MEMORY IN LARGE LANGUAGE MODELS: MECHANISMS, EVALUATION AND EVOLUTION

Dianxing Zhang¹, Wendong Li^{1,*}, Kani Song^{1,*}, Jiaye Lu^{1,*}, Gang Li¹, Liuchun Yang¹, and Sheng Li^{1,†}

Digital China AI Research Institute zhangdxh@digitalchina.com lishengh@digitalchina.com

ABSTRACT

Under a unified operationalized definition, we define LLM "memory" as a persistent state that is written during pretraining, finetuning, or inference, can be subsequently addressed, and stably influences outputs. On this basis, we propose a four-way taxonomy (parametric, contextual, external, and procedural/episodic) and a "memory quadruple" (storage location—persistence—write/access path—controllability), and connect mechanism, evaluation, and governance through a causal chain of "write—read—inhibit/update." To avoid distorted comparisons arising from heterogeneous settings, we provide a three-setting parallel protocol (parameter-only, offline retrieval, online retrieval) that decouples model capability from information availability on the same data slice and timeline; we then construct a layered evaluation: parametric memory (closed-book recall, edit differential, memorization/privacy risks), contextual memory (position-performance curves and the "mid-sequence drop"), external memory (decoupling correctness from snippet-level attribution/faithfulness), and procedural/episodic memory (cross-session consistency and timeline replay, E-MARS+). The framework uniformly incorporates temporal governance and leakage auditing (Freshness hits, outdated answers, refusal slices), as well as uncertainty reporting via inter-rater agreement and paired tests/multiplecomparison correction. For updating and forgetting, we propose DMM-Gov dynamic governance: coordinating DAPT/TAPT, PEFT, model editing (ROME/MEND/MEMIT/SERAC), and RAG to form an auditable closed loop of "admission thresholds—progressive rollout—online monitoring—reversible rollback—change audit certificates", together with operational and acceptance specifications for timeliness/conflict handling and long-horizon consistency. Finally, we put forward four classes of testable propositions (minimum identifiability, a minimally sufficient evaluation card, causally constrained editing and verifiable forgetting, and the conditional boundary under which retrieval plus small-window replay is preferable to ultra-long-context direct reading), thereby providing a shared coordinate system and methodological baselines—reproducible, comparable, and governable—for research and deployment.

Keywords LLM memory; parametric memory; contextual memory; external memory; procedural/episodic memory; three-regime evaluation protocol; passage-level evidence attribution and faithfulness; long-context; knowledge editing and unlearning; temporal governance and auditing

1 Introduction

"When a physician consults an AI assistant for the latest treatment options but receives recommendations for a drug that was phased out three years ago; when an attorney asks an AI to retrieve case law and it confidently cites a statute that does not exist—these are not scenes from science fiction, but real risks arising from deficiencies in the 'memory' mechanisms of today's large language models. As LLMs move from the lab into high-stakes domains such as healthcare, finance, and law, their 'memory'—namely, the ability to store, update, and forget knowledge—has evolved from an academic topic into the lifeblood of safety, compliance, and trust." [1–5]

^{*}Equal contribution.

[†]Corresponding author.

The memory of large language models (LLMs) refers to a persistent state that is written at any of the stages of pretraining, finetuning, or inference and can subsequently be stably addressed, thereby systematically influencing model outputs. This capability spans the model's entire life cycle and is the cornerstone for the transition from "language understanding" to "knowledge application." Early studies—such as the LAMA probes—first systematically validated, under the parametric-only (PO) setting, the model's ability to recall facts from parameters, revealing its potential as an "unsupervised knowledge base." [6, 7] Subsequent mechanistic research further indicated that the Transformer's feed-forward layers (FFN) can be interpreted as key–value memory, providing a mechanistic handle for localizing and editing knowledge at the parameter level. [8–10]

However, a flourishing academic literature cannot mask practical difficulties. The current research ecosystem on LLM memory faces three core challenges that seriously hinder reliable deployment:

Blurred conceptual boundaries. Definitions, life cycles, and intervention modes for parametric, contextual, and external memory are often conflated. Basic questions such as "In a RAG system, are documents 'external memory' or 'contextual memory'?" lack consensus, leading to irreproducible experimental designs and non-comparable results. [11–16]

Fragmented evaluation lenses. Assessments often lump disparate aspects together indiscriminately. RAG systems must measure retrieval quality and also evaluate the faithfulness and source attribution of generated outputs to evidence, yet existing benchmarks (e.g., RAGAS, RAGChecker) often conflate the two, or exhibit blind spots when handling cross-source conflicts and identifying key evidence in long documents. [17–27]

Bias in automated judging. Evaluations relying on LLM-as-a-Judge have been shown to suffer from serious position, order, and self-preference biases, causing "spurious significance" to be mistaken for real progress. [28–30]

In response to these challenges, this paper aims to build an end-to-end "operating system" for LLM memory governance. Rather than merely proposing a set of new methods, we seek to construct a unified, operational analytical framework(Figure 1) that bridges the gap between academic research and industrial practice. Our key contributions are:

- (i) **Unified definitions and taxonomy.** We propose an operationalized definition—memory = persistent and addressable state—and, via a quadruple of storage location—persistence—update path—access method, characterize parametric, contextual (working), external (non-parametric), and procedural (episodic) memory; we clarify their boundaries vis-à-vis knowledge/ability/context state, linking them to reproducible experimental designs; [6, 8, 16, 31]
- (ii) **Survey of memory mechanisms.** Using Retrieval (R)—Write (W)—Inhibit (I) as the main thread, we cover the training phase (W: compressive writing of the corpus into weights; R: differentiable retrieval/data selection), the inference phase (R: context retrieval/external injection; W: context loading and procedural writes), and post-training shaping (I: control interfaces for instruction/preference alignment and for editing/forgetting), assembling mechanistic evidence and delineating attainable limits and risks (verbatim memorization, privacy exposure, and side-effect propagation); [11, 12, 32–62]
- (iii) **Evaluation framework.** We propose layered evaluation and unified metric design across the four memory types: parametric memory uses the PO setting plus pre/post-edit differentials; long-context focuses on positional robustness and sensitivity to lost-in-the-middle; external memory adopts a dual-channel assessment of retrieval quality × faithfulness and source attribution; procedural memory emphasizes cross-session consistency and trajectory replay. We additionally recommend timestamp-aligned protocols for time-sensitive scenarios; [1–3, 17–23, 25–27, 63–74]
- (iv) **Forgetting and updating.** We discuss evaluation protocols and side-effect boundaries for model editing/machine unlearning, with a Pareto analysis along three axes—target suppression, neighborhood preservation, and downstream steady state—supplemented by security regression tests such as membership inference and data extraction. [48–52, 57–59, 75, 76]
- (v) **Deployment-oriented engineering principles.** We distill five principles—external-memory-first, small-step editing, long-task read/write strategies, timestamp alignment, and debiasing for privacy and evaluation—forming an actionable governance checklist (see §5.4), together with evaluation and online monitoring metrics such as RAGBench/RAGAS/RAGChecker. [4, 5, 14, 15, 77–89]

Through this framework, we not only provide the research community with reproducible and comparable baselines, but also furnish the industry with a methodology for building LLM applications that are safe, reliable, and compliant. We hope this "operating system" will bridge theory and practice, advancing LLM memory research from a patchwork of techniques toward a unified engineering blueprint.

The remainder of this paper is organized as follows: Section 2 introduces the technical background and core pain points of LLM memory; Section 3 presents unified definitions, taxonomy, and formation mechanisms; Section 4—the core of

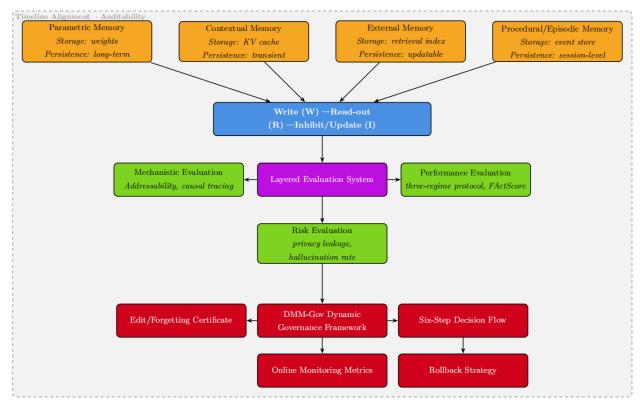


Figure 1: A unified framework for LLM memory research: mechanisms, evaluation, and governance.

this paper—details a layered evaluation framework covering the four memory types; Section 5 discusses strategies and governance frameworks for memory updating and forgetting; Section 6 analyzes current challenges and outlines future directions; Section 7 concludes.

2 Technical Background of LLM Memory

2.1 From Language Models to the Problem Space of "Memory Systems"

Contemporary large language models are not merely conditional distribution approximators; rather, along the full chain of pretraining–finetuning–inference they continually deposit and repeatedly invoke information states that are addressable. Early evaluations under the parametric-only (PO) (closed-book) setting show that, without accessing external evidence, models can directly recall a substantial amount of relational knowledge from parameters—an observation summarized as the feasibility and limits of "LM-as-KB". [6, 7, 90] Subsequent mechanistic studies further indicate that the Transformer's feed-forward layers (FFN/MLP) form mid-layer storage structures approximating key–value pairs, and that specific pathways can read out subject—attribute associations from the weights, making these layers the primary carriers of factual recall. [8–10, 42] Meanwhile, observations at the level of attention circuits have revealed reusable substructures such as induction heads, which copy patterns and align sequence positions under few-shot prompting, furnishing visible evidence for style continuation and short-term maintenance at inference time. [91–94] From this perspective, language models naturally possess multiform memory: part of it sinks into parameters and architecture, and part of it surfaces during inference as activations or external components, jointly shaping the upper bounds of factual recall, long-document utilization, and dialog coherence. Formal definitions and type boundaries are given in §3.1–§3.2; causal evidence for "write—read—inhibit/update (R–W–I)" is presented in §3.3.

2.2 Where Memory Lives and How It Is Invoked: From Network Internals to System Extensions

A commonly held mechanistic hypothesis is that multi-head attention acts as a content-addressable read operator, aggregating the context under a given query; the feed-forward network provides a parametric storage/write channel, imprinting high-mutual-information structures into the weights, thereby explaining the coexistence of robustness for

commonsense patterns and fragility for long-tail knowledge . [8–10] This division of labor is not exclusive; inter-layer and intra-layer coupling and exceptions persist. To break the temporal and capacity limits of single-segment inputs, Transformer-XL extends the visible time horizon to multi-segment sequences via segment-level recurrence and relative positional encoding, enabling cross-segment replay of historical activations and key-value caches; the Compressive Transformer retains more distant cues through hierarchical compression without significantly increasing cost. [77, 78] However, "visible" does not automatically mean "usable": as context length increases, attention becomes diluted and positional biases systematically depress the utilization of mid-span information—an failure mode repeatedly validated in subsequent long-context evaluations . [63–65] The corresponding position–performance curves and mid-span drop will serve as core robustness metrics in §4.3.

Relying on weights alone cannot cover timeliness and the long tail; accordingly, systems introduce non-parametric evidence paths at inference time. Retrieval-augmented generation (RAG) incorporates external documents into the generation loop via index-retrieve-fuse, improving factual consistency and traceability without altering weights; going further, REALM jointly optimizes the retriever and the language model end-to-end so that "finding sources" is acquired already during pretraining; RETRO connects by cross-attention to trillion-scale external stores, structurally loosening the equation "parameter scale = knowledge capacity"; kNN-LM performs local posterior correction via nearest neighbors, providing plug-and-play support for long-tail tokens and new-domain concepts . [11–14, 95] Existing evidence shows that end-to-end performance is highly sensitive to retrieval recall, and reranking often yields notable joint gains in recall and faithfulness. [17, 39–41, 96]

As models move from dialog assistants to agents with sustained goals and planning, the system must also structure and replay key events and intermediate states from interaction, in order to maintain cross-session consistency and track long-term objectives—Generative Agents and MemGPT provide operational blueprints via layered memory stores and "virtual memory"-style scheduling, respectively . [31, 97] The resulting picture is one of internal—external collaboration: the network interior consolidates general patterns, while system extensions supply fresh evidence and a process timeline; together they determine both achievable performance and governability. Component-level metrics and alignment criteria for external memory are given in §4.4, and evaluation for procedural/episodic memory appears in §4.5.

2.3 Principal Pain Points and Design Trade-offs of Memory Systems

- (1) The tension between controllable writes and revocability. Self-supervised training tends to imprint high-mutual-information and highly repeated fragments into the weights, which not only yields robust commonsense recall but also elevates risks of verbatim reproduction and privacy leakage. Under black-box conditions, training-data extraction has been demonstrated on early models, showing that specially crafted queries can reconstruct long spans of training samples. [57] Engineering de-duplication can reduce regurgitation rates, shorten convergence, and weaken privacy threats that use highly repetitive samples as attack vectors, all without materially harming perplexity; however, poorly chosen thresholds introduce new trade-offs between coverage and quality. [60, 61] Regarding the decision of "seen vs. unseen," conventional membership inference often approaches random-guess accuracy under standard pretraining, and is highly sensitive to distributional and temporal drift; hence, more robust risk assessment should jointly consider de-duplication thresholds, model scale, the corpus timeline, and evaluation protocols within a unified framework, rather than drawing conclusions from a single attack metric. [58, 59, 62]
- (2) **For long context, "visible" is not the same as "usable."** Enlarging the context window increases visibility, but attention allocation and positional bias systematically depress the utilization of mid-span evidence, so long context does not automatically translate into effective read/write. This failure mode has been repeatedly observed in Lost in the Middle and subsequent evaluations. [63, 65] Accordingly, structured reordering, anchor-guided prompting, and positional reweighting become necessary strategic complements. [64, 79–83]
- (3) **Reliability and governance cost of external evidence paths.** External memory (e.g., RAG) provides timeliness and traceability, while also importing the errors of the index–embedding–reranking–fusion pipeline into the generation loop: mismatches between embeddings and chunking can introduce noisy documents; cross-source evidential conflicts require explicit consistency adjudication and refusal mechanisms; the freshness and rollback capability of the index directly affect release cadence and the compliance audit trail. [1, 3–5, 17, 18, 20–24, 98] Layered evaluation of retrieval quality and faithfulness/source attribution metrics is presented in Chapter 4.
- (4) **Management of event and temporal structure.** As tasks expand from single-turn Q&A to long-horizon interaction, factual memory alone is insufficient; the system must treat process as a first-class citizen, maintaining behavioral consistency and progress toward long-term goals via queryable timelines and summary replay. [31, 73, 74, 97, 99–103] Omitting this layer leads to failure to retrieve key evidential snippets; conversely, writing procedural details directly into the weights induces non-revertibility and accumulation of side effects.

(5) Choosing the locus for in-parameter edits and compliant forgetting. Pointwise editing shows that write locations can be localized and manipulated: ROME rewrites factual associations via rank-1 updates at causal mediation sites in mid-layer MLPs; MEND performs rapid local adjustments via low-rank gradient transformations; MEMIT scales direct editing to tens of thousands of facts. [42–45] Nonetheless, there is inherent tension among effectiveness, locality, and generalization; sequential or batched edits readily trigger off-target effect diffusion. [104–108] A reasonable division of responsibilities and boundaries vis-à-vis external evidence paths determines the engineering feasibility of updates and forgetting. [46–49, 51–56, 76, 109, 110]

Synthesis. Parametric memory offers recall under the PO setting but is pressured on precise controllability and revocability; runtime contextual memory (CM) enlarges visibility, whose usability depends on positional and read/write strategies; external memory (EM) and procedural memory bring timeliness and traceability, while requiring rigorous component-level governance and consistency adjudication. Type taxonomy and decision criteria appear in Chapter 3; evaluation dimensions and protocols in Chapter 4; and engineering pathways and governance frameworks for updating/forgetting in Chapter 5. [6, 11, 16–19, 21, 31, 63, 65, 66, 97]

3 Taxonomy and Mechanisms of LLM Memory

3.1 Conceptual Boundaries and an Operationalized Definition

The prevailing literature indicates that large language models do more than conditionally generate on a given context: across different stages of training and inference they form collections of states that can be stably addressed at subsequent time steps and influence outputs. Accordingly, we define LLM memory as a persistent state that is written during pretraining, finetuning, or inference, and that can later be read to produce systematic effects on the model's outputs. To avoid equating "memory" with any single substrate, our analysis records four dimensions—storage location, persistence, write path, and access method—without presupposing a one-to-one mapping to specific implementations. Thus, long-term representations fixed in parameters and activations/caches during inference fall under the same semantic category, differing primarily in persistence and access path; external document stores, vector stores, and tool/API returns are likewise treated as addressable external states, provided they enter the generation loop and exert reproducible influence on outputs. This definition aligns with empirical observations that language models can recall facts under the parametric-only (PO) setting, and it is corroborated by mechanistic evidence that interprets Transformer feed-forward layers as key-value memory, thereby furnishing a unified semantics and testable interface for subsequent measurement and ablations [6, 8, 9].

Within this framework, memory, knowledge, ability, and context state exhibit a layered relationship. Knowledge denotes facts or relations that can be stably reproduced and verified, an important subset of which exists in parametric form and can be recalled under the PO setting by probes such as LAMA [6, 7]. Ability emphasizes task-facing algorithmic processes, often loaded on-the-fly during the forward pass as in-context learning; at the circuit level, studies document induction heads that copy and align across positions, while at the algorithmic level this loading is explained as implicit Bayesian updating or approximate gradient descent under fixed weights [91–94]. Context state refers to the instantaneous token sequence and KV cache exposed to the model at inference; its effective time horizon can be extended via segment-level recurrence or compressed replay, but it does not automatically become long-term memory [77, 78]. Compared with these notions, memory stresses a sustainable, addressable, and read–writable set of states that encompasses both long-term parametric representations and evidence/intermediate variables retrieved or injected via tools [11–13, 95].

From a life-cycle perspective, pretraining (with cross-entropy objectives) imprints high-mutual-information structures into parameters; finetuning and model editing alter activation thresholds and call probabilities of these structures; inference loads and reads relevant states via in-context learning, retrieval, and tool use. Risks along this path can be described uniformly: when the corpus contains low-entropy, highly repeated fragments, the probability of verbatim memorization and privacy exposure increases; conversely, systematic de-duplication can reduce regurgitation and extraction rates without materially harming perplexity, suggesting that memory evaluation should be co-designed with data governance [57–61]. The present section provides operational terminology and observables for the unified evaluation in §4 and for update/forgetting governance in §5.

3.2 Memory Types and Their Boundaries

Without altering the above definition, we minimally distinguish memory types by their observable carriers and invocation modes.

- (i) **Parametric memory.** Representations consolidated in the weights (especially FFN layers) through pretraining/finetuning; they support factual recall and pattern continuation under the PO setting but are sensitive to time-critical knowledge and long-tail facts, with controllability depending on dedicated editing and rollback mechanisms [6, 8, 42, 43, 45].
- (ii) **Contextual memory.** Activations and caches formed on the fly during inference by inputs and history; they enable style alignment and rule fitting under few-shot prompting, but are markedly affected by position and length. Multiple evaluations report that mid-span evidence is harder to utilize effectively in ultra-long sequences; this effect persists even as the visible window is enlarged [63–65].
- (iii) **External or "non-parametric" memory.** Mechanisms that incorporate documentary evidence or nearest neighbors into the generative distribution via retrieval; advantages include updatability and auditability, with canonical pathways including retrieval-augmented generation, differentiable retrieval during pretraining, and nearest-neighbor interpolation. At the same time, errors in indexing, chunking, and fusion can inject noise and conflict into answers, necessitating component-level constraints on faithfulness and source attribution [11–13, 17, 18, 25–27, 95, 96, 111].

