KG, possibly supported by known KGs or symbolic rules but more likely by a human-in-the-loop mechanism.

4.3 Interpreting and Modulating Relational Knowledge in Transformer Models

LLMs are known for storing a vast amount of parametric knowledge, including facts about how different entities (e.g., people, places, objects) are related to one another - for example, "Paris is the capital of France". Understanding how LLMs internally represent, retrieve, and manipulate such relations is important for improving model interpretability, trustworthiness, and controllability. Recent research has increasingly focused on identifying where this knowledge lives, how it flows through the model, and how it can be monitored or modified.

One line of research focuses on locating and isolating relational information within LLMs. Wang et al. [13] showed that certain hidden states within LLMs can solely express relation concepts without absorbing other entity concepts, and can thus be regarded as relational representations. Complementing this, Hernandez et al. [14] showed that many relational mappings can be approximated by a single linear transformation on subject representations. Geva et al. [15] tackled from a different angle by investigating *how* relations are retrieved internally at inference-time instead of *where* they are stored, revealing two pivotal points for information propagation and a three-step internal mechanism for attribute extraction. Moving beyond tracing information flow, Feng et al. [16] proposed *propositional probes*, a method for extracting lexical concepts directly from token activations, allowing models' latent world states to be monitored even under adversarial prompts like backdoors. Lastly, a second study by Hernandez et al. [17] introduced REMEDI, an in-context knowledge editing technique for mapping natural language statements to fact encodings within a model's internal representation system, offering a lightweight alternative to model retraining.

These advances point toward a broader goal: transforming relational knowledge inside LLMs from opaque activations into interpretable and manipulable structures. Consider a model that answers "Tokyo is the capital of ___" with "Japan". Instead of accepting this as a black-box response, it is important to uncover how the model internally represents the relation "is the capital of" and applies that relation across various subjects like "Tokyo" and "Paris", and how such patterns can be transferred and monitored. This involves identifying latent structures that encode the relational knowledge, tracing how the relational knowledge is composed and propagated at inference-time, and making necessary edits to change model behavior, e.g. steering the model to output "France" instead of "Germany" if it mistakenly answers "Berlin is the capital of France." Ultimately, these techniques push LLMs toward being systems whose factual behavior can be understood, audited, and corrected with precision.

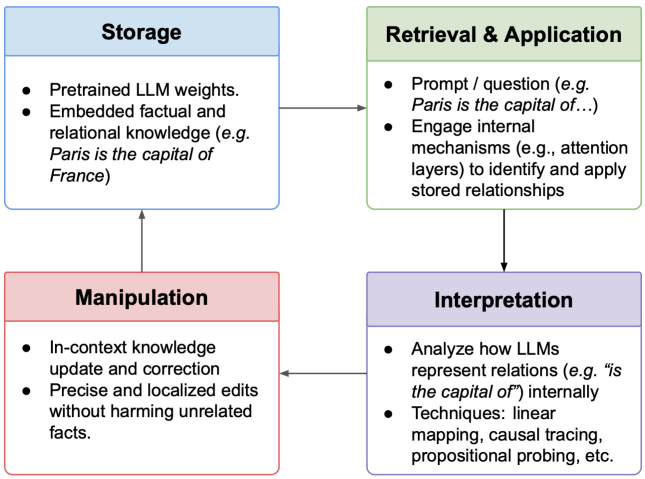

Figure 3.
Lifecycle of relational knowledge in LLMs, including storage, retrieval & application, inpterpretation and manipulation.

4.4 Leveraging Internal States and Decoding Strategies to Detect and Mitigate LLM Hallucinations

Leveraging the internal states of LLMs refers to approaches that draws on the hidden states, attention patterns, and decoding dynamics to detect hallucinations and guide generation. Being able to generate verifiably correct outputs means the model can self-identify its own knowledge boundary. Techniques such as internal probing[21][20], attention-head suppression[23], retrieval head masking[24], probing classifiers[25] and contrastive decoding[26] reveal promising directions to have advanced the field to better understand where hallucination occurs and ways to mitigate them.