When cross-session persistence and long-horizon behavioral consistency are required, intermediate steps, event snippets, and timelines from interaction can be stored in structured form and replayed as needed; this procedural/episodic memory emphasizes temporal structure and re-playability and often shares infrastructure with external memory in practice [31, 97, 112, 113].

Boundary ambiguity and coupling in practice: the "dual identity" problem in RAG. The boundary between contextual and external memory is not cleanly separable at the system level; instead, there is substantial coupling and overlap. The canonical case is RAG: documents returned by the retriever must be serialized and concatenated into the prompt before reaching the generator, thereby becoming part of the model's current context window.

Evaluation priorities and metric decoupling. To address this, we advocate, in coupled scenarios (e.g., RAG), a source-first, effect-separation principle:

Priority:

prioritize source attribution. For diagnosing bottlenecks or reporting core capabilities, first assess the contribution of external memory—i.e., retrieval quality (Recall@k, nDCG) and faithfulness/attribution to evidence (e.g., FActScore, Citation Recall). External memory is the defining value of RAG relative to purely parametric models; its updatability and auditability are the design raison d'être. If the retrieved document is wrong or irrelevant, higher downstream context-utilization cannot compensate.

Metric choice:

explicit decoupling. To avoid misattributing poor context utilization as ineffective external knowledge, employ layered metrics:

- Layer 1: **Retrieval-quality metrics.** Does the system find the correct evidence? (e.g., Recall@5, nDCG@10) [39, 41, 96].
- Layer 2: **Attribution and faithfulness metrics.** Is the generated answer faithful to the retrieved evidence, with correct citations? (e.g., FActScore, Citation Precision/Recall) [25–27, 111, 114].
- Layer 3: **Context-utilization efficiency.** Conditional on correct and cited evidence, evaluate how efficiently the model uses this now-contextualized external evidence—e.g., place the same evidence at beginning/middle/end to measure positional losses [63–65].

Boundary judgments should follow observable behavior. Stable answering under the PO setting without external materials indicates parametric memory; phenomena that require placing information in the prompt/history indicate contextual memory; answers that depend on citable external evidence reflect external memory; and behaviors that require cross-turn replay of structured events/timelines to maintain consistency involve procedural/episodic memory. For borderline cases, combine clues such as whether information entered the weights, whether it persisted across sessions, and whether it entered the generation loop via retrieval/tools. Note that retrievers and caches do not automatically constitute memory; they count only when their returns enter the generation loop and reproducibly affect outputs [11–13]. Regarding parametric vs. contextual memory, it is apt to treat weights as "what is knowable on call," and activations/caches as "what is currently called." They differ systematically in substrate, time scale, write path, and risk profile: the former is persistent and hard to revoke, prone to verbatim memorization/data extraction; the latter is transient and failure-prone, with issues arising from positional bias and misreads. In measurement and governance,

these differences map to distinct metrics and processes [57, 60, 61, 63, 65]. This typology provides the unified interface for the evaluation dimensions in §4 and for technical choices in §5.

Operational criteria (decision rules).

- If, under the PO setting with retrieval and tools disabled and without cross-session caches, the model still recalls stably, attribute the behavior to parametric memory.
- If information is usable only when present in the current context/KV cache and shows marked sensitivity to evidence position and length, attribute it to contextual memory (CM).
- If answers depend on returns from external indices/retrieval and can provide passage-level citations, attribute them
 to external memory (EM).
- If maintaining consistency requires cross-turn/session replay of structured events/timelines, attribute it to procedural/episodic memory. One-off tool outputs within a single turn do not count as memory, unless they are structuredly written and replayed in subsequent turns with reproducible impact on outputs.

3.3 Formation Mechanisms: Write, Read, and Inhibit/Update

Embedding memory within a write—read—inhibit/update cycle helps unify training, inference, and system integration along a single causal chain.

Write begins in pretraining: optimizing next-token/masked-token prediction compresses high—mutual-information structures. A common mechanistic hypothesis holds that FFNs form addressable key–value channels, consistent with empirical observations that general/high-frequency patterns are recalled under the PO setting [6, 8, 9]. Probes such as LAMA repeatedly show this at a macro level, while intra-layer evidence demonstrates that interventions on critical MLP layers substantially degrade factual recall, narrowing "where knowledge resides" to actionable structural units. Associated memorization risks have been widely observed: low-entropy, highly repeated fragments are more likely to be fixed verbatim (see de-duplication/extraction evidence). Training-data extraction under black-box conditions shows the feasibility of reconstructing training spans, whereas cross-corpus de-duplication reduces regurgitation and extraction probabilities with negligible perplexity loss—suggesting that write strength and privacy risk can be partially mitigated through data-level governance [57, 59–61].

Read occurs at inference; it is not a literal query to stored weights but a process in which attention and KV caches map prompts and history into usable rules within the given context. Circuit-level evidence records induction heads that copy/align across positions, micro-explaining style continuation under few-shot prompting; algorithmic treatments cast in-context learning as implicit Bayesian updating or approximate gradient descent under fixed weights. Together they indicate that transient ability loading and long-term knowledge storage coexist, though with different time scales and robustness [91–94]. To extend the visible range on which read relies, Transformer-XL uses segment-level recurrence and relative positional encoding to reach beyond adjacent segments, while the Compressive Transformer retains more distant cues via hierarchical compression at limited cost; yet "visible \neq usable," as positional bias and attention dilution suppress effective read of mid-span evidence. Under fixed resource budgets, multiple evaluations report that structured reordering, anchor prompting, and mid-span reweighting often yield more reliable gains than window expansion alone [63, 64, 77–83].

Externalization and fusion provide a third pathway for memory. RAG incorporates documentary evidence into the generative loop via index-retrieve-cross-attention fusion, fundamentally improving traceability and timeliness; REALM jointly trains retrieval during pretraining, internalizing "finding sources" as a learnable module; RETRO links by chunked cross-attention to trillion-scale external stores, substantially loosening the coupling "parameter scale \approx knowledge capacity"; kNN-LM locally post-hoc corrects the LM distribution via nearest neighbors, offering training-free, on-the-fly support for long-tail terms and new-domain concepts. Meanwhile, mismatches in indexing, chunking, and fusion can inject noise and conflict into answers; absent faithfulness and provenance constraints, the system may expose uncertainty to users. Ensuring that external evidence functions as memory rather than search noise typically requires elevating component-level objectives (e.g., retrieval contribution and generative faithfulness) to the same priority as end-to-end accuracy, and auditing the ingress and use of evidence within the generation loop [11–13, 17, 18, 25–27, 95, 96, 111].

Inhibit/update determine the governability of memory over time. Continued pretraining and instruction/domain finetuning reallocate which units are easier to activate, thereby changing the probability distribution of when and how items are recalled; the tension between stability and plasticity implies that overly deep parameter updates risk

catastrophic forgetting. For controllable rewriting and compliant suppression, model editing offers low-overhead, reversible paths: ROME applies rank-1 updates at causal mediation sites in mid-layer MLPs to directly rewrite target associations; MEND performs rapid local adjustments via low-rank gradient transforms; MEMIT scales direct editing to tens of thousands of facts and shows scalability on medium/large models. A common theme is that effectiveness, locality, and generalization cannot be simultaneously optimal; hence standardized validation protocols along target suppression, neighborhood preservation, and downstream performance yield more comparable results [42, 43, 45, 104–108]. In scenarios requiring revocability and accountability, programmatic tool access can fold external system returns into context so that "do not output certain information" policies take effect at runtime—thus achieving governance without touching base weights [4, 5].

Putting the three pathways together, memory co-evolves inside and outside the parameters: the availability of parametric memory, the timeliness of contextual loading, the traceability of retrieval/fusion, and the governability of editing/abstention jointly set the balance among capability—timeliness—controllability. This chapter lays out the causal chain and design space; concrete testable propositions and closed-loop measurements—e.g., trade-offs between de-dup thresholds and leakage, the marginal gains of positional reweighting in long context, the Pareto frontier of editing/forgetting across target suppression—retained performance—neighborhood generalization, and the joint optimization of retrieval contribution and generative faithfulness—will be unified in the evaluation protocols of §4 and instantiated as engineering processes and governance frameworks for update/forgetting in §5 [6, 8, 11–13, 17, 18, 42, 43, 45, 57, 59–61, 63, 65, 95].

4 Evaluation of LLM Memory

4.1 Scope and Methodology

This chapter delineates the scope, methodological stance, and organizing logic for evaluating memory in large language models. Rather than forcing a single score or a unified yardstick across all tasks, we acknowledge the heterogeneity of different memory forms and ensure comparability and reproducibility through shared design primitives: unified terminology and object definitions, parallel operating regimes (parametric-only / offline retrieval / online retrieval), auditable families of metrics and comparison dimensions, and transparent versioning and temporal governance. The overall goal is to establish a layered, interpretable, and engineering-usable paradigm for memory evaluation, not to substitute multi-dimensional evidence with a single number. [115, 116]

Under this framework, each memory form has distinct evaluation foci:

- (i) Parametric memory (§4.2): closed-book attainability, maintainability under editing/forgetting and their side effects, and privacy risks; [42, 43, 57, 58, 75]
- (ii) Contextual memory (§4.3): sensitivity to length and position, cross-span integration, and robustness to interference; [63–66, 71]
- (iii) External memory (§4.4): end-to-end coupling among retrieval–attribution–generation, evidential faithfulness and timeliness; [11–13, 37, 38, 117]
- (iv) Procedural/episodic memory (§4.5): correctness and stability of the long-horizon write-replay-inhibit loop and its engineering cost. [73, 99, 112]

4.1.1 Chapter Positioning and Scope

A memory store denotes any carrier that can be read/written during inference and exerts a stable influence on outputs, including parameter weights (parametric memory), visible context (single- or multi-turn), externally retrieved documents/tables/graphs/multimodal evidence, and session-level memory stores. In contrast, infrastructure—retrievers/rerankers, caching/compression strategies, executors, etc.—is not itself memory; however, once its products enter the generation loop and affect outputs, they fall within the scope of evaluation. [6, 8, 16]

This section does not attempt to homogenize metrics across the four memory types. Instead, it summarizes for each the evaluation objectives, typical operating regimes, and metric families, and identifies comparable dimensions and common pitfalls, thereby providing the methodological frame for §§4.2–4.5.

4.1.2 Typical Evaluation Regimes: From Parametric-Only to Online

To separate capability from information availability, prior work commonly adopts three regimes; whether to run the three in parallel depends on the study's purpose.

Parametric-Only (PO): disable external evidence and tools; examine parametric recall, side effects of editing/forgetting, and privacy risks (§4.2). This also provides a "zero-injection" reference baseline for contextual/external/procedural memory.

Offline Retrieval/Replay: fix the index or the session memory store; useful for stepwise diagnosis of retrieve \rightarrow attribute \rightarrow generate, for analyzing the marginal contribution of context injection (§4.3), end-to-end faithfulness in RAG (§4.4), and structured replay (§4.5).

Online Retrieval: connect to dynamic knowledge sources; focus on freshness hits, outdated answers, and refusal calibration (§§4.4–4.5), mirroring real deployment. [1–3]

Reporting standards and control requirements. To ensure decoupling of "model capability" from "information availability," any conclusion about capability change must satisfy: [17–19, 21, 28, 29, 118]

Controlled baselines: provide a same-domain, same-slice Parametric-Only (PO) or Offline-Retrieval baseline (sharing the data slice and time window with the compared regime);

Version lock: fix the index/session-store version and disclose snapshot timestamps and de-duplication strategy;

Unified outputs: under all regimes, report answer text, evidence citations (or replay items), model confidence, and refusal flags;

Statistical testing: for cross-regime comparisons, use paired bootstrap/permutation tests with Holm–Bonferroni or FDR multiple-comparison correction;

Reproducibility details: disclose random seeds, decoding and retrieval hyperparameters, hardware, and per-unit cost. If any condition is unmet, explicitly state the limitation in the text and provide sensitivity analyses.

Reporting guidelines and control recommendations. To decouple *model capability* from *information availability*, studies that **claim capability changes** or **compare regimes** (PO/offline/online) should, at a minimum, include the following [17–19, 21, 28, 29, 118]:

- **Controlled baselines:** Provide a same-domain, same-slice Parametric-Only (PO) or Offline-Retrieval baseline (sharing the data slice and time window with the compared regime).
- **Version lock:** Fix the index/session-store version and disclose snapshot timestamps and the de-duplication strategy.
- **Unified outputs:** Under all regimes, report answer text, evidence citations (or replay items), model confidence, and refusal flags.
- **Statistical testing:** For cross-regime comparisons, use *paired* bootstrap or permutation tests with Holm–Bonferroni or FDR multiple-comparison correction.
- **Reproducibility details:** Disclose random seeds, decoding and retrieval hyperparameters, hardware, and per-unit cost.

When any item cannot be met (e.g., resource or compliance constraints), explicitly state the limitation in the main text and provide robustness/sensitivity analyses in lieu of the missing control.

4.1.3 Families of Metrics and Comparison Dimensions

For ease of cross-section reference, we do not force a single reporting protocol across memory types; instead, we organize commonly used measures into eight metric families. In Table 4.1:Metric Families — Task Mapping Matrix we list, for each family, representative metrics, the primary subsections where they apply, and typical outputs. In tandem, Table 4.2:Comparison Dimensions — Recommended Plots and Statistical Notes provides four cross-task comparison axes: (i) capability vs. information availability, (ii) correctness vs. faithfulness, (iii) quality vs. cost, (iv) stability and uncertainty. Each axis is paired with standardized visualizations (e.g., length–performance curves, AURC risk–coverage curves, cost–accuracy frontiers) and minimal statistical requirements (e.g., 95% confidence intervals, paired permutation/bootstrap tests, and Holm/FDR multiple-comparison corrections). Throughout the chapter, this framework is reused so that task specificity is preserved while methodological comparability is achieved. [7, 25, 26, 39–41, 96, 98, 111, 119–121]

To avoid terminological overload, the main text uses Tables 4.1 and 4.2 solely as navigation panels. The subsequent sections (§§4.2–4.5) instantiate, in their respective contexts, the required families and dimensions, as well as the associated formulas, evaluators, and implementation details. Any claim of capability change across settings or methods must provide same-domain, same-slice PO or offline controls, together with paired statistics and an explicit presentation of uncertainty.

Table 4.1: Metric Families — Task Mapping Matrix

Family	Representative Metrics	Main Apps (§)	Typical Outputs	Notes / Implementation Points
Accuracy / Task Performance	EM, F1, ROUGE, Keyword/Keyphrase Recall	4.2/4.3/4.4/4.5	Sub-task / slice-level macro average; 95% CI	Multi-step tasks: report sub-problem & aggregate levels
Groundedness / Attribution	Citation Coverage, Unsupported Claim Rate (UCR), span-level alignment, NLI consistency	4.3/4.4/4.5	Span-level P/R, FActScore, atomic-fact support rate	Specify evidence scope: context / replay / retrieved docs
Retrieval / Ranking (IR)	Recall@k, nDCG, MRR, Hit@k	4.3/4.4	Top- k curves, MRR \pm CI	Present jointly with end-to-end groundedness/EM
Sensitivity & Robustness	Length-consistency / latency curves; position-consistency curves; stability under noise/conflict/spelling perturbations	4.3/4.4/4.5	Curves + slope / half-life	Fix bins for length/position and perturbation density
Timeliness & Selective Answering	Freshness-Hit, Out-of-Date, selective accuracy, refusal rate, Brier/ACE	4.4/4.5	AURC (area under risk–coverage curve)	Timestamp & index version explicit; report an- swerable/unanswerable slices separately
Maintainability & Side Effects (Editing / Unlearning)	ESR (edit success rate), Locality (neighborhood preservation), Drawdown (general perf. drop)	4.2	Target / neighborhood / retained sets reported together	Report rollback behavior & degradation under sequential edits
Privacy & Memorization Risks	Verbatim reproduction rate, Exposure/Canary, training-data extraction, membership-inference AUC	4.2	Risk curves + alert thresholds	Disclose de-duplication & duplication-detection methods
Efficiency & Cost	Latency/turn, Tokens/turn, Throughput, Mem-footprint, Cost@Target	4.3/4.4/4.5	Performance–cost frontier	Specify hardware, decoding & caching strategies for reproducibility

Table 4.2: Comparison Dimensions — Recommended Plots and Statistical Notes

Comparison Dimension	Research Question	Recommended Plot	Statistical Notes
Ability vs Information Availability	Are conclusions driven by model capability or by evidence availability/freshness?	Side-by-side bars/lines with condition splits: PO / Offline / Online on the same slice	Paired tests + Holm/FDR correction; use the same time window and index version
Accuracy vs Groundedness	Is correctness supported by explicit evidence?	Dual-axis plot: EM/F1 together with UCR/FActScore on the same chart	Report both indicator families jointly; do not substitute with a single aggregate score
Quality vs Cost	Which option is more cost-effective?	Performance–cost frontier (e.g., Accuracy/F1 vs Cost@Target)	Disclose hardware, decoding and caching settings; report uncertainty bands
Stability vs Uncertainty	Do similar means hide volatility or bias?	Sensitivity curves: length/position/noise slices; normalized slope / half-life	Use unified binning/scripts; show 95% CIs and effect sizes

4.1.4 Data/Temporal Governance and Leakage Auditing: Minimum Reproducibility Disclosure(MRD)

This survey defines a Minimum Reproducibility Disclosure (MRD) schema (Table 4.3:Minimum Reproducibility Disclosure Checklist) to support auditability across data sources and time slices. The MRD is a lightweight, machine-readable record intended to enable like-for-like comparisons; it complements, rather than replaces, task-specific evaluation protocols.[4, 5, 21, 59–61, 115, 122]

Scope. Throughout § 4, MRD fields are used to annotate studies that assert capability changes or evaluate freshness. When certain fields are not reported, the omission is recorded and any available robustness or sensitivity analyses are noted.

Schema (minimum fields).

- **Temporal governance:** time windows and snapshot dates for training corpora, indices/session stores, and test sets; for freshness or online-retrieval settings, the most recent update time and update frequency.
- Leakage and overlap auditing: qualitative/quantitative characterization of training—index—test overlap or near-duplicates, including data sources, detection methods and thresholds, exclusion criteria, and impact fraction; discussion of potential benchmark contamination and mitigation for public benchmarks.
- **Implementation and resources:** details sufficient to contextualize results and support replication, including model/checkpoint versions and decoding hyperparameters; retriever/reranker types and core parameters (e.g., *k*, fusion strategy); hardware scale and per-unit cost (or token budget); random seeds and script versions.

Availability. A YAML template instantiating this MRD schema is provided in Appendix (Appendices A); this template supports consistent reporting and enables diff-based comparisons across regimes (PO/offline/online).

Dimension	Required Fields	Example / Format		
Temporal Governance	Training window; index / conversation store snapshot; test set sampling time	Training: 2022-01–2024-03; Index: 2025-05-01; Test: 2025-06 rolling		
Freshness / Online	Update frequency; definition of unanswerable cases	Web snapshot: 2025-07-15; Weekly updates; Unanswerable = no retrieval evidence		
Leakage Auditing	Deduplication methods and thresholds; overlap ratio; reference contamination explanation	MinHash Jaccard ≥0.8; Overlap ≤0.7%; Public benchmark Dec 2021 version		
Model & Parameters	Model / checkpoint; decoding; randomness seed	Llama-3.1-70B-instruct; T=0.2, top-p=0.9; seed=2025		
Retrieval / Re-ranking	$ \left \begin{array}{l} {\rm Retriever;} \ k \ {\rm / \ fusion;} \ {\rm re\text{-}ranking \ configuration} \\ \end{array} \right $	Contriever; $k=20$, RRF; Cross-Encodermsmarco		
Memory Settings	Operation mode; replay budget; refusal threshold	SR-off; ReplayBudget=6 turns; RefusalThreshold=0.7		
Resources & Cost	Hardware; unit cost; throughput	8×A100 80GB; \$X/1k tokens; Y req/s		
Limitations Statement	Key threats and sensitivities	Length >128k: not evaluated; position sensitivity		

Table 4.3: Minimum Reproducibility Disclosure Checklist

4.1.5 Review, Statistics, and Implementation Coupling: Reporting Standards and Pitfalls

To enhance auditability and cross-study comparability, we set unified reporting standards for the review process, statistical inference, and implementation-coupled factors. Unless otherwise specified, any claim of improvement/degradation, freshness-related effects, or stability differences must satisfy the following minimum requirements.

(A) Review and uncertainty.

Rater consistency. When using automated adjudication (LLM-as-a-judge, NLI-based decisions, etc.), report inter-rater or intra-rater (prompt/temperature) agreement (e.g., Cohen's κ , Krippendorff's α) with 95% CIs (bootstrap/stratified bootstrap).

Sampling and dual review. Key conclusions must be cross-checked by human spot checks (≥ 2 raters) with agreement statistics and arbitration rules.

Score-drift control. Fix the adjudicator model's version and prompts across batches, and conduct a sensitivity analysis of scoring drift.[28, 118, 123]

(B) Statistical inference and multiple comparisons.

Hypothesis testing. For cross-model/regime comparisons, use paired permutation or paired bootstrap tests and report p-values, 95% CIs, and effect sizes (Cohen's d, Cliff's δ).

Multiple correction. For simultaneous comparisons across tasks/slices/metrics, specify the multiple-comparison procedure (Holm–Bonferroni or FDR) and the family-wise error domain.

Unit of aggregation. Specify the aggregation level (sample macro/micro average, document-level, or session-level) and discuss its impact on uncertainty; for cross-session tasks, prefer session as the aggregation unit with stratified sampling.

(C) Implementation coupling and sensitivity control.

Long context and positional strategy. Report window size, positional encoding/truncation rules, and cache settings; for long-context tasks, provide length–performance/latency and position–performance sensitivity curves. [63–66, 68, 69, 71]

Attention/cache/compression. When using Flash/Streaming attention or KV clustering/compression, provide sensitivity regressions under equal-budget or equal-accuracy conditions to avoid mistaking engineering configuration differences for "memory capability differences." [79, 80, 84–86, 89, 124–126]

Retrieval pipeline. When end-to-end improvements are limited, report joint changes in retrieval recall (Recall@k), ranking quality (nDCG/MRR), and faithfulness (e.g., FActScore, Unsupported Claim Rate, UCR) to locate bottlenecks. [17, 18, 20–22, 96, 121, 127, 128]

Regime alignment. When attributing to "capability vs. information availability," provide at least one same-domain, same-slice PO or Offline control (see §4.1.2) and state alignment fields (snapshot date, k, replay budget, refusal threshold). [115]

This standard uses agreement/uncertainty to control adjudicator bias; paired tests + effect sizes + multiple correction to curb spurious advantages; and sensitivity controls to decouple engineering details from memory performance. It provides a verifiable common baseline for horizontal reading and limited comparisons across §§4.2–4.5.

4.2 Evaluation of Parametric Memory

In the evolution of LLMs from "language understanding" to "knowledge application", parametric memory serves as a core bridge connecting model capabilities to practical needs. It refers to the ability of models to implicitly encode massive facts, common sense, and associated knowledge into their parameters through large-scale pre-training—forming the foundation for models to perform tasks such as question answering, reasoning, and fact generation without relying on external knowledge bases. Compared with the traditional "external retrieval + generation" paradigm, parametric memory offers three irreplaceable advantages: 1. Higher response efficiency: It eliminates the need for real-time calls to external databases, making it suitable for low-latency scenarios like dialogue and real-time question answering; 2. Stronger robustness: It is not affected by noise in external data or retrieval biases, as validated by factual consistency evaluation frameworks [111]; 3. Support for complex knowledge integration: It can associate scattered facts into structured cognition (e.g., inferring "Li Bai was a famous poet of the Tang Dynasty" from "Li Bai was a poet" and "Li Bai lived in the Tang Dynasty"), a capability critical for handling complex tasks per systematic factual knowledge assessments [90].

However, the "implicitness" of parametric memory also poses two key challenges: - On one hand, there is doubt about whether the model truly "masters" knowledge. Models may form spurious correlations through co-occurrence patterns in corpora (e.g., frequent co-occurrence of "Dante" and "Florence") rather than understanding the logical basis of facts—a limitation highlighted by holistic LLM evaluation [116]; - On the other hand, knowledge "maintainability" is insufficient. When facts are updated (e.g., the change of a country's leader) or contain errors, precise modifications cannot be made like in structured knowledge bases. Instead, full-model retraining is required, which incurs extremely high costs—a problem emphasized by dynamic benchmarking research [146].

These challenges have given rise to parametric memory evaluation, which demands systematic methods to verify four dimensions of model memory: - *Accuracy*: Whether correct facts are stored; - *Localizability*: Which modules store the knowledge; - *Editability*: Whether knowledge can be precisely modified; - *Consistency*: Whether associated knowledge is affected after editing.

Such evaluation provides a basis for model optimization and scenario-specific deployment.

Recent research has made significant progress toward these goals: from early parametric memory only (verifying whether models can serve as knowledge bases), to studies on addressability (locating knowledge storage modules and enabling precise editing), and further to evaluating the ripple effects of knowledge editing (focusing on the consistency of associated knowledge after edits). Parametric memory evaluation has now formed a complete workflow covering "memory-location-editing-consistency". The following discussion will focus on two directions: closed-book fact recall, and addressability & edit differential. To further enhance the evaluation framework, an extended metrics system is introduced, including advanced indicators for long-term retention, knowledge interplay, and factual grounding.

4.2.1 Parametric memory only

Parametric memory only focuses on "whether models can retrieve facts using only information stored in their parameters, without assistance from external knowledge bases". Its core objective is to address the question: "Can pre-trained models function as potential knowledge bases?" Traditional evaluation methods mostly rely on comparisons with structured knowledge bases, while this direction directly tests the model's implicit memory of facts through the design of "fill-in-the-blank" tasks. A representative study in this area is the paper *Language Models as Knowledge Bases?*.

The core contribution of this paper is to challenge the perception that "models can only capture linguistic patterns". It is the first work to systematically analyze the relational knowledge that can be accessed in state-of-the-art pre-trained language models (e.g., BERT, ELMo) without fine-tuning, treating these models as "unsupervised knowledge bases" and evaluating their ability to recall facts and common sense. As stated in the paper: "We present an in-depth analysis of the relational knowledge already present (without fine-tuning) in a wide range of state-of-the-art pretrained language models."

To conduct this evaluation, the authors proposed the LAMA (LAnguage Model Analysis) probe framework: facts are transformed into "fill-in-the-blank" cloze sentences (e.g., "Dante was born in [Mask]"), and the model's memory of the fact is determined by the ranking of its predictions for the masked token. The paper explains: "For the purpose of answering the above questions we introduce the LAMA probe, consisting of a set of knowledge sources, each comprised of a set of facts."

Methodology Design and Experimental Setup 1. Multi-source dataset construction: To cover different types of knowledge, facts were extracted from four sources and converted into cloze format to ensure comprehensive evaluation: - Google-RE: Manually extracted entity-relation facts aligned with Wikipedia text, guaranteeing the authenticity and textual relevance of facts; - T-REx: Automatically aligned facts from a subset of Wikidata, featuring larger scale and broader entity types; - ConceptNet: Common sense relations (e.g., "Birds can fly") extracted from OMCS sentences, testing the model's memory of unstructured common sense; - SQuAD: An open-domain question-answering dataset manually converted into cloze format, verifying the model's fact recall in question-answering scenarios.

- 2. Model and baseline selection: Six mainstream pre-trained models were evaluated, including fairseq-fcony, Transformer-XL (large), ELMo (original and 5.5B variants), BERT-base, and BERT-large. A unified vocabulary was used to ensure fair comparison. Three types of traditional methods were set as baselines: Frequency baseline (Freq): Predictions based on token occurrence frequency in the corpus, verifying whether the model relies on simple statistics rather than knowledge; Relation Extraction (RE) model: The pre-trained RE model by Sorokin and Gurevych (2017), representing traditional structured knowledge extraction methods; DrQA: An open-domain question-answering system, representing the question-answering paradigm that requires external text retrieval.
- 3. Core evaluation metric: Mean precision at k (P@k) was adopted, calculated only for single-token targets. It represents the probability that the correct factual answer is among the top-k tokens predicted by the model, directly reflecting the precision of the model's fact recall.

Key Results and Findings The experimental results challenged the conventional view that "models require fine-tuning to handle knowledge". The core conclusions are as follows: 1. BERT-large achieves optimal performance: It outperformed other models across all knowledge-type tasks, with significant advantages in entity-relation and common sense recall. For 1-to-1 relations in T-REx (e.g., "someone's birthday"), its P@1 reached 74.5%, approaching the performance of traditional RE models combined with oracle entity linking (33.8% vs. 32.3%); in ConceptNet common sense tasks, its P@1 reached 19.2%, demonstrating the model's ability to memorize unstructured common sense. 2.

Comparable to traditional methods: In SQuAD question-answering tasks, BERT-large achieved a P@10 of 57.1%, narrowing the gap with DrQA (which requires external text retrieval, achieving 63.5%); in Google-RE tasks, its P@1 (10.5%) surpassed the baseline of traditional RE models (7.6%), indicating that parametric memory can achieve performance close to structured methods without supervision. 3. Robustness advantages: BERT showed greater robustness to variations in query phrasing. For example, its prediction consistency was higher between "Dante's birthplace is [Mask]" and "Dante was born in [Mask]". Additionally, the model's prediction confidence was positively correlated with accuracy, providing a basis for "judging whether the model is 'confident' in a fact" in practical applications.

Conclusions and Implications The authors put forward a core viewpoint: Pre-trained language models have the potential to serve as "unsupervised open-domain QA systems", and their parametric memory can supplement or even replace traditional knowledge bases. As noted in the paper: "Language models have many advantages over structured knowledge bases: they require no schema engineering, allow practitioners to query about an open class of relations, and require no human supervision to train."

However, this method still has limitations: - It only supports single-token target recall and cannot handle multi-token answers (e.g., "New York City"); - It performs poorly on N-to-M relations (e.g., "multiple works by an author"); - Models may learn associations through co-occurrence patterns in corpora (e.g., frequent co-occurrence of "Dante" and "Florence") rather than truly "understanding" the logical basis of facts.

Future improvements could focus on multi-token prediction, automatic template generation (to reduce biases from manual design), and multilingual knowledge evaluation—with memory-augmented architectures potentially addressing multi-token limitations.

4.2.2 Addressability & Edit Differential

Parametric memory only verifies the "existence" of parametric memory, while addressability & edit differential further explores the "localizability of memory"—i.e., whether the specific modules storing facts in the model can be identified, and whether facts can be precisely updated by modifying parameters (instead of retraining the entire model). This direction needs to address two issues: the storage location of facts in the model, and how to avoid interfering with other knowledge when editing a single fact. Representative studies include three core papers: *Locating and Editing Factual Associations in GPT, Rebuilding ROME: Resolving Model Collapse during Sequential Model Editing*, and *Evaluating the Ripple Effects of Knowledge Editing in Language Models*.

1. Fact Localization and Editing: Proposal of the ROME Method The paper *Locating and Editing Factual Associations in GPT* was the first to prove that factual associations in autoregressive transformer models (such as GPT) are stored in locally editable computational modules, rather than being scattered across the entire network. This provides a theoretical basis for "precise editing" of parametric memory [145]. As stated in the paper: "We analyze the storage and recall of factual associations in autoregressive transformer language models, finding evidence that these associations correspond to localized, directly-editable computations."

Core Methodology: Causal Tracing and ROME Editing - Causal Tracing: This method locates knowledge storage modules through causal mediation analysis, involving three model runs: 1. Clean run: Input a complete factual prompt (e.g., "Paris is the capital of [Mask]") and record the model's internal hidden states; 2. Corrupted run: Replace the topic word with an irrelevant word (e.g., "Xyz is the capital of [Mask]") and observe the model's prediction bias; 3. Corrupted-with-restoration run: During the corrupted run, restore only the hidden state of a specific module in a specific layer to the result from the clean run. If the model's prediction accuracy recovers, this module is identified as critical for storing the fact.

The Average Indirect Effect (AIE) is calculated to quantify the module's contribution—the higher the AIE, the stronger the causal impact of the module on fact recall.

- ROME (Rank-One Model Editing): Based on the results of causal tracing, a "rank-one update" method is proposed to modify facts. For middle-layer MLP modules (which causal tracing identified as having the highest AIE), the weight matrix is updated with a rank-one matrix to precisely replace a specific fact (e.g., changing "Paris is the capital of France" to "Paris is the capital of Japan") without modifying other parameters.

Key Findings 1. Middle-layer MLPs are the core of fact storage: Causal tracing showed that middle-layer MLP modules (e.g., layer 15 in GPT) had the highest AIE (reaching 8.7%) when processing the "last token of the topic", while the contribution of the attention mechanism was only 1.6%—proving that factual associations are mainly stored in MLPs rather than attention layers [115]. 2. Effectiveness of ROME editing: In the zero-shot relation extraction

(zsRE) task, ROME achieved performance comparable to fine-tuning and meta-learning methods. More importantly, on counterfactual datasets (facts not seen during pre-training, such as "Paris is the capital of Japan"), ROME maintained both "generalization" (adaptation to different prompt phrasings) and "specificity" (no interference with irrelevant facts), whereas previous methods had to sacrifice one or the other [145]. As noted in the paper: "ROME achieves good generalization and specificity simultaneously, whereas previous approaches sacrifice one or the other."

2. Editing Stability: Resolution of Model Collapse by r-ROME The paper *Rebuilding ROME: Resolving Model Collapse during Sequential Model Editing* points out that original ROME suffers from "model collapse" during sequential editing (batch updating of multiple facts). Some edits (referred to as "disabling edits") cause a sharp decline in the model's generation ability (e.g., outputting repeated text). This problem stems from flaws in implementation details rather than ROME's core logic .

Root Cause of Collapse and Remedial Solutions - Cause of collapse: Original ROME used key vectors asymmetrically in the update equation—mixing "average prefix key vectors" and "original prompt key vectors" when calculating the update matrix. This led to abnormally large L2 norms of some updates (an order of magnitude higher than normal updates), causing parameter oscillations in the model. - Remedial solutions: Two improved implementations were proposed: - r-ROME: Consistently uses "average prefix key vectors" in the update equation to ensure unified calculation logic; - p-ROME: Consistently uses "original prompt key vectors" to improve the accuracy of single-fact editing.

Key Results 1. Complete resolution of collapse: The L2 norm of the update matrix in r-ROME was an order of magnitude smaller than that in original ROME. In 5,000 sequential edits on the CounterFact (counterfactual dataset), no model collapse occurred, whereas original ROME failed within 100 edits [145]. 2. Performance improvement: r-ROME achieved a higher overall score (92.22) in single edits than original ROME (89.32), with stronger "locality" in editing (less impact on irrelevant modules). 3. Inherent limitation: r-ROME still cannot solve the "progressive degradation of sequential editing"—as the number of edits increases, the model's performance on downstream tasks (e.g., GLUE) slowly declines, indicating that the editing capacity of parametric memory has an upper limit.

3. Editing Consistency: Evaluation of Ripple Effects The paper *Evaluating the Ripple Effects of Knowledge Editing in Language Models* points out that existing evaluations only focus on "whether the target fact is correctly edited" and ignore ripple effects—i.e., the impact of editing one fact on associated facts (e.g., after editing "Paris is the capital of France", can the model synchronously update associated knowledge such as "The capital of France is Paris" and "Paris belongs to France"?). To address this gap, the authors proposed the RIPPLEEDITS evaluation framework, filling the void in "editing consistency" evaluation [129].

Evaluation Framework Design 1. Six evaluation criteria: Comprehensively covering core dimensions of ripple effects: - Logical generalization: Whether the model can correctly infer associated facts after editing (e.g., after editing "A is the capital of B", can it correctly answer "What is the capital of B"?); - Compositionality I/II: Whether the model can handle combinations of multiple facts (e.g., after editing "A is in B" and "B is in C", can it infer "A is in C"?); - Topic aliases: Whether consistency is maintained across different phrasings of the topic (e.g., "Paris" and "The City of Light"); - Preservation: Whether irrelevant facts remain unaffected (e.g., editing "Paris is the capital of Japan" should not affect "London is the capital of the United Kingdom"); - Relation specificity: Whether only the target relation is modified, without affecting other relations (e.g., editing "the country where Paris is located" should not affect "the population of Paris").

- 2. RIPPLEEDITS dataset: Contains 5K fact edits, with 10–15 associated test queries per edit. It covers metadata such as "recent/old facts" and "head/tail entities", enabling analysis of ripple effects in different scenarios.
- 3. Method comparison: Three parametric editing methods (MEND, ROME, MEMIT) and one contextual editing baseline (which modifies facts through contextual prompts without parameter updates) were evaluated.

Key Findings 1. Shortcomings of existing methods in ripple effects: Parametric methods (e.g., ROME) perform well in single-fact editing but achieve an average score of less than 50% in ripple effect tasks—with high failure rates particularly in logical generalization and compositionality tasks. This proves that models cannot synchronously update associated knowledge. 2. Superiority of contextual editing: Contextual editing (which does not require parameter updates, e.g., adding "It is known that Paris is the capital of Japan; please answer..." to the prompt) achieved the highest score in RIPPLEEDITS. This is because it can directly associate facts through context, avoiding the cascading effects of parameter modifications. 3. Influencing factors: - Model scale: Larger models (e.g., those with over 10B parameters) demonstrate stronger ability to handle ripple effects, proving that greater capacity leads to better knowledge integration; - Entity frequency: Editing "head entities" (e.g., Paris) is more likely to cause logical errors than editing

"tail entities" (e.g., niche cities). This is because models have stronger prior knowledge of head entities, making it harder to synchronously update knowledge after edits.

Table 4.4: Comparison of Papers on Knowledge Storage and Editing in Language Models

Paper	Models Context Lengtl & Task Design		Evaluation Metrics	Findings	Limitations
Language Models as Knowledge Bases?	fairseq-fconv; Transformer- XL; ELMo; BERT-base; BERT-large	Single-fact cloze; no long sequence	call: Google-RE,	BERT-large best; T-REx 1-to-1 P@1 ~74.5%; comparable to RE/DrQA; robust to query variation	Single-token only; poor on N- to-many; relies on co-occurrence
Locating and Editing Factual Associations in GPT	GPT family	Short prompts, final token focus	factual datasets;	Mid-layer MLPs store facts; ROME edits MLPs; better generalization/specificity	Single-layer edits; prompt- sensitive; small counterfactual set
Rebuilding ROME: Resolving Model Collapse during Sequential Model Editing [130]	GPT-2 XL; GPT-J (6B)	Short prompts; sequential edit- ing	liability, general-	ROME collapses due to key-vector asymmetry; r-ROME fixes it; supports \sim 5000 edits	Gradual degrada- tion persists; un- clear scalability; weak counterfac- tual generaliza- tion
Evaluating the Ripple Effects of Knowledge Edit- ing in Language Models [131]	,	Prompts include target + associ- ated facts	RIPPLEEDITS (5K edits); 6 task types; ripple- effect score	Parameter methods <50% ripple; prompt-based methods outperform; larger models + tail entities more robust	Prompt design critical; no multimodal; long-term effects not studied

4.2.3 Enhanced Evaluation Metrics Framework

The core workflow from 4.2.1 and 4.2.2 provides essential tools, but it leaves critical questions unanswered for practical deployment:

- How to quantify a model's reliance on its parametric memory versus contextual information in a hybrid setting?
- How to measure the longevity of knowledge and the impact of sequential edits over time?
- How to assess the interaction between internal memory and external information when they conflict?
- How to move beyond simple recall to evaluate the verifiability and grounding of the knowledge generated from parameters?