LLMs often generate confident but incorrect outputs - even when their internal representations encode the correct answer [25]. This misalignment between internal knowledge and external behavior poses a fundamental challenge: how can we exploit internal states to identify hallucination risk and guide the model toward more factual outputs? The problem is exacerbated in subtle settings, such as false-premise queries[23]. Therefore, it is paramount to develop tech-

niques that are robust in any of these situations to have safe and confident LLM generations. The solution may lie in the various probing techniques deployed by these papers to tackle hallucinations as shown in Figure 4. Suppose a model is asked: *Q1: "When did Einstein win the Nobel Prize in Physics?"* → "1921"; but when asked: *Q2: "Why did Einstein win the Nobel Prize in 1920?"*, it replies confidently: "For the photoelectric effect in 1920." Although the model stores the correct year (1921), it hallucinates under a subtly false premise. Whispers that Shake Foundations [23] shows this arises from "false premise attention heads" that distort knowledge retrieval. They deploy FAITH to identify and suppress a small set of harmful attention heads. Probing estimators may be deployed in some select MLP layers [20] or across all internal states [21]. Others contrast probes on retrieval heads[24] to come up with a conditional entropy score. According to LLMs know more than they show[25], truthfulness information is encoded in the correct answer token, so they probe there. DoLA[26] relies entirely on the contrasted logits projected from the later layers of the transformer model to amplify factual knowledge. These techniques come with their own limitations. For example, DeCoRe cannot be applied to encoder-only models, "Whispers" and "LLMs Know More Than They Show" do not generalize across datasets or tasks, and DoLA cannot correct misinformation acquired during pre-training. Despite their limitations, these works push toward LLMs that can introspectively assess and improve the factual integrity of their own outputs.

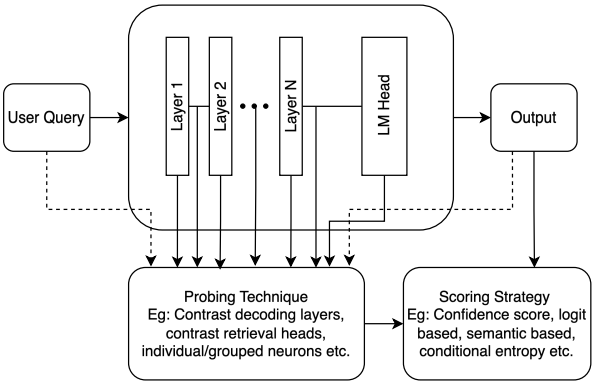

Figure 4.
An overview of the general pipeline followed by various methods for detecting and mitigating hallucinations in LLMs. The placement of the probes can be anywhere within the LLM and input/output tokens depending upon the probing technique being used

4.5 Advances in Commonsense Knowledge Graphs and Concept Extraction

Commonsense knowledge enables LLMs to understand and generate human language in a natural and contextually appropriate manner. Unlike factual knowledge, commonsense knowledge often includes inferential, intuitive, and often culturally grounded associations about the physical world, social behaviors, and everyday events. This type of knowledge is essential for various natural language processing (NLP) tasks such as question-answering (QA) and dialogue systems. Because commonsense knowledge is often implicit and rarely stated directly in textual data used to train LLMs, developing methods to help LLMs with the ability to acquire and apply such knowledge has sparked interest from the research community.

Recent advances in large pre-trained language models (PLMs) have opened up new avenues for both storing and applying commonsense knowledge, whether by encoding it implicitly in model parameters, prompting the models directly, or distilling their internal representations into structured resources such as knowledge graphs. Zhou et al. (2020) [31] presented a pretraining method, concept-aware language model (CALM), which enhances commonsense knowledge by packing it into the parameters of a transformer-based model without relying on external knowledge graphs and enables the model's better performance on natural language understanding (NLU) and generation (NLG) tasks. Fang and Zhang (2022) [30] introduced a data-efficient concept extraction method from the internal representation of PLMs for commonsense explanation generation tasks; for example, given the counter-commonsense claim "The school was open for summer," the method prompts the model to extract commonsense concepts like "vacation" and "holiday" from it to support explanation generation. Additionally, efforts have also extended to multilingual settings in this field. For example, CN-AutoMIC [27] is a Chinese Commonsense knowledge graph (CKG) extracted from mT5-XXL, one of the biggest publicly available multilingual PLMs, and is found to surpass the direct translation version of similar English CKGs in quality and diversity.

LLMs often appear to "know" a lot of commonsense knowledge, i.e. the everyday understanding of how the world works. But this knowledge is usually hidden inside the model in a way that is hard to see and understand, or use directly. Researchers are working to identify what commonsense LLMs have and learn how to use such knowledge more effectively in tasks like QA, reasoning, and explanation. A core research question is: How can we extract, represent, and apply the commonsense knowledge embedded in large language models in a way that makes it visible, interpretable, and useful for various downstream tasks? Consider an example in which an LLM is presented with the sentence "Jack dropped a glass on the floor" and asked, "What likely happened next?" A human would easily infer that the glass probably broke, relying on commonsense knowledge about the fragility of glass and universal gravitation, while the model may produce a plausible answer, such as "the glass shattered", the reasoning behind that response (e.g., "glass is fragile" or "dropping causes impact") may remain hidden. Future work may explore systematically mining the model's internal "world state", the commonsense knowledge base which stores its perceptions of the world and conditions its decisions across tasks.

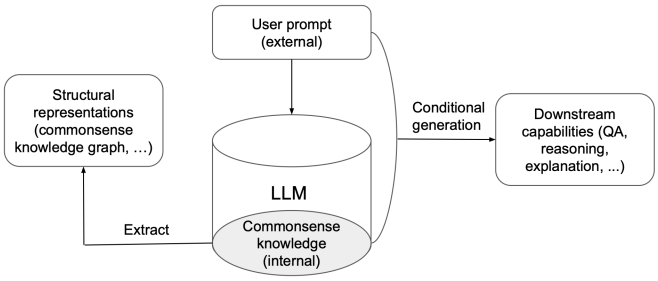

Figure 5.
An illustrative diagram of the process of applying LLMs' commonsense knowledge on downstream tasks and extracting LLMs' commonsense knowledge into structural representations.

4.6 *Probing and Editing Knowledge Neurons in Large Language Models*

Large Language Models often produce factual statements with high confidence, yet little is understood about how these facts are stored within the model's internal structures or retrieved at inference time. The overarching aim of this body of work is to edit the factual knowledge stored within an LLM and replace it with custom knowledge, resulting in a safe knowledge boundary useful for safety-critical applications, such as in the medical field. Early work by Dai et al. [44] first introduced the concept of "knowledge neurons", individual neurons whose activations are strongly correlated with factual recall. Follow-up research demonstrated that factual information is typically distributed redundantly across overlapping neuron subsets called "degenerate neurons" rather than isolated units [32]. More recent developments have proposed query-sensitive localization methods that dynamically identify neuron groups most relevant to a given input [35], challenging the earlier view that factual knowledge resides in fixed, localized neurons. The various activations can be seen in Figure 6. These advances enable precise model editing, as demonstrated by [33][22] without the need for complete retraining, laying the foundation for more robust and controllable language models.

These findings oppose prior assumptions about knowledge attribution and introduce new complexity to the problem of factual knowledge editing. As seen in practice, the phrasing of a factual statement can be expressed in semantically diverse ways. This variance, coupled with the redundancy of fact encoding across different shifting neuron groups, needs to be taken into account when formulating a knowledge editing framework. Imagine a language model trained on facts like "Paris is the capital of France" and "Einstein was born in Ulm." These two facts, although related, are stored in entirely separate groups of neurons. To retrieve or edit a fact, earlier methods try to locate knowledge neurons whose activation significantly correlates with "Paris" or "Einstein"[44]. Follow-up techniques dynamically identify degenerate neurons; the same fact being redundantly stored across disjoint neuron subsets[32], and query-relevant neurons whose activation varies with input phrasing. For instance, one neuron group may represent "Einstein" in physics contexts, and another when queried in German geography. Therefore, editing Einstein's birthplace without affecting his contributions to physics requires identifying and modifying only the relevant neuron group. This has been addressed by [35]. PMET[33] and DEPN[22] provide specialized techniques to update the identified hidden states, thereby editing factual knowledge or hiding private facts within the LLM. Researchers have also tried to provide solutions for decoder-only models and long-form generation[36]. Together, these advances converge on a more nuanced view: factual knowledge in LLMs is not stored in isolated neurons but in query-dependent, structurally connected, and sometimes degenerate neuron sets. Identifying, interpreting, and editing these sets remains an evolving frontier toward better controllability and transparency of LLMs. The existence of query-specific knowledge neurons is still not properly understood and may be attributed to the pre-training process[35].