To answer these questions, we extend the core dimensions and introduce advanced metrics that quantify memory, long-term memory, knowledge interactions, and factual grounding. These enhancements build on existing metrics such as P@K and AIE to provide a more robust framework for LLM evaluation in dynamic and hybrid scenarios.

1. Parametric Proxy Rate (PPR): PPR directly extends the "Parametric Memory Only" principle of LAMA into practical QA settings. It measures the proportion of model responses based solely on parametric knowledge versus those aided by an external context (oracle). A very high PPR indicates strong memorization but may also signal a risk of generating outdated or hallucinated content when internal knowledge is incorrect, a problem that knowledge editing (from 4.2.2) aims to solve. Thus, PPR serves as a crucial baseline for assessing the need for and impact of editing techniques in application contexts.[132]

$$PPR(M) = \frac{Acc_M(random)}{Acc_M(oracle)},$$
(1)

where parameter responses are derived entirely from internal parameters. PPR helps identify situations where there is over-reliance on memorized data (which can lead to hallucinations) and, combined with edit evaluation, serves as a baseline for consistency checks.[132]

2. Long-Term Retention Score (LTRS): The sequential editing studies in 4.2.2 revealed that model performance degrades after many edits. LTRS formalizes this evaluation over time. It measures the stability of parametric

knowledge after simulated delays or update cycles, directly addressing the maintainability challenge. By combining LTRS with the RIPPLEEDITS framework, researchers can now evaluate not just the immediate consistency of an edit, but how both the target fact and its associated knowledge ripples decay over time, providing a dynamic view of model knowledge health. [112]

$$LTRS = \frac{Delayed Recall Accuracy}{Initial Recall Accuracy} \times (1 - Decay Factor)$$
 (2)

with decay factor derived from temporal benchmarks. This metric extends maintainability assessment, is particularly applicable to dynamic knowledge scenarios, and can be combined with RIPPLEEDITS for ripple effect analysis.[112]

3. Knowledge Interaction Ratio (KIR): Findings from RIPPLEEDITS showed that contextual information can sometimes override parametric knowledge more effectively than parametric edits can. KIR quantifies this interaction explicitly. It measures the efficiency of a model's integration of parametric knowledge (PK) and contextual knowledge (CK). A poorly balanced KIR might indicate that a model is overly rigid (ignoring new context) or overly susceptible (discarding correct internal knowledge), thus evaluating the robustness of the hybrid memory system that is central to modern LLM applications.[133]

$$KIR = \frac{\text{Hybrid Response Accuracy} - \text{Pure Parametric Accuracy}}{\text{Contextual Contribution Weight}}$$
 (3)

It addresses hybrid memory challenges, such as when context overrides parameters, and enhances robustness evaluation in hybrid mechanisms.[133]

4. Factual Grounding Score (FGS): while LAMA and ROME evaluate factual accuracy against a known benchmark, FGS expands the concept of accuracy to trustworthiness in open-ended generation. It measures the percentage of a model's factual claims that can be verified against external sources. This metric is critical for mitigating the risk of "confabulation" from parametric memory, ensuring that the model's knowledge outputs are not just plausible but also grounded and reliable.[134]

$$FGS = \frac{Number of Supported Sentences}{Total Sentences with Factual Information} \times 100\%$$
 (4)

The score expands the accuracy dimension to include grounding against external documents, mitigating illusions in parameter output.[134]

The enhanced framework is the culmination of the parametric memory evaluation story. It takes the core concepts—existence (4.2.1), control (4.2.2), and consistency (4.2.2)—and scales them to meet the complexities of real-world deployment. PPR assesses reliance on memory, LTRS monitors its stability, KIR evaluates its interaction with new information, and FGS certifies the quality of its output. Together, they form an integrated ecosystem that allows researchers and developers to move beyond isolated tests and toward a holistic, operational assessment of knowledge in LLMs.

4.3 Evaluation of Contextual Memory

With the continuous extension of context windows in LLMs, assessing their capacity to retain and exploit information over long sequences has emerged as a central research challenge. Existing studies suggest that contextual memory can be examined across three stages: input, reasoning, and output. At the input stage, evaluations focus on information retention under varying context lengths and positional conditions. Tasks such as key evidence retrieval and mid-sequence extraction highlight the models' sensitivity to token position in long texts. The reasoning stage emphasizes the ability to integrate retained evidence for pattern alignment and rule induction, with particular attention to cross-span integration, chain-of-thought reasoning, and robustness to distractors—factors that largely determine whether models can effectively leverage contextual information. The output stage assesses the alignment and faithfulness of generated responses to the input evidence, using metrics such as factual consistency and completeness of citation chains to distinguish genuine context usage from hallucinatory completion. This layered perspective moves beyond single accuracy-based metrics and provides a more interpretable and reproducible framework for evaluating contextual memory in LLMs.

4.3.1 Evaluation Dimensions and Key Metrics

1. Input Stage A number of existing long-context evaluation studies explicitly focus on memory retention and information localization in the input stage. Representative tasks include key evidence retrieval and mid-sequence extraction, designed to expose models' deficiencies in positional sensitivity when processing long texts. This phenomenon was first

revealed by Lost in the Middle [63], which showed in multi-document QA and key-value retrieval tasks that mainstream models exhibit a sharp decline in utilizing middle-position information, forming a characteristic "U-shaped curve". The primary evaluation metrics are Answer Accuracy and Evidence Recall, defined as:

$$Accuracy = \frac{Number of correct answers}{Total number of questions},$$
 (5)

$$Recall = \frac{|E_{gold} \cap E_{pred}|}{|E_{gold}|},$$
(6)

where $E_{\rm gold}$ denotes the annotated evidence set and $E_{\rm pred}$ the evidence identified by the model. Building on this, Found in the Middle [64] analyzed the problem at the mechanism level, attributing it to attention biases toward positional information. It proposed a calibration method to decouple position from relevance, achieving up to 15 percentage points of improvement in mid-span evidence utilization. Its evaluation was based on Evidence F1:

Evidence-
$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$
 (7)

These diagnostic studies highlight that input-stage evaluation should go beyond overall performance, aiming to reveal differential behaviors across positions. In line with this, the Needle-in-a-Haystack Test [135] embeds a unique key segment into ultra-long text to directly test retrieval ability. Its core metric is Hit Rate:

$$HitRate = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1} \{ pred_i = gold_i \}.$$
 (8)

Other works, such as Position Interpolation [81], investigate input fidelity from the perspective of extended positional encoding, adopting metrics such as Perplexity (PPL) and Cross-Position Consistency. PPL measures the uncertainty of the model's predictive distribution given an input position, with lower values indicating greater stability, while Cross-Position Consistency evaluates the consistency of predictions when the same semantic span is placed at different positions. Comprehensive benchmarks, including LongBench [66] and InfiniteBench [68], also incorporate input-stage subtasks such as fact retrieval and span extraction. Their evaluation relies on metrics including Exact Match (EM), which checks whether predictions exactly match the reference, ROUGE-L, which captures the longest common subsequence between prediction and reference, and Evidence Recall, thereby assessing both answer correctness and evidence usage. In Chinese contexts, Bamboo [70] similarly includes retrieval and extraction tasks to evaluate evidence recall under ultra-long input conditions. In addition, benchmarks such as LV-Eval [71] construct tiered input lengths (16k–256k) with inserted distractors to expose degradation curves and vulnerability to noise under extreme lengths. The evaluation employs Length-Normalized Accuracy (LN-Acc) and Noise Robustness Score (Robustness):

$$LN-Acc(L) = \frac{Acc(L)}{\log(L)},$$
(9)

$$Robustness = 1 - \frac{Acc_{with \ noise}}{Acc_{clean}}.$$
 (10)

Overall, these studies place less emphasis on complex reasoning, centering instead on whether models can accurately capture and reproduce key information in long inputs—thereby providing a reliable memory foundation for subsequent reasoning and generation stages.

2. Evaluation of the Intermediate Stage In the evaluation of intermediate-stage contextual memory, the focus shifts from simple information retention to evidence integration, logical reasoning, and robustness against interference. Early work such as Chain-of-Thought [136] introduced explicit reasoning chains to guide step-by-step inference. A common evaluation task in this line is step-wise QA, with metrics including final answer accuracy (Answer Accuracy) and step coverage, defined as:

$$StepCoverage = \frac{|S_{gold} \cap S_{pred}|}{|S_{gold}|},$$
(11)

where S_{gold} denotes the set of annotated reasoning steps. Subsequent studies, such as Faithfulness vs. Plausibility [118], emphasize verifying whether intermediate reasoning chains genuinely rely on input evidence rather than hallucinated completions. Their core task, explanation verification, adopts metrics including the Faithfulness Score (whether explanations are supported by explicit evidence) and the Plausibility Score (whether explanations are semantically reasonable). Both are typically judged by natural language inference (NLI) classifiers:

$$Faithfulness = \frac{|E_{\text{supported}}|}{|E_{\text{all}}|},$$
(12)

where $E_{\text{supported}}$ denotes explanation units that can be grounded in the input. From a benchmark perspective, SCROLLS [69] provides cross-document QA, summarization, and NLI tasks, with metrics covering Exact Match (EM), ROUGE-L, and Entailment Accuracy. Entailment Accuracy evaluates whether generated answers are semantically entailed by the reference answers, typically determined using NLI models. LongBench extends beyond needle retrieval by introducing multi-hop QA and cross-paragraph reasoning tasks, assessed with F1 and EM, focusing on whether models can effectively integrate evidence in large-scale contexts. In the Chinese context, Bamboo incorporates multi-evidence integration and chain-of-reasoning QA into its task set, with evaluation based on Evidence F1 and Multi-hop Answer Accuracy, highlighting the model's ability to induce and reason over context under ultra-long input conditions. Unlike standard Answer Accuracy, this metric requires correct answers to be derived from multiple evidence fragments. RULER [65], by contrast, places particular emphasis on robustness in the presence of distractors and noise. It adopts the RobustQA task with the following metric:

$$RobustAcc = \frac{N_{correct under noise}}{N_{total examples}},$$
(13)

Correct under noise measuring whether models can still extract and use key information correctly under redundancy or conflicting input. Going further, TimeQA [2] integrates contextual memory with temporal reasoning, proposing a time-sensitive QA framework with 5.5K facts and 20K QA pairs, distinguishing between explicit and implicit reasoning. Its core metric, Temporal Consistency, is defined as:

$$TempConsist = \frac{N_{consistent answers}}{N_{total temporal QA}}.$$
(14)

QAconsistent answers highlighting limitations of long-context models in maintaining temporal consistency and robustness. This work broadens the evaluation of the intermediate stage, extending beyond cross-paragraph integration to the use of dynamically evolving knowledge. Although not a long-context benchmark, EntailmentBank [119] contributes valuable insights by verifying stepwise reasoning chains through entailment checks. It defines Entailment Accuracy as:

$$Acc_{entail} = \frac{|H_{entailed}|}{|H_{all}|},$$
(15)

where H_{entailed} denotes the set of intermediate hypotheses correctly derived by the model. Overall, evaluation tasks and metrics in this stage converge on three key questions: how models integrate context (multi-hop and chain reasoning), how intermediate reasoning chains can be validated (faithfulness, entailment, and stepwise reasoning), and how robust models are against interference (distractors and conflicting evidence). Together, these assessments complement input retention metrics by establishing a comprehensive framework for evaluating intermediate-stage capabilities.

3.Evaluation of the Output Stage Building on the input stage, which focuses on whether models can accurately read and locate key information, and the intermediate stage, which emphasizes evidence integration and reliable reasoning, the evaluation of the output stage further shifts toward assessing the faithfulness and attributability of generated results. Representative work such as QAGS [114] targets summarization faithfulness evaluation by constructing QA pairs to check whether generated summaries align with the source text. Its core metric is QA-F1, defined as:

$$F1_{QA} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}},$$
(16)

where precision and recall are measured based on the alignment between answers and source evidence. RARR (Gao et al., 2022) [137] introduces verification of generated content using external knowledge bases or retrieval mechanisms. Within the retrieval-augmented QA setting, it defines Attribution Accuracy:

$$AttrAcc = \frac{N_{answers with correct source}}{N_{total answers}},$$
(17)

highlighting attribution and verifiability as key dimensions. FactCC [138] addresses factual error detection in summarization by formulating a binary classification task (factually correct vs. incorrect), evaluated with Accuracy and F1. FactScore [26] builds on a QA-based verification task, generating questions and answering them against the source text, with consistency measured by QA-F1:

$$FactScore = \frac{|Q_{correct}|}{|Q_{all}|},$$
(18)

thus establishing factual consistency as a central metric for the output stage. In retrieval-augmented generation, RAGAs [19] propose a set of metrics—Answer Faithfulness, Context Precision/Recall, and Answer Relevancy—to assess the alignment between generated output and contextual evidence:

- Answer Faithfulness (AF): whether the output is fully supported by the context.
- Context Precision (CP) and Context Recall (CR):

$$CP = \frac{|C_{used} \cap C_{gold}|}{|C_{used}|}, CR = \frac{|C_{used} \cap C_{gold}|}{|C_{gold}|},$$
(19)

• Answer Relevancy (AR): whether the output directly addresses the user's question.

The TRUE Benchmark [111] aggregates factual consistency tasks from summarization, QA, and dialogue, and unifies evaluation through Fact Consistency Accuracy and QA-F1, thereby providing a cross-task reliability testing platform. For scientific QA and citation generation, Luo et al. [90] introduce the Attribution & Citation framework, with the core metric Citation Recall:

$$CitationRecall = \frac{|R_{pred} \cap R_{gold}|}{|R_{gold}|},$$
(20)

which requires models not only to generate answers but also to provide verifiable citation chains rather than hallucinated responses. In addition, comprehensive benchmarks such as HELMET [67] categorize long-context evaluation into seven application scenarios, introducing length-controlled experiments and model-based scoring, with emphasis on factual consistency and citation completeness. Similarly, Bamboo and LV-Eval extend beyond input and intermediate tasks to include faithfulness checks at the output level, with tasks such as long-document summarization and QA, evaluated by metrics including Fact-F1, ROUGE-L, and Faithfulness Score, thereby achieving end-to-end coverage. Overall, evaluations at this stage primarily target dimensions of faithfulness, attribution/citation, coverage, robustness, and quality (e.g., fluency and relevance). The central question is no longer whether information is remembered or whether reasoning is correct, but whether the model can faithfully and verifiably reproduce and cite contextual information at the generation level.

4.3.2 Benchmark Comparison

After outlining the evaluation dimensions across the input, intermediate processing, and output stages, comparing existing benchmarks plays a bridging role. On the one hand, these benchmarks provide standardized measures across models and tasks, enabling comparisons of input capacity limits, evidence integration and reasoning chains, as well as output faithfulness and robustness within a unified framework. On the other hand, their differences in coverage, metric orientation, and task design reveal both the evolutionary logic of evaluation systems and the gaps yet to be addressed.

From the perspective of the input stage, LongBenchV2 and Bamboo employ multi-task settings (QA, summarization, code completion) and variable sequence lengths to expose degradation curves in information capacity. However, they still have shortcomings in capturing positional sensitivity and cross-document multi-hop reasoning. HELMET further scales to the 128K level, introducing model-based scoring and citation consistency, thereby highlighting standardization and comparability in evaluation.

At the intermediate processing stage, LV-Eval stands out by systematically introducing distractor facts and keyword substitutions, directly stress-testing model robustness and clearly exposing vulnerabilities in reasoning chains. Although Needle-in-a-Haystack focuses on a single task, it pushes the boundaries of "retrieving key information" to an extreme, serving as a crucial complement for assessing information utilization and resilience to noise.

At the output stage, SCROLLS centers on document-level tasks, emphasizing the faithfulness and compressive quality of generated content. In contrast, HELM [115] and Dynabench [116] elevate the perspective, focusing on evaluation procedures and data construction mechanisms. They propose dynamic, adversarial, and multi-dimensional metrics to mitigate the limitations of single-task benchmarks in robustness and interpretability.

Overall, the side-by-side presentation of these benchmarks helps clarify the hierarchical characteristics of capabilities across the input-intermediate-output stages while also revealing the complementarities and shortcomings of the current landscape: some emphasize capacity and retrieval, others focus on reasoning and robustness, and still others on generative faithfulness and evaluation workflows. Such diversity makes comparative tables not only a means of information synthesis but also a foundation for building more comprehensive evaluation frameworks.

Table 4.5: Long-Context Benchmarks — Models, Tasks, Metrics, Findings, and Limitations

Paper Title	Model	Context Length Range	Task Design	Evaluation Metrics	Results and Findings	Limitations
Needle-in-a- Haystack	open-source and closed-source models (e.g., GPT-3.5, Llama series, Claude)	Context length scal- able to 100K+	Insert key to- kens into ultra- long contexts, requiring the model to pre- cisely locate and respond amid distracting noise	Precision, Recall, Posi- tional Accu- racy	Most models exhibit "forgetting" or po- sitional drift in ex- tremely long inputs	Tasks too narrow; lack reasoning/- generation; not reflective of real- world scenarios
SCROLLS	Document-level models such as T5, Longformer, LED, BigBird	5K-50K	Cross-task suite including long-document QA, summarization, and natural language inference (NLI)	Accuracy, Entailment	Document-level tasks test models' abilities in information com- pression and cross- passage integration; different tasks expose model weaknesses in distinct ways	
HELM	30+ models including GPT-3, OPT, BLOOM, Jurassic-1	2K-32K	Generalized task set (QA, summarization, dialogue, rea- soning) with holistic evalua- tion	Accuracy, Calibration, Robustness, Fairness, Effi- ciency	Proposes the first framework evaluat- ing performance, fair- ness, robustness, and efficiency; closed- source models out- perform open-source across these metrics	
Dynabench	RoBERTa, T5, GPT-3, etc.	Mostly ≤4K	Tasks cover QA, NLI, and dialogue	Accuracy, F1, human discrimination accuracy	Model robustness drops significantly when facing adversar- ial data and dynamic distributions; continu- ous iteration exposes model blind spots	scalability chal-

Paper Title	Model	Context Length Range	Task Design	Evaluation Metrics	Results and Findings	Limitations
LongBenchV2	GPT-3.5-Turbo- 16k, Llama2- 7B-chat-4k, LongChat- v1.5-7B-32k, XGen-7B-8k, InternLM-7B-8k, ChatGLM2-6B, ChatGLM2-6B- 32k, Vicuna-v1.5- 7B-16k	4K, 8K	Single- Document QA, Multi- Document QA, Summarization, Few-shot Learn- ing, Synthetic Task, Code Completion		In long-context tasks, a performance gap between open-source models and commercial models. Models benefit from scaled positional embeddings and continued training on longer contexts	Rouge to eval- uate long-text
HELMET	59 long-context models (closed- and open-source)	8K, 16K, 32K, 64K, 128K	RAG, passage re-ranking, long-document QA, summariza- tion, many-shot in-context learn- ing, synthetic recall	Accuracy, model-	Open-source models trail in long-text and complex instruction tasks, with a widen- ing gap over context length; performance degradation is task- dependent, severe in reordering and citation generation	Model-based scoring has bias; tasks are English- only with limited multilingual/mul- timodal cover- age; 128K evalu- ation is costly
Bamboo	gpt-3.5-turbo-16k, Claude2-100k, ChatGLM2-32k, Vicuna-v1.5-16k, LongChat-v1.5- 16k	4K, 16K	tion detection, ranking, lan-	cination detec- tion accuracy, ranking accu-	Performance de- grades over long contexts; degrada- tion and hallucina- tion/ranking gaps vary by task; data control and task stan- dardization enable stable comparisons	Conservative length cover- age; insufficient memory ability tests; task variety lacks real-world relevance (e.g., evidence-based, cross-modal tasks)
LV-Eval	15 long-context models	16K, 32K, 64K, 128K, 256K	Single-hop and multi- hop QA over 11 bilingual datasets; adds distractor-fact insertion and keyword/phrase replacement	metrics with EM/F1; robustness via distractor facts/keyword	Sharp drops at 64K+ (stronger at 128K/256K); error spikes after distractor insertion; confusion and retrieval under long contexts weaker than expected	code; extreme- length evalua-

4.3.3 Open Challenges and Future Directions

Although current research on long-context memory evaluation has established a relatively systematic framework across the three stages of input capacity, reasoning utilization, and output faithfulness, several unresolved challenges remain:

Fragmentation and limited coverage of evaluation dimensions. At the input stage, most benchmarks emphasize evidence localization and information retrieval, highlighting positional sensitivity and length degradation, but they rarely consider cross-modal contexts or interactive inputs. In the intermediate stage, while multi-evidence integration and robustness designs have been introduced, they are often confined to static text tasks and thus fail to reflect temporal consistency and knowledge evolution in dynamic environments. At the output stage, faithfulness checking has made solid progress

in summarization and QA, yet existing metrics often fall short in more complex generative tasks such as dialogue, decision-making, or cross-domain citation.

Limited interpretability of metrics. Although refined indicators such as Evidence Recall, Faithfulness Score, and Attribution Accuracy have been proposed, these typically capture only pointwise correspondences (e.g., answer alignment or citation chain coverage) and lack quantification of reasoning path completeness or generation robustness. Current evaluations often expose performance differences without clarifying the reasons for model failures, thereby offering limited diagnostic and guidance value. Monolingual and static bias in benchmarks.

Mainstream benchmarks such as SCROLLS, LongBench, and HELMET are predominantly based on English corpora, with limited evaluation resources in Chinese and other languages. Moreover, most benchmarks are constructed using static datasets, making it difficult to capture the memory challenges posed by user interaction, real-time information updates, and cross-domain generalization. Disconnection between extreme-length testing and real applications. While benchmarks such as LV-Eval have extended input windows up to 256K tokens, many tasks remain synthetic in nature and diverge from real-world demands such as long-document retrieval, multi-section writing, or scientific paper comprehension. This tension between "stress testing" and "application-oriented evaluation" reduces the practical relevance of current metrics for deployment scenarios.

Lack of unified frameworks and holistic evaluation. Existing benchmarks emphasize different dimensions—capacity, robustness, or faithfulness—resulting in fragmented coverage. There is no unified end-to-end evaluation framework that integrates the input–reasoning–output chain, nor interoperability between task types and metric layers. Consequently, model performance across different benchmarks remains difficult to align or interpret.

Future directions for long-context memory evaluation may unfold along several paths: Integrated cross-stage evaluation: Develop unified benchmarks that span input, reasoning, and output, enabling holistic characterization of a model's "memory chain" rather than fragmented task-specific assessments. Process interpretability and causal diagnosis: Establish metric systems that can trace reasoning chains and validate intermediate states—for example, stepwise entailment checks or causal dependency analysis—to pinpoint where errors arise. Multilingual and multimodal expansion: Extend evaluation to Chinese, low-resource languages, and multimodal inputs such as text—image, code, and structured data, thereby enhancing coverage and practical utility. Dynamic and interactive evaluation: Inspired by adversarial approaches such as Dynabench, introduce human-in-the-loop evaluation, time-sensitive QA, and knowledge update scenarios to test long-term consistency and robustness under realistic conditions. Application-oriented task design: Align benchmarks with high-value application domains—such as retrieval-augmented generation, scientific writing assistance, or legal case analysis—ensuring that tasks and metrics provide actionable insights for model optimization and deployment.

4.4 Evaluation of External Memory

4.4.1 Evaluation Data, Methods, and Metrics

External memory systems mitigate the inherent limitations of parametric models—specifically in knowledge timeliness, traceability, and controllability—by incorporating updatable, non-parametric knowledge during inference. The typical pipeline "query rewriting \rightarrow retrieval/reranking \rightarrow reading \rightarrow generation" exhibits tightly coupled modules, wherein errors may propagate and amplify across stages. Consequently, evaluation must first ensure comparability and interpretability of end-to-end performance, followed by fine-grained diagnostics of critical components such as retrieval and generation. Furthermore, evaluation dimensions must continuously expand in alignment with technological advancements such as real-time responsiveness, multimodality, and domain specialization. Current research has gradually converged on a four-tier collaborative framework: "retrieval quality — generation quality — end-to-end performance — robustness and timeliness," implemented within unified datasets and benchmark ecosystems. Below, we present an integrated exposition of metrics, methods, and datasets under this framework.

- 1. Retrieval Quality: Dual Tracks of Static Relevance and Task Utility Retrievers and rerankers are regarded as the "performance ceiling" of RAG systems; downstream task accuracy exhibits near-linear dependence on retrieval quality. Any recall gap or ranking bias inevitably leads to irreversible degradation during generation [11].
- (1) Static relevance adopts standard zero-shot benchmarks such as BEIR and KILT, employing metrics including nDCG@k, MRR@k, Recall@k, and Precision@k for cross-system comparability. An nDCG@10 \geq 0.7 is commonly treated as a warning threshold for adequate ranking; in knowledge-intensive domains such as medicine and law, Recall@k \geq 80% is often set as a hard constraint to prevent loss of critical evidence prior to generation [7, 21, 41].
- (2) Task utility is quantified via "marginal contribution" $\frac{\Delta EM}{\Delta F_1}$: the greater the decline in downstream metrics upon removal of a single passage, the higher its practical value. eRAG series experiments demonstrate that reranking yields a

relative improvement of 6–12% in passage-level recall, simultaneously enhancing answer correctness and evidence faithfulness[139].

- (3) Logical coherence transcends surface-level semantics, evaluating internal document coherence and document-query alignment along factual, causal, and narrative chains—including information completeness, causal consistency, and contextual coherence. Due to the difficulty of automated parsing, this dimension still relies primarily on human scoring or task-oriented indirect metrics (e.g., reasoning accuracy of generated answers), representing the largest blind spot in automated evaluation [37, 117][5,6]. In summary, mainstream consensus advocates dual-track reporting in evaluation: one track preserves cross-system comparability via metrics such as nDCG@10, MRR@10, and Recall@100; the other directly measures practical contribution to final outputs using task-driven metrics like $\frac{\Delta EM}{\Delta F_1}$, thereby establishing a "dual-track" evaluation paradigm [37].
- 2. Generation Quality: Four Progressive Dimensions Evidence Attribution, Output Performance, Context Utilization, and Logical Interpretability Generation quality exhibits significant interaction effects with retrieval results: when retrieval nDCG@5 > 0.75, generation optimization can improve FActScore by approximately 14.2%; otherwise, gains diminish by over 62%. Thus, multidimensional evaluation from an end-to-end perspective is essential. (1) Evidence attribution and faithfulness verify that generated content strictly originates from retrieved evidence, suppressing pseudocitations and hallucinations. The ALCE framework quantifies citation-augmented generation across fluency, correctness, and citation quality; QAFactEval constructs QA chains to test output-context consistency; FActScore decomposes long-form text into atomic facts and computes the proportion supported by evidence—the higher the proportion, the more solid the factual foundation. RAGAS and ARES provide reference-free faithfulness scores, with ARES introducing confidence intervals to mitigate drift bias in LLM-based evaluators; FACTS Grounding and GaRAGe conduct rigorous groundedness evaluation based on span-level annotations. Given that correctness and faithfulness are not equivalent, it is recommended to report both metrics and supplement them with span-level precision and recall to avoid macro-level metrics masking local pseudo-citations.
- (2) Output quality and task performance must be assessed using task-specific metrics. Open-domain QA commonly employs EM and F1; long-form generation typically uses ROUGE or BLEU. However, surface overlap metrics may obscure evidence omission issues and must therefore be reported jointly with faithfulness metrics. RAGTruth provides word-level and span-level hallucination annotations; its hallucination density metric locates fabricated content within generations, offering fine-grained feedback for model refinement.
- (3) Context adequacy and selective answering examine system behavior under evidence-sufficient and evidence-deficient conditions. By partitioning test samples into "sufficient evidence" and "insufficient evidence" categories, one can evaluate accuracy under answerable conditions and refusal capability under unanswerable conditions. Selective accuracy measures the proportion of correct answers among answerable samples; refusal rate measures the proportion of correct refusals among unanswerable samples—both jointly reflect the system's risk control capability under uncertainty. The Sufficient Context framework recommends joint analysis of these metrics with retrieval recall: evaluating evidence utilization efficiency on the "sufficient" subset and assessing system restraint on the "insufficient" subset.
- (4) Logical interpretability examines whether generated content possesses traceable reasoning paths. Semantic interpretability measures the proportion of assertions traceable to evidence—the higher the proportion, the more transparent the system's decision-making. The OPI index holistically evaluates generation quality, logical correctness, and input consistency by harmonically averaging logical relation accuracy and semantic similarity, thereby suppressing inflated scores in single dimensions. A tiered evaluation strategy is recommended: the foundational tier centers on FActScore and hallucination frequency to ensure factual reliability; the enhanced tier introduces faithfulness and OPI to improve interpretability; the specialized tier incorporates domain-specific metrics to form a gradient evaluation system. Evaluation methods should combine automated metrics, human validation, and user studies to enhance robustness.
- 3. End-to-End System Efficacy: A Retrieval-Generation Synergy Perspective End-to-end evaluation seeks to integrate retrieval and generation stages to holistically measure RAG performance. Methods such as RECALL, FeB4RAG, and Long²RAG construct closed-loop test environments to simultaneously assess retrieval precision and generation quality. For instance, RECALL combines EventKG and UJ datasets to measure retrieval accuracy and generation error recurrence rate, revealing system stability under noise interference; Long²RAG, targeting long-form generation tasks, designs joint metrics for key-point recall and long-text generation accuracy, emphasizing information completeness. Such evaluations reveal synergistic effects and bottlenecks across system components: studies find that even with significant improvements in retrieval relevance, generation quality gains remain relatively limited, indicating that knowledge transformation efficiency has become a key constraint on overall performance. However, most end-to-end evaluations still rely on synthetic news-like data, lacking the complex noise and interaction patterns of real-world scenarios, potentially leading to significant deployment-performance gaps. In contrast, comprehensive evaluation frameworks such as CRAG and RAGBench support multi-domain, multi-question-type task settings and provide

interpretability analysis interfaces [17,18], offering more representative evaluation environments for complex scenarios and addressing limitations of single-task benchmarks.

- 4. Robustness and Timeliness: Facing Real-World Perturbations and Knowledge Updates Robustness and timeliness evaluations aim to test whether RAG systems can maintain expected performance under real-world conditions such as knowledge updates, input perturbations, and evidence conflicts. Recently, benchmarks such as RARE, RGB, and QE-RAG have systematically injected perturbations and designed tasks to transform "stability" into measurable properties, providing clear performance boundaries for deployment-grade systems.
- (1) RARE introduces controlled perturbations at query, document, and retrieval levels to quantify system response consistency under semantic drift, information loss, and ranking jitter; RGB focuses on noise filtering, negative-sample rejection, information integration, and counterfactual reasoning, stress-testing decision reliability under conflicting or misleading contexts; QE-RAG evaluates the effectiveness of underlying fault-tolerance mechanisms using real-world input noise such as spelling errors and word-order inversions. Collectively, these works reveal RAG's adaptive limits in complex environments and drive evaluation protocols to evolve from static accuracy to dynamic robustness.
- (2) Current robustness metrics span the entire query-document-retrieval pipeline:
- a) Query robustness constructs perturbed query sets via paraphrasing or inserting irrelevant phrases, computing the average proportion of correct answers retained post-perturbation to measure semantic generalization and intent preservation;
- b) Document robustness injects irrelevant or conflicting sentences into input passages to test whether models can suppress noise and maintain answer consistency, directly reflecting the efficacy of evidence filtering mechanisms;
- c) Real-world retrieval robustness evaluates generation sensitivity to retrieval result fluctuations by switching retrieval or reranking strategies, simulating variations from heterogeneous retrieval pipelines in deployment environments;
- d) Token-level F1 and Exact Match under the QE-RAG framework measure generation quality for queries with spelling errors—the former based on token overlap, the latter requiring exact string-level answer matches—forming a spectrum of error tolerance from loose to strict:
- e) Noise robustness measures the system's ability to ignore irrelevant information via task accuracy on noisy corpora; negative-sample rejection capability quantifies the system's conservative tendency to abstain when lacking supporting evidence, via a weighted combination of rejection rate and accuracy; counterfactual robustness, under context-parameter knowledge conflicts, measures the proportion of answers prioritizing external evidence, evaluating dependence strength on external memory.
- (3) Timeliness evaluation is increasingly scenario-specific. CRAG categorizes facts by update velocity into real-time, fast-changing, slow-changing, and stable classes, assigning differentiated reference answers to avoid evaluating dynamic knowledge with static labels; the HOH dynamic benchmark continuously injects time-sensitive queries to validate RAG's practical gains in alleviating LLM knowledge-update bottlenecks; FreshQA uses freshness hit rate and outdated answer rate as core metrics to quantify the system's ability to balance timeliness and accuracy in multi-hop reasoning, driving architectures such as MQRF-RAG to achieve 14.45% relative improvement. It should be noted that evaluations based on "closed-set answerability" assumptions (e.g., KILT) fundamentally differ from FreshQA's dynamic update setting; experimental reports must explicitly state contextual premises to prevent metric misalignment.
- 5. Benchmarks and Datasets: From Single-Function to Multi-Dimensional Dynamics In recent years, RAG evaluation has evolved in benchmark design and dataset construction from single-function validation toward multidimensional, systematic assessment. The current evaluation ecosystem has gradually formed a classification framework centered on retrieval quality, generation quality, end-to-end performance, and robustness/timeliness, reflecting the evolution of evaluation priorities and revealing deeper differences across methods in objective setting, data selection, and metric design.
- (1) For retrieval quality, methods such as RAGAs, ARES, and MultiHop-RAG employ standard datasets like WikiEval, NQ, and Hotpot, using traditional IR metrics such as MAP, MRR, and Hit@K. GraphRAG constructs datasets with factual associations and complex reasoning paths to test graph-structured retrieval; CoFE-RAG introduces multi-document formats and diverse query types to improve coverage of real-world document heterogeneity. However, such evaluations still largely rely on static corpora, exhibiting limited capacity to model knowledge updates and dynamic scenarios. BEIR and KILT, as general-purpose evaluation backbones for cross-domain zero-shot retrieval and knowledge-intensive tasks, provide broadly comparable benchmarks for RAG retrieval capabilities[7, 41]. Their coverage of diverse tasks and unified Wikipedia snapshots support evaluation of model generalization in unseen domains, serving as foundational platforms for most retrieval module validation.

- (2) For generation quality, methods such as ARES, FeB4RAG, and DomainRAG quantify generation accuracy using F1, ROUGE-L, and Exact-Match, while introducing LLM-as-a-Judge mechanisms to assess semantic consistency and clarity. ReEval leverages datasets like RealTimeQA, containing time-sensitive questions, to test model performance under outdated or missing knowledge. Although automated metrics enhance evaluation efficiency, their alignment with human judgment remains contested: BERGEN's research reveals weak semantic-level correlation in LLM evaluations, exposing current deficiencies in reliability and interpretability. Methods such as ALCE, QAFactEval, FActScore, FACTS, GaRAGe, and RAGTruth aim to enhance interpretability and factual alignment of generated content, providing fine-grained analyses from citation quality to span-level knowledge attribution[22, 25, 26, 98, 117, 127], thereby deepening evaluation from "whether correct" to "why correct," enhancing result credibility and debugging support.
- (3) For end-to-end system efficacy, methods such as RECALL, FeB4RAG, and Long²RAG construct closed-loop test environments to simultaneously evaluate retrieval precision and generation quality; comprehensive frameworks like CRAG and RAGBench support multi-domain, multi-question-type task settings and provide interpretability analysis interfaces[20, 21], offering more representative evaluation environments for complex scenario comparisons.
- (4) For robustness and timeliness, RGB and CRUD-RAG address noise robustness and knowledge manipulation dimensions respectively: the former tests model resistance to misleading information; the latter designs CREATE, UPDATE, DELETE scenarios to evaluate system management of knowledge lifecycles. Benchmarks such as FreshQA, RAMDocs, and RGB further intensify stress testing for knowledge conflicts and update delays[1, 24, 121], revealing model reasoning fragility under knowledge inconsistency. Additionally, with expanding application scenarios, long-context and multimodal evaluation have become focal points: LongBench, RULER, and Long²RAG pose challenges for document-level understanding and long-text generation; MM-Needle, VisDoMBench, and T²-RAGBench construct cross-modal retrieval and generation tasks [23, 65, 66, 140, 141], extending evaluation from pure text to image-text fusion scenarios. Emerging efforts such as Visual-RAG and RAG-Check have begun constructing multimodal datasets but remain exploratory in real-world interaction and dynamic knowledge fusion.

In summary, RAG evaluation datasets and benchmarks are evolving toward specialization, contextualization, and dynamism: evaluation dimensions have expanded from initial accuracy testing to encompass retrieval, generation, consistency, and timeliness; data sources have extended from general QA datasets to domain-specific, multimodal, and real private corpora. General-purpose frameworks such as BEIR, KILT, CRAG, and RAGBench provide foundations for cross-system comparison, while specialized benchmarks such as ALCE, FActScore, FreshQA, and LongBench deepen evaluation capabilities in specific dimensions. However, the current evaluation ecosystem still faces significant challenges: lack of a unified framework across dimensions hinders cross-benchmark comparability; synthetic data, while facilitating controlled experiments, diverges from real-world scenarios; automated evaluation relying on LLM judgments may introduce evaluation bias. Future research must, while preserving evaluation granularity, strengthen cross-dimensional integration and develop more ecologically representative dynamic evaluation paradigms, advancing RAG evaluation from technical validation toward practical value measurement.