5 Conclusion

This systematic review consolidates current knowledge on how LLMs autonomously determine the limits of their internal knowledge without recourse to external retrieval or prompting. The six thematic clusters identified collectively show that an LLM's parameters already encode both factual content and rich uncertainty cues. Approaches grounded in graph alignment and relational tracing demonstrate that latent representations can be projected onto structured ontologies, while studies of decoding dynamics and activation patterns reveal reliable internal markers of epistemic confidence thus establishing a principled foundation for models that can both expose and regulate their own knowledge limits without external instrumentation and work on neuron localization further indicates that factual memories can be modulated with minimal collateral impact, suggesting a path toward precision governance of model behavior. Despite the progress within approaches for knowledge boundary detection in LLMs, every existing method that directly attacked the problem of introspective boundary formalization retains some dependence on externally curated thresholds, priors, or evaluation scaffolds, indicating that a fully self-referential boundary detector remains an open challenge. Future research should prioritize sealed-loop benchmarking that perturbs only internal states, develop lightweight fusion modules that integrate heterogeneous epistemic signals into unified abstention mechanisms, and embed interactive boundary visualizations within model documentation to expose domain-specific blind spots before deployment. Framing knowledge-boundary detection as an integrated systems problem, spanning representation, inference, and governance, will be essential for realizing language models that can rigorously declare the limits of their knowledge and thus be trusted in high-stakes settings.

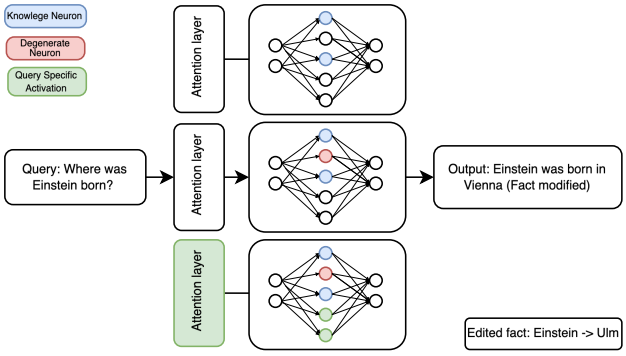

Figure 6.
A simplified illustration of how factual knowledge is stored, discovered, and edited in transformer-based language models. First, a user query (e.g., "Where was Einstein born?") activates a dynamic set of neurons. Then, editing methods like PMET precisely modify only the relevant neurons (e.g., to change Einstein's birthplace from "Ulm" to "Vienna").

References

[1] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. 2023. doi: 10.48550/ARXIV.2311.05232. URL https://arxiv.org/abs/2311.05232.

[2] Song Han, Zhengyi Guan, Sihui Li, Jin Wang, and Xiaobing Zhou. Knowledge graph completing with dual confrontation learning model based on variational information bottleneck method. In *2023 IEEE 23rd International Conference on Software Quality, Reliability, and Security (QRS)*, pages 741–750. IEEE, October 2023. doi: 10.1109/qrs60937.2023.00077.

[3] Shibo Hao, Bowen Tan, Kaiwen Tang, Bin Ni, Xiyan Shao, Hengzhe Zhang, Eric P. Xing, and Zhiting Hu. Bertnet: Harvesting knowledge graphs with arbitrary relations from pretrained language models, 2022.

[4] Mathias Lykke Gammelgaard, Jonathan Gabel Christiansen, and Anders Søgaard. Large language models converge toward human-like concept organization, 2023.