 Table 4.6: RAG Evaluation Methods — Classification and Specification (Compact Version)

Evaluation Type	Method Name	Dataset / Source	Release Date	Dimensions & Metrics
Retrieval-Quality- Oriented	RAGAs [19]	WikiEval	2023.09	Contextual Relevance: Extracted Sentences / Total Sentences, Average Cosine Similarity
	MultiHop- RAG[142]	Self-constructed Dataset	2024.01	Multi-hop Retrieval Quality: MAP, MRR, Hit@K
	CoFE-RAG [143] GraphRAG[144]	Self-constructed Dataset VIINA	2025.06 2024.04	Retrieval Quality, Query Localization Ability: Keyword-based Retrieval Score, Question Locating Accuracy Graph-structured Retrieval, Reasoning Comprehensiveness & Diversity: Fact Retrieval Accuracy, Complex Reasoning Score
	EnronQA[145]	Enron Email Dataset	2025.05	Personalized Retrieval Quality: Contextual Relevance Score
	Visual-RAG[146]	Self-constructed Dataset	2025.02	Multimodal Retrieval Quality: Image Retrieval Accuracy
	ByoKG-RAG[147] Hypercube-RAG [148]	Self-constructed Dataset Self-constructed Dataset	2025.07 2025.05	Graph Retrieval Quality: Graph Retrieval Score Retrieval Efficiency: Retrieval Efficiency Score
	eRAG[96] BEIR[41]	NQ, HotpotQA, etc. MS MARCO, TREC-COVID, NQ, etc.	2024.04 2021.04	Task Utility-Oriented Retrieval (Marginal Contribution): Δ EM, Δ F1 Zero-shot Cross-domain Retrieval Quality: nDCG@10, Recall@100, MRR@10
	KILT[7]	11 datasets based on unified Wikipedia snapshot (e.g., FEVER, NQ)	2020.09	Retrieval Performance in Knowledge-Intensive Tasks: Task-specific metrics (e.g., Label Accuracy for FEVER, EM/F1 for NQ)
Generation-Quality- Oriented	ARES [17]	NQ, Hotpot, FEVER, WoW, MultiRC, ReCoRD	2023.11	Answer Faithfulness, Answer Relevance: Confidence Intervals (for generation faithfulness)
	MedRAG[149]	Self-constructed Dataset	2024.02	Generation Accuracy: Accuracy
	CDQA [150]	Self-constructed Dataset	2024.03	Generation Accuracy: F1
	ReEval [151]	RealTimeQA, NQ WikiEval	2024.06 2023.10	Hallucination Detection: F1, Exact-Match, LLM-as-a-Judge, Human Evaluation Answer Relevance, Fact Attribution (Groundedness): LLM-as-a-Judge
	ASTRID [152] BERGEN [153]	OA Datasets	2023.10	Surface & Semantic Consistency: EM, F1, Precision, Recall, BEM, LLMeval
	RAG-Check [154]	Self-constructed Dataset	2025.01	Multimodal Generation Quality: Generation Quality Metric
	Long ² RAG [155]	Self-constructed Dataset	2025.06	Long-form Generation Quality: Long-Form Generation Accuracy
	FActScore[26]	ASQA	2023.05	Atomic Fact-level Faithfulness: Proportion of atomic facts supported by evidence
	QAFactEval[25]	SummaC, ASQA, etc.	2021.12	Generation Consistency via QA Chains: QA-based F1, Precision, Recall
	ALCE[117]	Specialized dataset with diverse questions & retrieval corpus	2023.05	Fluency, Correctness, Citation Quality of Citation-Augmented Generation: Citation Precision, Citation Recall, Answer Accuracy
	Sufficient Context[156]	Applied on benchmarks: FreshQA, Musique, HotpotQA, etc.	2024.11	Evidence Adequacy, Selective Answering & Refusal Capability: Sufficient/Insufficient Context Label, Selective Accuracy, Refusal Rate
	GaRAGe[22]	Self-constructed Dataset	2025.06	Fine-grained evaluation of LLM's ability to identify & utilize relevant evidence spans: Relevance-Aware Factuality, Grounding Accuracy
	FACTS Grounding[127]	Self-constructed Dataset	2025.01	Span-level Knowledge Attribution & Fact Alignment: Grounding Score — Information pending confirmation
Robustness & Timeliness-Oriented	RGB [121]	Self-constructed Dataset	2023.12	Noise Robustness, Negative-sample Rejection, Counterfactual Robustness: Accuracy, Rejection Rate, Error Detection Rate, Error Correction Rate
	CRUX [157]	Self-constructed Dataset Synthetic QA Dataset	2025.06 2025.08	Context Integrity, Redundancy Control: LLM-based Evaluation, Human Data Assessment Robustness under Data Diversity & Privacy Preservation: Diversity Score, Privacy Masking Accuracy
	Synthetic Datasets Generation [158]	Synthetic QA Dataset	2023.08	Robustiess under Data Diversity & Filvacy Fieservation. Diversity Score, Filvacy Masking Accuracy
	RAG Without the Lag [159]	Self-constructed Dataset		Interactive Debugging Robustness, Pipeline Stability: Debugging Efficiency, Pipeline Performance Metric
	RARE [160] QE-RAG [161]	Self-constructed Dataset Self-constructed Dataset	2025.06 2025.04	Robustness to Query, Document, and Retrieval Perturbations: RARE-Met (Systematic Robustness Metric) Robustness to Input Query Errors: Token-level F1 and EM for error-containing queries
	CRAG[21]	Self-constructed Dataset	2024.06	Comprehensive Evaluation: Popular/Long-tail Entities, Static/Time-sensitive Facts, Retrieval Failure Handling: Accuracy, Interpretability Analysis Interface
	FreshQA[1]	Self-constructed Dataset	2023.10	Ability to Retrieve Up-to-date Information: Freshness Hit Rate, Outdated Answer Rate
	HOH[162]	Self-constructed Dataset	2025.06	Negative Impact of Outdated Information on RAG Performance: Accuracy Drop Magnitude, Dependency on Outdated Information
	CRUD-RAG[163]	Self-constructed Dataset	2024.01	Capability in Knowledge Lifecycle Management (Create/Read/Update/Delete): Task Accuracy under each CRUD operation scenario
E-14- E-10 (RAMDocs[164]	Self-constructed Dataset	2025.04	Stress Testing under Knowledge Conflicts, Noise, and Misinformation: Accuracy
End-to-End System Performance- Oriented	FeB4RAG [165]	BEIR, Generated (Source: News), Labeller	2024.02	Consistency, Correctness, Clarity, Coverage: Human Evaluation, F1, Exact-Match, ROUGE-L, LLM-as-a-Judge
	DomainRAG [166]	College Admission Information	2024.06	Structured Output Capability (System-level Performance): LLM-as-a-Judge
	RECALL [167]	EventKG, UJ	2023.11	Response Quality, System Stability: Accuracy (QA), BLEU, ROUGE-L, Misleading Rate, Mistake Reappearance Rate
	RAGBench[20]	Self-constructed Dataset	2024.06	Interpretable End-to-End Evaluation: TRACe Framework: Utilization, Relevance, Adherence, Completeness

4.4.2 Recommended Reproducible Evaluation Framework

The effectiveness of external knowledge memory can be evaluated across four dimensions: retrieval quality, generation quality, end-to-end system performance, and dynamic adaptability (robustness & timeliness).

As the knowledge entry point of RAG systems, retrieval quality exerts a decisive influence on overall system performance. This study identifies MRR, Hit@K, and Contextual Relevance Score as the core metrics for this dimension.

On the BEIR and KILT datasets, we recommend primarily using MRR to evaluate cross-domain zero-shot retrieval capability. These datasets provide a unified testbed covering 18 diverse tasks, effectively validating model generalization performance in unseen domains. For multi-hop reasoning scenarios, Hit@K@5 and Contextual Relevance Score should be jointly applied on the MultiHop-RAG dataset: the former evaluates the system's ability to retrieve sufficient evidence, while the latter assesses semantic alignment between retrieved results and complex queries. On GraphRAG-Bench, particular attention should be paid to the correlation between Contextual Relevance Score and Complex Reasoning Score to evaluate the effectiveness of graph-structured knowledge retrieval. For personalized retrieval scenarios, Enron email corpus testing should combine Hit@K@3 with human evaluation to validate the system's precise targeting capability in private knowledge environments. In multimodal retrieval evaluation, the Visual-RAG and MM-Needle datasets should integrate Image Retrieval Accuracy with Contextual Relevance Score to comprehensively measure cross-modal semantic understanding.

The generation quality dimension directly determines the output credibility of RAG systems, with F1, Exact-Match, Faithfulness, and LLM-as-a-Judge forming the core evaluation metrics.

For standard QA tasks, the NQ and FEVER datasets should primarily employ F1 and Exact-Match to evaluate factual QA capability: NQ emphasizes accuracy in open-domain QA, while FEVER specifically tests fact verification ability. For multi-hop reasoning scenarios, F1 and Faithfulness should be measured simultaneously on HotpotQA—the former assessing answer accuracy, the latter verifying whether the model over-relies on single evidence sources. In domain-specific evaluations, the MIRAGE medical dataset should prioritize Faithfulness, combined with expert review to validate the reliability of medical advice; the DomainRAG educational consulting dataset should adopt a combination of F1 and LLM-as-a-Judge to assess both factual accuracy and pedagogical appropriateness. For knowledge attribution capability, fine-grained analysis should be conducted using tools such as ALCE, QAFactEval, FActScore, FACTS, GaRAGe, and RAGTruth, comprehensively evaluating interpretability and factual consistency of generated content—from citation quality to span-level alignment [6][7][8][11][12][15]. For timeliness evaluation, the ReEval RealTimeQA dataset should use the ratio of Temporal F1 to Faithfulness to quantify the impact of knowledge updates on generation quality. In multimodal generation evaluation, the RAG-Check and T²-RAGBench datasets should employ multidimensional LLM-as-a-Judge scoring, with particular emphasis on semantic consistency between generated content and retrieved images [24].

End-to-end system efficacy evaluation focuses on the integrated performance of the full RAG pipeline, with Answer Accuracy, Consistency, and Coverage serving as key metrics.

In comprehensive evaluation scenarios, the CRAG and RAGBench datasets should jointly employ Answer Accuracy and Coverage: the former measures final answer correctness, while the latter evaluates information completeness; their combination reveals the system's balancing capability across diverse domain tasks [17][18]. For knowledge manipulation capability testing, CREATE/READ/UPDATE/DELETE tasks should be designed on the CRUD-RAG and UHGE-mal datasets, measuring Answer Accuracy for each operation type—with particular attention to system response latency following knowledge updates in UPDATE tasks. In domain-specific validation, the DomainRAG university admissions consulting dataset should combine Answer Accuracy with Consistency analysis: the former evaluates answer correctness, while the latter tests output stability across similar consultation queries—critical for educational advisory services. For long-text generation tasks, Coverage and Key Point Recall should be prioritized on the LongBench and RULER datasets to evaluate the system's capacity for long-context comprehension and information integration [20][21]. Additionally, the BEIR extended set should combine Answer Accuracy with MRR to analyze the correlation between retrieval quality and final answer accuracy, providing clear guidance for system optimization.

Dynamic adaptability evaluates system stability and adaptability in real-world environments, with Robustness to Noise, Temporal Freshness, and Privacy Preservation as core metrics.

For noise robustness testing, systematically inject varying proportions of misleading information into the RGB synthetic dataset and UHGE-mal, measuring Robustness to Noise (defined as the ratio of misleading information proportion to decline in answer accuracy), with particular focus on low-noise regimes (5%–10%), which better approximate real-world scenarios. For timeliness evaluation, FreshQA and RealTimeQA datasets should conduct pre- and post-knowledge-update comparative tests, quantifying system responsiveness to knowledge changes via the Temporal Freshness metric (defined as the recovery speed of answer accuracy after knowledge updates). The RAMDocs dataset,

designed specifically for knowledge conflict scenarios, should be evaluated using Answer Accuracy and Consistency. For privacy preservation evaluation, Diverse And Private Synthetic Datasets should include known sensitive information fragments, assessing data protection capability via the Privacy Preservation metric (defined as the proportion of sensitive information not leaked). For long-context adaptability, the CRUX dataset should measure the ratio of Context Integrity Score to Redundancy Rate to evaluate the system's ability to preserve critical content while processing redundant information. Finally, in interactive scenario evaluation, the RAG Without the Lag dataset should employ the Debugging Efficiency metric (defined as time/steps from user feedback to answer correction) to measure system adaptability in real user interactions. For comprehensive evaluation of multimodal RAG systems, VisDoMBench and T²-RAGBench provide systematic cross-modal testing environments and should be evaluated by integrating multimodal retrieval and generation metrics.

4.4.3 Challenges and Future Directions

Although the overall research trajectory of RAG evaluation frameworks is becoming increasingly clear, several critical issues still require in-depth investigation.

Current evaluations often conflate accuracy with faithfulness, leading to misjudgments of system capabilities. Research shows that in high-stakes domains such as medical QA, models may generate answers that appear correct but lack evidential support—highlighting the necessity of decoupling these two dimensions for independent assessment. Methods such as ALCE and FActScore have begun advancing this direction by decomposing outputs into fine-grained factual units to enable sentence-level verification of generated content [6][8]. However, discrepancies in the definition and computation of groundedness across different benchmarks undermine the comparability of evaluation results. Frameworks such as QAFactEval and RAGTruth propose evidence-chain-based evaluation paradigms, aiming to establish more consistent standards [11][12]. Future work should further promote the standardization of span-level support evaluation—particularly in high-risk applications such as healthcare and law—by mandating the joint reporting of accuracy and faithfulness metrics to ensure comprehensiveness and reliability of evaluation outcomes.

Most existing evaluations rely on static knowledge bases and closed-set questions, failing to reflect the dynamic evolution of knowledge in real-world scenarios. Benchmarks such as FreshQA and RAMDocs reveal system vulnerabilities when confronted with knowledge updates and information conflicts [13][14]. Experiments indicate that model answer accuracy may significantly degrade when knowledge base content changes—degradations that static benchmarks cannot capture. Stress-testing suites like RGB, by injecting noise and misleading information, further expose insufficient robustness in complex environments [19]. Collectively, these studies demonstrate that evaluations based solely on static datasets tend to overestimate real-world system performance. Future evaluations must systematically incorporate dynamic scenarios such as knowledge updates, conflict resolution, and timeliness responsiveness, constructing test environments that simulate real knowledge lifecycles to more accurately measure practical system capabilities.

Evaluations using LongBench and RULER show that model performance on complex tasks degrades as context length increases—particularly evident in scenarios requiring deep reasoning [20][21]. This suggests that blindly expanding context windows is not a sustainable strategy for performance improvement. In contrast, high-quality external retrieval combined with effective reranking mechanisms can significantly enhance answer quality while controlling computational costs. Methods such as MultiHop-RAG validate the advantages of precise retrieval in multi-hop reasoning tasks [25]. These findings advocate shifting evaluation focus from merely expanding context length toward optimizing retrieval efficiency and knowledge integration. Future research should place greater emphasis on quantifying the relationship between retrieval quality and system performance, exploring architectural designs that achieve optimal performance under constrained resources.

Although LLM-as-a-Judge improves evaluation efficiency, the stability of its judgments and cross-domain generalization capability remain questionable. Scoring biases may exist across different evaluator models, and performance may be unreliable in specialized domains or on adversarial samples. The PPI method proposed by ARES introduces confidence intervals to provide uncertainty quantification for evaluation scores, facilitating more cautious interpretation of results [10]. This approach helps identify high-risk judgments in evaluations, avoiding misjudgments caused by evaluator bias. Future evaluations should adopt hybrid review strategies—combining the efficiency of automated evaluation with the reliability of human assessment—and perform sample-based calibration on critical instances to enhance overall evaluation credibility.

Furthermore, performance and efficiency are critical considerations for external knowledge retrieval systems in real production environments. Practical deployment of external memory systems requires balancing retrieval quality, response latency, computational cost, and knowledge update frequency. Platforms such as CRAG and RAGBench, which support cross-domain and interpretable evaluation, facilitate analysis of system behavior under different configurations [17][18]. Studies show that parameters such as top-k values and context length significantly impact system performance, often exhibiting diminishing returns. Knowledge base update frequency must also be optimized based on the trade-off

between timeliness gains and computational overhead. In long-context and multimodal scenarios, benchmarks such as LongBench, RULER, MM-Needle, VisDoMBench, and T²-RAGBench provide testbeds for evaluating the cost-effectiveness of different technical approaches [20][21][22][23][24]. Future evaluations should explicitly incorporate engineering constraints, establishing comprehensive frameworks that balance performance with sustainability, thereby guiding the design and optimization of RAG systems in real-world business applications.

4.5 Evaluation of Procedural/Episodic Memory

This section focuses on LLM procedural/episodic memory under cross-turn and cross-session conditions: namely, the quality of long-term writes, the traceability and sufficiency of replay, and the inhibit/abstention mechanisms when evidence is insufficient, knowledge is stale, or sources conflict. We adhere to the three operating regimes of §4.1 to separate model capability from information availability, and organize the measurement dimensions via the E-MARS+framework: Encode/Memorize & temporal anchoring (P1), Replay & attribution (P2), Suppress/Freshness/Conflict (P3), Long-horizon stability (P4), and Cost efficiency (P5). We then specify evaluation objects and research questions in §4.5.1, instantiate operating regimes in §4.5.2, define metrics in §4.5.3 and map them to public benchmarks in §4.5.4, and finally discuss threats to validity and key takeaways in §§4.5.5–4.5.6. Our aim is not to compress outcomes into a single score, but to provide a layered, interpretable, and engineering-usable reporting paradigm. [73, 99, 112]

4.5.1 Problem Definition and Evaluation Objects

Unlike one-off long-context understanding, procedural & episodic memory emphasizes cross-turn/session write-replay-inhibit mechanisms: how the system deposits key information as long-term memory, retrieves it at appropriate times, and abstains when knowledge is outdated or evidence is insufficient. Existing work shows that merely enlarging the global context is insufficient for robustness and auditability over long horizons (LoCoMo, Long-MemEval); eventifying/proceduralizing history and driving inference via structured replay yields significant gains (TReMu). [73, 99, 112] Accordingly, we expand the evaluation objects into four complementary families:

- A. Episodic events and timelines: extraction of who-what-where-when-state, order restoration, and recency/state awareness (Episodic Memories Benchmark, SORT/Book-SORT); [74, 168]
- B. Procedural processes and strategies: multi-step workflows, tool selection, and self-reflection/retry (LoCoMo, LongMemEval, Reflexion, Voyager); [73, 112, 169, 170]
- C. Identity/preferences and constraints: identity and preference retention, compliance/safety thresholds, and refusal strategies (MemGPT, Generative Agents, HippoRAG, AriGraph); [31, 97, 102, 171]
- D. Conflict resolution and inhibition: selection under old/new or cross-source conflicts, detection and inhibition of outdated evidence (RAMDocs, FreshLLMs/FreshQA) . [1, 2, 24]

4.5.2 Operating Regimes

To separate capability from information availability, we strictly follow the three regimes of §4.1 and instantiate them for episodic memory scenarios:

Parametric-Only (PO): disable retrieval and replay to form a closed-book baseline of "bare parametric memory," observing the lower bound of recall and long-horizon degradation without external injection.

Offline-Retrieval: freeze the index or session memory store; in this section we split the pipeline into two:

Long-Context (LC-off): direct reading of raw history, diagnosing length/position sensitivity and the upper bound of direct reading (RULER, LongBench, LV-Eval); [65, 66, 71]

Structured-Replay (SR-off): structure history into events/processes/preferences and replay on demand, facilitating the decoupling of write-replay-attribution (protocolized in TReMu). [99]

Online-Retrieval: connect to dynamic knowledge sources or updatable memory stores, focusing on freshness, outdated answer rate, and consistency under conflict (FreshQA, LongMemEval online scenarios). [1, 73]

Unless otherwise stated, we report PO, LC-off, and SR-off in parallel for key slices; when timeliness/conflict is involved, we add OR/SR-on results, preserving comparability with §4.1.