[5] Hanieh Khorashadizadeh, Nandana Mihindukulasooriya, Sanju Tiwari, Jinghua Groppe, and Sven Groppe. Exploring in-context learning ca-

pabilities of foundation models for generating knowledge graphs from text, 2023.

[6] Shaorong Xie, Qifei Pan, Xinzhi Wang, Xiangfeng Luo, and Vijayan Sugumaran. Combining prompt learning with contextual semantics for inductive relation prediction. *Expert Systems with Applications*, 238: 121669, March 2024. ISSN 0957-4174. doi: 10.1016/j.eswa.2023.121669.

[7] Ivan Vulić, Edoardo Maria Ponti, Robert Litschko, Goran Glavaš, and Anna Korhonen. Probing pretrained language models for lexical semantics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.586.

[8] Vinitra Swamy, Angelika Romanou, and Martin Jaggi. Interpreting language models through knowledge graph extraction, 2021.

[9] Vito Walter Anelli, Giovanni Maria Biancofiore, Alessandro De Bellis, Tommaso Di Noia, and Eugenio Di Sciascio. Interpretability of bert latent space through knowledge graphs. In *Proceedings of the 31st ACM International Conference on Information and amp; Knowledge Management*, CIKM '22, pages 3806–3810. ACM, October 2022. doi: 10.1145/3511808.3557617.

[10] Ankan Karmakar, Chintan Patel, and Venkata Santosh Kumar Delhi. From unstructured data to knowledge graphs: An application for compliance checking problem. In *Proceedings of the 41st International Symposium on Automation and Robotics in Construction (ISARC 2024)*, Chennai, India, 2024. IAARC. URL https://www.iaarc.org/publications/fulltext/111_ISARC_2024_Paper_214.pdf. Accessed: 2025-05-01.

[11] Roi Cohen, Mor Geva, Jonathan Berant, and Amir Globerson. Crawling the internal knowledge-base of language models. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1856–1869. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-eacl.139.

[12] Maxim Ifergan, Leshem Choshen, Roee Aharoni, Idan Szpektor, and Omri Abend. Beneath the surface of consistency: Exploring cross-lingual knowledge representation sharing in llms, 2024.

[13] Zijian Wang, Britney White, and Chang Xu. Locating and extracting relational concepts in large language models. 2024. doi: 10.48550/ARXIV.2406.13184.

[14] Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. Linearity of relation decoding in transformer language models, 2023.

[15] Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models, 2023.

[16] Jiahai Feng, Stuart Russell, and Jacob Steinhardt. Monitoring latent world states in language models with propositional probes, 2024.

[17] Evan Hernandez, Belinda Z. Li, and Jacob Andreas. Inspecting and editing knowledge representations in language models, 2023.

[18] Arthur Conmy, Augustine N. Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability, 2023.

[19] Alex Tamkin, Mohammad Taufeeque, and Noah D. Goodman. Codebook features: Sparse and discrete interpretability for neural networks, 2023.

[20] Ziwei Ji, Delong Chen, Etsuko Ishii, Samuel Cahyawijaya, Yejin Bang, Bryan Wilie, and Pascale Fung. Llm internal states reveal hallucination risk faced with a query. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 88–104. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.blackboxnlp-1.6.

[21] Mohammad Beigi, Ying Shen, Runing Yang, Zihao Lin, Qifan Wang, Ankith Mohan, Jianfeng He, Ming Jin, Chang-Tien Lu, and Lifu Huang. Internalinspector i2: Robust confidence estimation in llms through internal states. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 12847–12865. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.751.

[22] Xinwei Wu, Junzhuo Li, Minghui Xu, Weilong Dong, Shuangzhi Wu, Chao Bian, and Deyi Xiong. Depn: Detecting and editing privacy neurons in pretrained language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2875–2886. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.emnlp-main.174.

[23] Hongbang Yuan, Pengfei Cao, Zhuoran Jin, Yubo Chen, Daojian Zeng, Kang Liu, and Jun Zhao. Whispers that shake foundations: Analyzing and mitigating false premise hallucinations in large language models, 2024.