4.5.3 Metric System: The E-MARS+ Five-Panel

To balance measurability and interpretability in procedural/episodic settings, we design E-MARS+ (Encode/Memorize—Attribute/Replay—Suppress/Freshness/Conflict—Stability—Cost) to provide layered measurement across the

four evaluation families(Table 4.7:E-MARS+ Five-Panel Overview (for Procedural/Episodic Memory Evaluation)). Unless otherwise noted, metrics are macro-averaged over samples with 95% confidence intervals; cross-regime/model comparisons use paired permutation or bootstrap tests with Holm/FDR multiple-comparison correction. Adjudicators may be human, NLI consistency models, or LLM-as-a-judge . [28, 118]

Table 4.7: E-MARS+ Five-Panel Overview (for Procedural/Episodic Memory Evaluation)

Panel	Concept & Goal	Main Metrics (Symbols)	Computation Points (1-line definition)	Aggregation / Statistics	Typical Clips / Benchmarks (Examples)
P1 Write & Temporal Anchoring		TAE (Temporal Anchor-	Event / procedure extraction F1; absolute error of temporal anchoring (window-normalized)	Macro average; 95% CI	Episodic Memories, SORT / Book-SORT; Lo- CoMo, LongMemEval; TReMu
P2 Replay & Attribution	Whether replay items are sufficient and traceably support the answer	Fraction); ASR@k; UCR (Unsupported	Proportion of answers traceably supported by replayed evidence; Top-k replay coverage; not con- tradicted by replay items; step-order accuracy	level precision/recall;	TReMu (temporal replay), Episodic, SORT; LoCoMo / Long- MemEval (process/tools)
P3 Inhibition / Freshness / Conflict	Consistency when choosing answers under unanswerable, outdated, or conflicting evidence	Freshness-Hit / Out-		, ,	LongMemEval (unanswerable consistency), FreshQA (freshness), RAMDocs (conflict)
P4 Long-Horizon Robustness	Degradation and inter- ference resistance across turns / sessions	h _{1/2} (half-life); Spurious-	Accuracy decay slope over turns; half-life with- out retraining; error rate of spurious replay; aver- age accuracy under per- turbations		LoCoMo, Long- MemEval; dynamic conversation / task evalu- ations; Beyond-Prompts
P5 Cost & Efficiency	Latency / resource / throughput trade-offs un- der equal performance	kens/turn (re-	End-to-end latency; de- coding + retrieval effi- ciency; per-quality unit cost	mance vs. cost; sensitiv-	LC-off / SR-off / OR/SR-on comparisons; StreamingLLM; KV- cache compression; memory-sensitive bench- marks

4.5.4 Benchmark and Task Mapping

For reproducibility and apples-to-apples comparison, we align E-MARS+ panels (P1–P5) with representative datasets and specify operating regimes and primary adjudication modes. Table 4.8:E-MARS+ Panels: Overview of Dataset–Task–Metric–Operating-Regime Alignment summarizes the evaluation objects, representative benchmarks, primary metrics, default regimes, and notes. [1, 24, 30, 31, 63–66, 71, 73, 74, 79, 97, 102, 112, 168–171]

To mitigate threats to internal/external validity, we follow these controls:

Adjudicator bias and uncertainty (internal validity). When using NLI or LLM-as-a-judge for support decisions, employ dual adjudication with light human spot checks; report basic agreement (e.g., Cohen's κ) and 95% CIs, avoiding single adjudicator conclusions . [28, 118]

Length/position bias and truncation policy (internal validity). Under LC-off, state the context window, positional encoding/truncation rules; where needed, use Found-in-the-Middle calibration as a control. [63, 64]

Replay-entry construction error (internal validity). Under SR-off/-on, extraction/aggregation errors during the write phase can be systematically amplified, manifesting as RSF decreases and UCR increases.

Temporal governance and potential leakage (external validity). In online/dynamic scenarios, disclose the time windows and snapshots of training/index/test to avoid mistaking freshness differences for capability differences; give qualitative accounts of near-duplicates/contamination for public benchmarks. [59–61, 122]

Consistency of comparison protocol (external validity). For cross-regime comparisons, fix retriever/reranker types and key hyperparameters; for cross-model comparisons, use identical random seeds/decoding strategies and budget caps. [17, 21]

Table 4.8: E-MARS+ Panels: Overview of Dataset-Task-Metric-Operating-Regime Alignment

Evaluation Target	Representative Datasets / Tasks	Key Metrics (mapped to E - MARS+)	Run Settings (match §4.1)	Notes
Narrative Events / Timelines	Episodic Memories Benchmark; SORT / Book - SORT	Event -F1, TAE (P1); RSF, Step - Order Acc (P2)	LC-off (length/position diagnosis); SR-off (structured replay off)	Support span - level NLI or LLM -judge for segment verifica- tion.
Procedural Processes / Strategies	LoCoMo; Long- MemEval process clips; Reflexion; Voyager	Procedure -F1 (P1); Tool - Attribution, RSF / ASR@k (P2); Slope@T, $h_{1/2}$ (P4)	PO baseline + LC-off / SR-off; enable OR/SR-on when needed	Also observe long -horizon decay (P4) and synchronous/tool attribution (P2).
Preferences and Policy Constraints	LongMemEval— consistency/ refusal; MemGPT; Generative Agents; Hip- poRAG; AriGraph	Preference - Retention, Policy - Violation Rate (P1/P2); Abstention@Unanswerable AURC (P3)	SR-off; turn on OR/SR-on when freshness is required	Use for person- al/preference safeguarding and policy -compliant selective answer- ing.
Conflict Resolution and Inhibition	RAMDocs (conflict/contradiction/ error prove- nance); FreshQA/ FreshLLMs	CCR, Freshness -Hit, Out -of -Date - Use, Abstention@Unanswerable AURC (P3)	OR or SR-on (dynamic sources), or SR-off as baseline	Record snap- shot dates/update times; prioritize conflict slices in reporting.
Length & Position Control (cross targets)	RULER; Long- Bench; LV -Eval; mechanism di- agnostics: Lost/- Found -in -the - Middle	Length/position performance curves; P4 (de- cay/robustness); P5 (latency/cost)	Primarily <i>LC-off</i> , compared against <i>SR-off</i>	First report performance—length/position sensitivity curves, then show model deltas.

4.5.5 Summary

Within the three operating regimes of §4.1, we construct a multi-object evaluation paradigm covering events—processes/strategies—preferences/constraints—conflict/inhibition, and unify writing, replay, inhibition, long-horizon stability, and engineering cost into an auditable metric set via the E-MARS+ five panels. Mapping to benchmarks such as LoCoMo, LongMemEval, Episodic, SORT/Book-SORT, TReMu, RAMDocs, and FreshQA shows that, in long-term interaction, structured replay is more portable and cost-effective than ultra-long-context direct reading; in dynamic scenarios, freshness and consistency under conflict should be measured explicitly, and selective answering should be used to achieve a controllable trade-off between risk and coverage. This framework supports literature verification and offers actionable diagnostics and optimization cues for engineering deployment. [1, 2, 24, 73, 74, 99, 112, 168]

5 Strategies for Forgetting and Knowledge Updating in LLM Memory

5.1 Problem Definition and Objectives

This chapter focuses on the dynamic management of memory in large language models (LLMs): performing knowledge updating and memory forgetting over different substrates (parametric, contextual, external, procedural/episodic) across the full life cycle, and achieving system-level governance under auditable and reversible constraints. We first articulate the limitations of static memory paradigms and the necessity of dynamic management, then define the objects of study and scope of operations, and finally present an objectives—metrics system aligned with evaluation—deployment.

5.1.1 Limitations of Static Memory and the Need for Dynamic Management

Parametric static memory centered on one-shot pretraining exhibits four systematic deficiencies in real-world scenarios: (1) Knowledge staleness: sluggish response to facts that change over time (positions, regulations, data, etc.); under

the PO setting, freshness is visibly inferior to systems with external evidence [1–3, 7, 11]. (2) Insufficient factuality and traceability: hallucinations are more likely without evidence alignment, making third-party audits difficult [25, 27, 111, 117, 127, 138, 172]. (3) Privacy and compliance risks: training samples may be memorized unintentionally and reproduced under specific prompts; highly repetitive/noisy data amplify membership inference and data leakage risks [57–61, 122]. (4) Working-memory bottlenecks: enlarging the context does not necessarily stabilize evidence utilization, especially for mid-span evidence and ultra-long sequences [63–68].

Therefore, static pretraining alone cannot simultaneously satisfy timeliness, compliance, and verifiability; a dynamic memory management framework (updating, forgetting, and governance) geared to practical applications is urgently needed.

5.1.2 Scope and Objects

We partition LLM memory substrates into: (i) parametric memory (long-term knowledge in weights); (ii) contextual memory (activations and KV caches at inference); (iii) external memory (explicit evidence injected via retrieval or toolchains); and (iv) procedural/episodic memory (events and trajectories across turns). Around these substrates, we discuss two core operations: knowledge updating (introducing or correcting knowledge while preserving existing capabilities) and memory forgetting (suppressing, removing, or externalizing specific knowledge for compliance or safety). Evaluation and reporting follow Chapter 4: use a unified timeline (aligned timestamps for training corpora, indices, and test sets) and report in parallel under closed-book (PO) / offline RAG / online retrieval regimes [7, 8, 11, 16]

Contributions of this chapter: C1 Unified perspective: Propose DMM-Gov (Dynamic Memory Management & Governance), discussing update—forget—audit uniformly across the four memory substrates (parametric/context/external/procedural), with timeline-aligned evaluation and reporting under PO/offline-RAG/online-retrieval regimes [7, 41]. C2 Executable thresholding: Turn updating and forgetting into pre-registrable launch thresholds (ESR/Locality/Drawdown/Generalization; Citation Coverage/Unsupported Claim; four-dimensional forgetting acceptance), with default thresholds and rollback strategies [25–27, 127]. C3 Sequential/lifelong robustness: Propose composite editing routes—"small steps—spacing—sharding" and alignment-based/time-window/event-level (LTE/AToKE/ELKEN + AlphaEdit/WISE/SERAC)—and compose LEME/ConflictEdit/EVOKE as mandatory regressions for long-form consistency, conflict reversibility, and out-of-template generalization [44, 46, 47, 105, 106, 109, 110, 129, 173]. C4 Audit certificate: Standardize "edit/forgetting certificates," requiring versioning—evidence—time window—revocability for third-party verifiable audits (aligned with NIST/OWASP) [4, 5].

5.1.3 Objectives and Challenges: From Principles to Verifiable Metrics

Dynamic management must strike an auditable balance among three objective sets, validated by clear, repeatable metrics.

- (A) Correctness and timeliness: The challenge lies in rapid knowledge changes and the coexistence of errors and newly updated facts. Metrics include Editing Success Rate (ESR); freshness hit rate / outdated-answer rate (TimeQA, FreshLLMs); share of evidence-backed answers and citation coverage (FACTS, ALCE, FActScore, QAFactEval) [1–3, 25–27, 127].
- (B) Controllability and auditability: Point edits can impinge on unrelated capabilities and lack an evidence chain. Metrics include Locality (neighborhood stability), Drawdown (general capability regression), Generalization (synonymic/compositional/multi-hop), and versioning & operation logs (the "edit/forgetting certificate") [42–44, 104].
- (C) Privacy and safety: sample exposure, dangerous capabilities/biases, adversarial resurgence. Metrics include verbatim regurgitation and extraction attack success rates, membership inference AUC (supporting signal only) [57, 58]; dangerous-capability compliance rate (WMDP) [52]; robustness to adversarial resurgence (RWKU, MUSE adversarial settings) [48, 51].

For edit controllability, beyond standard Locality/Drawdown/Generalization, explicitly include conflict consistency and transfer generalization: long-form narrative and cross-paragraph consistency can be validated with LEME; conflict and reversibility pitfalls can be stress-tested with ConflictEdit/Pitfalls; EVOKE diagnoses "overfitting to edit templates" under out-of-template/out-of-domain settings [105, 106, 129]. Temporal consistency and valid time windows should be audited via the time-aware AToKE scheme and metrics [46].

Below, we discuss knowledge updating (§5.2, by parametric/context/external/procedural substrates), memory forgetting (§5.3, optimization/representation/system-level strategies with unified acceptance), and selection & governance from research to release (§5.4).

5.2 Knowledge Updating

This section presents a unified framework for LLM memory updating, aiming to introduce/correct knowledge and control side effects without sacrificing general capability or auditability. We detail four routes by substrate: parametric memory (DAPT/TAPT, PEFT, model editing) for precise closed-book rewrites; contextual memory focusing on read/write at inference and positional calibration; external memory (RAG/KG) offering hot updates and evidence alignment; and procedural/episodic memory for cross-session consistency. The selection follows a "rule of thirds": prioritize external memory for high-timeliness/traceability; use model editing for small, unambiguous fact corrections (with robustness constraints in sequential/lifelong settings); and use DAPT/PEFT with versioned releases for broad yet relatively stable scope. Throughout, we evaluate parametric rewrites via ESR/Locality/Drawdown/Generalization, assess freshness and source attribution for external/contextual routes, and report under timeline alignment (train-index-test) with gray rollout—monitoring—rollback.

5.2.1 Parametric Memory: Continued Pretraining, Parameter-Efficient Finetuning, and Model Editing

When updates involve broad and relatively stable knowledge, two-stage self-supervised Domain/Task-Adaptive Pretraining (DAPT/TAPT) can refresh register and factual distributions while preserving base capabilities [33]. When updates span multiple domains in parallel and require rapid rollback, parameter-efficient finetuning (PEFT) (e.g., Adapters, Prefix/Prompt Tuning, IA³) adjusts only small add-on/scaling parameters on a frozen backbone, naturally supporting versioning and revocation [34–36]. When needs converge to a small number of explicit fact fixes, model editing (ROME/MEND/MEMIT/SERAC) performs targeted rewrites at causally related internal sites, enabling fast, addressable weight updates [42–45].

Sequential/lifelong updates tend to induce edit interference and capability regression. AlphaEdit reduces neighborhood disturbance via null-space constraints; WISE isolates edited knowledge through main/side memory routing, supporting shard merging and rollback; SEAL explores an online, self-editing framework. Engineering-wise, we recommend "small steps-spacing-sharding," with regression against sequential-edit stress cases; for high-risk edits, first deploy SERAC externalized overlay for gray validation before merging into model parameters [44, 110, 173].

Beyond structural constraints (null space/routing), alignment-based editing offers a complementary path: LTE marries knowledge editing with instruction/preference alignment, improving ESR and Locality while reducing regression risk. For time-valid facts, AToKE parameterizes edit targets as timestamps/time windows, enabling auditable switching of old/new knowledge compatible with rollback. To address causal inconsistency from single-point edits, ELKEN proposes event-level editing—jointly updating and consistency-checking multiple entities and constraints—to mitigate cascading side effects of "point fixes, global mismatches." On evaluation, LEME long-form regression tests the persistence and consistency of edited knowledge across long contexts and narrative chains; ConflictEdit/Pitfalls stress conflict/reversibility/template transfer; EVOKE reveals template overfitting risks and recommends out-of-template/out-of-domain/long-horizon acceptance [46, 47, 105, 106, 109, 129].

Process specification: construct target / neighborhood / retention three-way slices; deploy DAPT/PEFT as "adapters + old backbone" in gray rollout; for editing, log an "edit ledger—version ID—reverse-edit points," start with low-traffic gray and track ESR/Locality/Drawdown. Upon spillover, rollback or switch to SERAC/RAG overlays.

Acceptance and regression add-ons. Beyond the "target/neighborhood/retention" slices, add three mandatory regressions: (i) Long-form consistency: include LEME long-form regression [105]; (ii) Conflict/reversibility: include ConflictEd-it/Pitfalls stress sets [106]; (iii) Out-of-template/out-of-domain: use EVOKE-style settings to prevent overfitting to public formats such as CounterFact/zsRE [129]. Time-sensitive edits must record effective/expiry times and align testing with AToKE [46]; for composite events, perform event-level consistency per ELKEN [47].

5.2.2 Contextual Memory: Read/Write Strategies and Positional Calibration

Problem and motivation. In long documents and multi-turn reasoning, contextual memory suffers simultaneously from positional bias and compute/memory budgets: mid-span under-response ("lost-in-the-middle") weakens evidence use, and ultra-long sequences inflate VRAM and latency, limiting feasible window sizes [63] . We therefore systematize three routes: dependency extension, streaming/KV management, and positional calibration.

Method taxonomy. (i) Dependency extension: extend effective dependency chains by introducing reusable states and relative positional encoding; representative methods include Transformer-XL and Compressive Transformer; the more recent Infini-attention retains long-range information at controlled overhead, enabling feasible inference with quasi-"infinite" context [77, 78, 80]. (ii) Streaming inference and KV management: maintain long-sequence stability via StreamingLLM-style online refresh/eviction, combined with KV compression and cross-layer clustering/sharing to reduce VRAM/latency (e.g., PoD, EPIC, KVzip) [79, 85–87, 174]. (iii) Positional calibration: to address low utilization

of mid-span evidence, apply reweighting and anchor instrumentation (e.g., Found-in-the-Middle) to markedly improve mid-span usability, in mutual corroboration with lost-in-the-middle diagnostics [63, 64, 81–83].

Implementation recommendations. Follow a light-to-heavy three-step sequence: first perform structured reordering (paragraph/evidence layout, summarize—replay, and anchor prompting) to improve key-information visibility without extra inference cost; second apply positional reweighting/anchor instrumentation to improve mid-span hits; third, expand the window with paired KV management when latency/VRAM budgets allow. To avoid attention dilution and cost blowups from blind window expansion, design context extension to complement retrieval augmentation (RAG) and enlarge windows only when net gains are demonstrated.

Evaluation protocol. Offline, report position–accuracy curves and performance-vs-length slopes, and quantify mid-span hit rate; online, jointly monitor latency/throughput and long-sequence stability. Dependency-extension routes use Transformer-XL/Compressive as baselines [77, 78]; streaming/KV-compression routes characterize trade-offs via VRAM, throughput, and accuracy retention [79, 85, 86, 174]; positional calibration reports mid-span gains and non-regression at head/tail positions [64]. Long-context unified protocols may follow RULER/HELMET/LongBench [65–68]. Overall, contextual-memory optimization should be co-designed with RAG to achieve auditable trade-offs among accuracy—cost—interpretability. Additionally, for edited knowledge in long contexts, we recommend integrating LEME long-form evaluation into the joint reporting of position—performance curves and length slopes [105].

5.2.3 External Memory: Updatable Indices and Evidence Fusion

At inference, external memory injects auditable evidence via retrieve-rerank-fuse, turning "knowledge updates" into re-embedding / index increments / rerank refreshes. Representative paradigms include RAG [11], REALM (jointly training retriever and LM) [12], RETRO (cross-attention to massive external stores) [13], ATLAS (retrieval augmentation in few-shot settings) [38], and kNN-LM (local posterior correction) [95]; evaluation/alignment can follow KILT [7].

Evidence thresholds and metric bundle. In high-risk domains, use evidence thresholds: trigger refusal/degradation on low-confidence retrieval—attribution; report citation coverage and unsupported-claim rate in tandem. — Retrievers: DPR/ColBERT/Contriever [39, 40, 175]; — Diagnostics & quality: RAGAS, RAGChecker, ARES, CRAG, RankRAG, RAMDocs, RGB [17–19, 21, 24]; — Attribution & faithfulness: FACTS, ALCE, FActScore, QAFactEval [25–27, 127]; — Freshness: TimeQA, FreshLLMs, MIRAGE-Bench [1, 2, 128]; — Tasks/heterogeneous benchmarks: KILT, BEIR [7, 41].

Online mapping: Citation Coverage, Unsupported Claim Rate, Retrieval Recall@k / nDCG@k, Conflict-Handled@k (from RAMDocs), Freshness hit rate. If RAGChecker/RankRAG low confidence triggers the gate, then degrade/refuse and rollback the index or lower retrieval weight [18, 24] . Pre-registered thresholds (examples, domain-adjustable): Citation Coverage ≥ 0.85 ; Unsupported Claim Rate ≤ 0.05 ; Recall@5 ≥ 0.85 ; Freshness ≥ 0.80 ; Conflict-Handled@5 ≥ 0.80 for conflicting evidence. Falling short triggers gray rollback and root-causing (RAGChecker) plus retraining/reembedding.

Pipeline layering: Index layer (versioning & de-duplication, unified chunk-embed-rerank config with conflict detection) \rightarrow Fusion layer (evidence thresholds, sentence/passage-level attribution, cross-source consistency) \rightarrow Ops layer (periodic re-embedding, incremental indexing, snapshot rollback, and a "fact-evidence-timestamp" change log).

5.2.4 Procedural/Episodic Memory: Session-Level Write and Replay

Cross-session consistency relies on explicit organization of events and timelines. In practice, use a pipeline of "event capture \rightarrow summarization & entity tables \rightarrow timeline & vectorized archiving \rightarrow trigger-based replay." Generative Agents structure "observe-remember-reflect-plan" for sustained behavioral consistency; MemGPT uses fast/slow memory and virtual context to achieve cross-session management within limited windows [31, 97]. LoCoMo evaluation shows timeline consistency and long-range dependencies remain current weaknesses; thus, in production, enable explicit references for key slots, configure TTL/decay and conflict merging; if mis-replay increases or timelines drift, revert to the previous memory-store version or temporarily suspend auto-replay [73, 112, 176, 177].