[24] Aryo Pradipta Gema, Chen Jin, Ahmed Abdulaal, Tom Diethe, Philip Teare, Beatrice Alex, Pasquale Minervini, and Amrutha Saseendran. Decore: Decoding by contrasting retrieval heads to mitigate hallucinations, 2024.

[25] Hadas Orgad, Michael Toker, Zorik Gekhman, Roi Reichart, Idan Szpektor, Hadas Kotek, and Yonatan Belinkov. Llms know more than they show: On the intrinsic representation of llm hallucinations, 2024.

[26] Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models, 2023.

[27] Chenhao Wang, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. Cn-automic: Distilling chinese commonsense knowledge from pretrained language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9253–9265. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.emnlp-main.628.

[28] Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. 2020. doi: 10.48550/ARXIV.2010.05953.

[29] Wenxing Hong, Shuyan Li, Zhiqiang Hu, Abdur Rasool, Qingshan Jiang, and Yang Weng. Improving relation extraction by knowledge representation learning. In *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 1211–1215. IEEE, November 2021. doi: 10.1109/ictai52525.2021.00191.

[30] Yanbo Fang and Yongfeng Zhang. Data-efficient concept extraction from pre-trained language models for commonsense explanation generation. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5883–5893. Association for Computational Linguistics, 2022. doi: 10.18653/v1/2022.findings-emnlp.433.

[31] Wangchunshu Zhou, Dong-Ho Lee, Ravi Kiran Selvam, Seyeon Lee, Bill Yuchen Lin, and Xiang Ren. Pre-training text-to-text transformers for concept-centric common sense, 2020.

[32] Yuheng Chen, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. Journey to the center of the knowledge neurons: Discoveries of language-independent knowledge neurons and degenerate knowledge neurons, 2023.

[33] Xiaopeng Li, Shasha Li, Shezheng Song, Jing Yang, Jun Ma, and Jie Yu. Pmet: Precise model editing in a transformer, 2023.

[34] Yuheng Chen, Pengfei Cao, Yubo Chen, Yining Wang, Shengping Liu, Kang Liu, and Jun Zhao. Cracking factual knowledge: A comprehensive analysis of degenerate knowledge neurons in large language models, 2024.

[35] Yuheng Chen, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. Knowledge localization: Mission not accomplished? enter query localization!, 2024.

[36] Lihu Chen, Adam Dejl, and Francesca Toni. Identifying query-relevant neurons in large language models for long-form texts, 2024.

[37] Patrik Puchert, Poonam Poonam, Christian van Onzenoodt, and Timo Ropinski. Llmmaps – a visual metaphor for stratified evaluation of large language models, 2023.

[38] Amirhossein Kazemnejad, Mehdi Rezagholizadeh, Prasanna Parthasarathi, and Sarath Chandar. Measuring the knowledge acquisition-utilization gap in pretrained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4305–4319. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.285.

[39] Shiyu Ni, Keping Bi, Lulu Yu, and Jiafeng Guo. *Are Large Language Models More Honest in Their Probabilistic or Verbalized Confidence?*, pages 124–135. Springer Nature Singapore, 2025. ISBN 9789819617104. doi: 10.1007/978-981-96-1710-4_10.

[40] Yile Wang, Peng Li, Maosong Sun, and Yang Liu. Self-knowledge guided retrieval augmentation for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.691.

[41] Xunjian Yin, Xu Zhang, Jie Ruan, and Xiaojun Wan. Benchmarking knowledge boundary for large language models: A different perspective on model evaluation, 2024.

[42] Lida Chen, Zujie Liang, Xintao Wang, Jiaqing Liang, Yanghua Xiao, Feng Wei, Jinglei Chen, Zhenghong Hao, Bing Han, and Wei Wang. Teaching large language models to express knowledge boundary from their own signals, 2024.

[43] Amirhossein Kazemnejad, Mehdi Rezagholizadeh, Prasanna Parthasarathi, and Sarath Chandar. Measuring the knowledge acquisition-utilization gap in pretrained language models, 2023.

[44] Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*, 2021.