For evaluation, use long-term consistency and timeline slices from LoCoMo/LongMemEval/MemAE/MemBench for offline regression; online, track mis-replay rate and temporal misalignment rate [73, 112, 176, 178].

5.3 Memory Forgetting

We unify forgetting qualification as four-dimensional acceptance: thoroughness (verbatim/semantic/adversarial), utility retention, scalability, and sustainability. Evaluation centers on TOFU/MUSE/RWKU, with MIA as a supporting signal only—not standalone evidence of "successful forgetting" [48, 49, 51, 58].

Threat model (brief). Attackers may (i) access the model as a black box and perform prompt engineering, or (ii) approximate the training distribution and induce verbatim reproduction [57, 122]. Defense objective: under the above capabilities, reduce the verbatim reproduction/extraction success for target samples/concepts to on-par with an "unseen" model (TOFU/MUSE) [48, 49], while maintaining retention-set utility $\geq 98-99\%$, and ensuring robustness to resurgence/adversarial return (RWKU) [51].

Where to implement forgetting. At the optimization level, build on preference-alignment paradigms to suppress specified knowledge via negative preference or contrastive objectives, offering controlled trade-offs among forgetting quality—utility retention—training efficiency [50]. At the representation level, remove or suppress specific concepts/capabilities in intermediate representations—e.g., reduce dangerous capabilities on WMDP while preserving general abilities, or apply closed-form linear concept erasure for verifiable suppression [52–54, 56, 179]. At the system level, externalize facts to forget as retrievable entries and overlay outputs at inference, naturally gaining revocability and auditability [11].

Evaluation. TOFU constructs forget/retain pairs to test "as if never learned" behavior, and reveals resurgence risks due to format rephrasing [49]; MUSE offers a more engineering-realistic composite evaluation across verbatim/knowledge memory, privacy, utility, scale, and order sustainability, indicating the difficulty of achieving thorough forgetting—low side effects—scalability simultaneously [48]; proxy evaluation for dangerous capabilities (WMDP + representation-level forgetting) and robustness to real-world knowledge resurgence (RWKU) show adversarial validation is indispensable [51–54]. Process-wise, first define forget/retain/neighborhood sets and compliance proofs, choose methods and set rollback points; then validate jointly across thoroughness (verbatim/semantic/adversarial)—utility retention—sequential/scale sustainability—recoverability, form metric cards and a failure-case library, and coordinate cascading deletions in data, index, caches, and logs. As a preemptive governance measure, reduce high repetition in training data to lower verbatim-memorization incentives [60, 61]. To avoid misreading failed edits as successful forgetting, cross-validate conflict consistency, long-horizon persistence, and out-of-template generalization via ConflictEdit/Pitfalls, LEME, and EVOKE before/after forgetting [105, 106, 129]; time-related deletions should reuse AToKE time-window settings [46]

Four-dimensional acceptance for memory forgetting: dimensions, metrics, reference benchmarks, and pre-registered thresholds.

5.4 From Research to Release: Selection and Governance

5.4.1 Selection Principles

To reach verifiable trade-offs among accuracy, auditability, and Ops cost, we recommend three principles:

External/procedural memory first. For scenarios with high timeliness, frequent changes, and traceability needs, prefer RAG/KG/procedural memory (non-parametric paths), enabling "update by switching stores" with rollback-friendly release [7, 11].

Parametric edits: prudent and reversible. Apply small-batch, revocable weight edits only to sparse and stable "hard facts." For sequential/lifelong settings, pair with AlphaEdit/WISE/SERAC for robustness and external gray rollout [44, 110, 173].

Closed-loop governance. Throughout the process, enforce versioning \rightarrow canary \rightarrow monitoring \rightarrow rollback \rightarrow audit; prioritize de-dup and cleaning in training and indexing to reduce downstream editing/forgetting burden [4, 60, 61] .

5.4.2 A "Six-Step Decision Flow" for Selection and Deployment

- S1 Identify timeliness and conflict. If knowledge is highly time-sensitive / conflict-prone \rightarrow choose external memory first (RAG/RETRO/kNN-LM or SERAC) [11, 13, 44, 95].
- S2 Determine granularity of change. If it is a small set of clear facts \rightarrow parametric editing (ROME/MEND/MEMIT; pair with AlphaEdit/WISE for sequential cases; gray with SERAC before merging) [42–45, 110, 173].
- S3 Constrain consistency. If temporal evolution is involved \rightarrow adopt time windows and versioned evidence (AToKE); if multi-entity/constraints are involved \rightarrow event-level editing (ELKEN) [46, 47].

S4 Pre-register thresholds. Before updating, set and publish thresholds: ESR/Locality/Drawdown/Generalization (parametric), Citation Coverage/Unsupported Claim/Freshness/Recall@k (external/retrieval) [17, 19, 25, 26, 127].

S5 Canary and online monitoring. Roll out with low traffic; continuously track Citation Coverage, Unsupported Claim Rate, Recall@k / nDCG@k, Freshness, and the trends of ESR/Locality/Drawdown for edits [18, 19, 21].

S6 Rollback and evidence hardening. If monitored metrics exceed thresholds, auto-degrade or rollback (index/edit); version and archive "fact–evidence–time-window" change records for audit [4, 5].

5.4.3 Deployment and Monitoring Standards

Version isolation: manage weights, retrieval indices, and procedural-memory logs under separate versioning; track cross-version dependencies via snapshots and fingerprints.

Canary ramp-up: ramp by business risk and data domain; validate thresholds on low-risk slices before expansion.

Online metrics: for external paths, report Citation Coverage, Unsupported Claim Rate, Recall@k/nDCG@k, Freshness; for parametric paths, report ESR, Locality, Drawdown, Generalization; for procedural memory, monitor mis-replay rate and temporal misalignment rate.

Triggers: when Unsupported Claim Rate or Drawdown exceed thresholds, or conflict-handling rate drops, automatically trigger degrade/refusal/rollback and localization (retrieval diagnosis and edited-neighborhood regression).

5.4.4 Evidence-Backed Update/Forgetting "Certificates": Elements, Ranges, and Calibration

We observe an emerging practice of documenting edits and forgetting events for verification and compliance. Synthesizing prior work [4, 5, 11, 17, 19, 25, 26, 44, 46, 47, 105, 106, 110, 129, 173], we outline *minimal elements* commonly reported and *illustrative threshold ranges* that appear in the literature or industrial case studies. These references are not one-size-fits-all and should be calibrated to task risk, domain, and cost functions, with uncertainty and sensitivity analyses.

Target (F1). Fact/concept, sources, evidence links, snapshots.

Time (F2). Effective/expiry timestamps; time-window settings (AToKE).

Method (F3). Update/forgetting path (RAG/edit/PEFT/system override) and constraints (e.g., AlphaEdit/WISE/ELKEN).

Verification (F4). Examples reported include ESR around 0.90, Locality ≈ 0.95 , Drawdown $\leq 1-2\%$, Citation Coverage $\geq 0.80-0.90$, Unsupported Claim $\leq 0.05-0.10$, Recall@5 $\geq 0.80-0.90$, Freshness $\geq 0.75-0.85$; forgetting is evaluated along *thoroughness-utility-scalability-sustainability*.

Regression (F5). Long-form consistency (LEME), conflict/reversibility (ConflictEdit), and out-of-template/domain generalization (EVOKE).

Rollback (F6). Rollback points/snapshots and impact-surface analysis; fallback via SERAC/RAG.

Audit (F7). Version IDs, operation logs, ownership/approval chain; alignment with NIST AI RMF / OWASP LLM Top-10.

Calibration & Reporting Protocol

- Controlled baselines. Report PO/Offline/Online results on the *same* data slice and time window, and include effect sizes $(\Delta_{abs}/\Delta_{rel})$ relative to the chosen baseline.
- **Uncertainty.** Provide 95% confidence intervals (bootstrap) and *paired* permutation or bootstrap tests with Holm–Bonferroni or FDR correction for multi-metric comparisons; state sample sizes.
- **Risk-tiered targets.** Map metrics to risk tiers (low/medium/high impact) derived from domain cost functions; use the ranges above as illustrative defaults, then pre-register study-specific targets or justify deviations.
- Sensitivity/robustness. Report sensitivity to decoding/retrieval knobs (e.g., temperature, top-p, k, fusion strategy) and to snapshot timing; include at least one *time-slice robustness* check to rule out freshness confounds.
- Evidence mapping. For each metric, cite representative sources supporting the chosen range; an evidence table collating distributions and typical values appears (with pointers to datasets, domains, and evaluation conditions).

Positioning in this survey. Within this survey, F1–F7 are presented as reporting recommendations, together with observed ranges. Concrete deployment thresholds are calibrated on a per-study basis and reported with uncertainty estimates, sensitivity analyses, and baseline-controlled effect sizes.

Limitations. Reported ranges can shift with dataset curation, leakage controls, and evaluator design; we therefore emphasize comparability within a shared slice and encourage release of machine-readable "certificate" artifacts to support independent re-analysis.

6 Challenges and Future Directions

Under a unified definition and a three-regime evaluation protocol (PO / Offline-Retrieval / Online-Retrieval), this paper juxtaposes the in-parameter and extra-parameter memory channels and presents an integrated framework spanning mechanisms—evaluation—updating/forgetting. Despite rapid progress, several core problems and methodological gaps around "memory" still directly constrain comparability, auditability, and large-scale deployment. Below we survey key challenges across four dimensions—theoretical representation, evaluation methodology, governability of updating/forgetting, and system deployment—and propose testable propositions for follow-up research.

6.1 Theory and Representation: Causal Localization and Representation Entanglement Remain Unresolved

- (1) **Insufficient verifiability of causal localization.** Existing evidence supports encoding and read out of knowledge in mid-layer MLPs in a key-value form [8, 9, 42], but such localization mostly relies on intervention heuristics and correlational observations. How to establish, without strong structural priors, a falsifiable causal chain (e.g., quantifying the mediation effect of "edit \rightarrow mediator \rightarrow output") still lacks a unified paradigm.
- (2) Entanglement between knowledge and ability. In-context learning can be viewed as circuit-level copy-and-align behavior [91], or characterized as implicit Bayesian updating or approximate gradient descent under fixed weights [92, 93]. These two views have not yet been unified in terms of time scales and generalization boundaries, yielding empirical phenomena where ability loading and knowledge access are hard to distinguish.
- (3) **Scaling laws of memorization and controllability.** Prior work suggests that verbatim memorization and privacy risk are tightly related to data entropy, repetition, and model scale [59, 122], while de-duplication mitigates risk and may even improve quality [60, 61]. Yet there is no unified characterization of the triad (parameter scale—editability—leakage rate); reproducible studies across distributions and scales with uncertainty quantification are needed.

Open Proposition A. Without strong structural priors, do minimal identifiability conditions exist for "knowledge loci," and can they be reproduced across model families?

6.2 Evaluation Methodology: Harmonized Protocols and Temporal Governance Are Still Incomplete

- (1) Inconsistent notions of passage-level groundedness. It is now agreed in RAG that "correctness \neq faithfulness," but passage-level source attribution and citation coherence remain incompatible across benchmarks [17–19, 26, 27, 127]. The lack of unified annotation granularity, scoring rules, and confidence estimation hinders apples-to-apples comparison.
- (2) For long context, "visible \neq usable." Lost in the Middle and Found in the Middle reveal systematic positional bias [63, 64], yet many evaluations substitute window visibility for usability. Standard reporting of position-performance curves and mid-span drop is needed to compare marginal effects of window expansion, reordering, and reweighting.
- (3) **Reviewer drift and statistical validity.** Instability of LLM-as-a-Judge with respect to order and self-preference has been repeatedly documented [28, 29]. Without adjudication protocols that include confidence intervals and multiple-comparison correction, small improvements cannot be distinguished from noise.
- (4) **Missing temporal dimension.** Current benchmarks under-specify freshness and conflict. Although works like TimeQA and FreshLLMs point the way [1–3], a unified standard for timestamp alignment and versioned reporting has yet to form.

Open Proposition B. With the three regimes run in parallel as external conditions, construct a minimally sufficient evaluation card across tasks, lengths, and time—covering correctness, faithfulness, positional robustness, timeliness, and refusal—and provide unified norms for CIs and significance tests. As a companion, this paper recommends extending the edit/forgetting certificate with pre-registered equivalence margins and three-way slices—reporting effectiveness, locality, downstream steady state, and rollbackability—forming auditable artifacts, consistent with recent editing evaluations [105, 106, 129].

6.3 Updating and Forgetting: Effectiveness, Locality, and Scalability Are Hard to Achieve Simultaneously

- (1) Cascade side effects of pointwise/batch editing. ROME/MEND/MEMIT demonstrate the feasibility of addressable rewrites [42–45], yet in serial and large-batch settings they often exhibit neighborhood spillover and downstream regression, depending on model scale, locus selection, and data distribution.
- (2) **Verifiability and recoverability of forgetting.** Preference- or representation-level unlearning reduces extractability on target sets, but the strong criterion of "as if never learned" remains unstable under real-world and adversarial distributions [48–51, 75, 76]. Without falsifiable and recoverable dual standards, audit costs stay high.
- (3) **Ternary trade-off of privacy—utility—cost.** De-duplication and data governance markedly reduce regurgitation and extraction [57, 60, 61], but unified quantification of coverage/perplexity impact across thresholds is still lacking.

Open Proposition C. Do combinations of *causally constrained editing* and *verifiable unlearning* exist that attain a provable Pareto frontier over effectiveness, locality, and scalability?

6.4 Systems and Deployment: Joint Governance of Conflicting Evidence, Cost, and Accountability

- (1) **Retrieval bottlenecks and error amplification.** End-to-end performance is highly sensitive to recall; the retrieve–rerank–fuse error chain is amplified with long documents and cross-source conflicts [15, 17, 18, 24, 41, 96]. Task-driven document utility should be jointly reported with refusal/abstention metrics to avoid "surface-level correctness without evidence."
- (2) **Long-horizon consistency and mis-replay.** In multi-session/agentic scenarios, expanding the window alone cannot maintain timeline consistency; without eventification and timeline structuring, mis-replay and inconsistency increase [31, 73, 97, 113].
- (3) **Operations and compliance.** Versioned indices, snapshot rollback, and audit trails are hard constraints for real systems; academic reports rarely disclose the functional relationship among cost–latency–freshness or rollback strategies. In line with NIST AI-RMF and OWASP LLM Top-10 [4, 5], answer-only-with-evidence hard gates and cascading deletion/audit should be built into evaluation/release protocols.

Open Proposition D. Under fixed latency/cost budgets, can *high-quality retrieval* + *reranking* + *small-window replay* systematically outperform *ultra-long-context direct reading?* Boundary conditions and cross-domain transferability call for a unified protocol (cf. long-context evaluations [65–67]).

6.5 Future Research Directions

- (i) **Unified memory semantics and observable criteria.** Use the "four-way taxonomy + quadruple" (storage location—persistence—write/access path—controllability) as a semantic anchor; add formal observable equivalence classes and cross-regime invariance analyses.
- (ii) Causally consistent editing and verifiable unlearning. Combine circuit-level mediation analysis with constrained optimization; construct forgetting certificates with counterfactual tests and inverse edits [48, 49, 105, 106].
- (iii) **Time-aware evaluation and governance.** Under unified snapshot and dynamic-update scenarios, adopt timestampaligned protocols and metrics such as Fresh@k / Outdated% / Refusal@Stale [1–3], reported jointly with citation coverage.
- (iv) **Read/write orchestration for long context.** Develop learnable strategies combining positional reweighting, anchor-based reordering, and summary–replay; treat position–performance curves as a first-class metric [63–65, 67].
- (v) **Explainable fusion of external memory.** Under cross-source conflict and noise, introduce consistency adjudication—refusal—uncertainty estimation for joint optimization of correctness—faithfulness—auditability [17, 18, 20, 127].
- (vi) **Transferable quantification of privacy risk.** Using data entropy, repetition, and scaling laws as covariates, build predictive models across corpora and models, offering interval estimates and conservative bounds for regurgitation/extraction/membership inference [59–61, 122].

Overall, memory is neither a single substrate nor a single capability, but a multilayer collection of states and mechanisms co-determined by training objectives, inference dynamics, and system architecture. Coordinated advances in causally consistent mechanistic evidence, time-aware evaluation protocols, verifiable editing/unlearning, and auditable system governance will help capability—timeliness—controllability—cost reach higher levels within a unified experimental and engineering coordinate system.

7 Conclusion

This paper presents a systematic survey and methodological synthesis around memory in large language models (LLMs). Under a unified operationalized definition, we propose a four-way taxonomy (parametric memory, contextual memory, external memory, procedural/episodic memory) and a "memory quadruple" (storage location—persistence—write/access path—controllability), and we connect mechanistic evidence, evaluation protocols, and engineering governance through a causal chain of "write—read—inhibit/update." To mitigate distorted comparisons caused by heterogeneous research setups, we construct three parallel operating regimes—parametric-only (PO), offline retrieval augmentation (Offline-Retrieval), and online retrieval augmentation (Online-Retrieval)—to separate the contributions of information availability and model capability on the same samples and timeline.

In evaluation methodology, we provide layered metric systems and reproducible protocols covering the four memory types: for parametric memory, we emphasize factual recall under the PO setting, pre/post-edit differentials, and memorization/privacy risks; for contextual memory, we foreground the diagnosis that "visible \neq usable," with primary reporting via position–performance curves and mid-span drop; for external memory, we decouple retrieval quality from faithfulness/source attribution and jointly report correctness and passage-level groundedness; for procedural/episodic memory, we adopt componentized evaluation of cross-session consistency, timeline replay, and mis-replay control. All the above metrics are uniformly brought under a temporal dimension (Fresh@k, Outdated%, Refusal@Stale) and uncertainty reporting (confidence intervals, paired tests, and multiple-comparison corrections), complemented by human–AI mixed adjudication to reduce the drift risk of LLM-as-a-Judge.

For updating and forgetting, we place continued pretraining and parameter-efficient finetuning (PEFT), model editing (e.g., ROME/MEND/MEMIT), and external override (RAG/SERAC) on a single decision surface, describe their inherent trade-offs via a three-axis Pareto analysis (target suppression—neighborhood preservation—downstream steady state), and propose a minimal governance closed loop for release: versioning (weights/index/logs managed separately), canary and rollback, online monitoring (citation coverage, unsupported-claim rate, timeliness/conflict slices), together with systematic support for compliant deletion and audit trails. To address the challenge of verifiable forgetting, we propose an "edit/forgetting certificate" reporting framework that uses pre-registered equivalence margins and counterfactual validation to document evidence of effectiveness, locality, and recoverability, thereby turning research findings into auditable artifacts.

Synthesizing evidence across components, our main findings can be summarized in four points: (1) Memory is not a single substrate; it is a multi-layer collection of states shaped jointly by training objectives, network structure, and system architecture. (2) The upper bound of long-context capability is constrained by positional bias and attention dilution; structured reordering and anchor-based replay are often more cost-effective than blind window expansion. (3) Retrieval recall is the dominant bottleneck of RAG; reranking and answer-only-with-evidence hard gates yield synchronous gains in correctness and faithfulness. (4) Editing/unlearning rarely attains effectiveness, locality, and scalability simultaneously; small steps, reversibility, and external coverage first are needed to reduce cascading spillover.

This work has three main limitations: first, although we advocate unified protocols and offer reproducibility guidance, absolute comparability across datasets and implementations remains constrained by public resources and implementation details; second, some evaluations rely on proxy tasks or static snapshots, still short of real dynamic environments; third, formal conditions for causally consistent editing and verifiable unlearning have yet to reach consensus and require further testing across model families and scales. To this end, Chapter 6 proposed four open propositions—identifiability conditions, a minimally sufficient evaluation card, causally constrained editing & verifiable unlearning, and the boundary where retrieval + small-window replay outperforms ultra-long-window direct reading—as testable paths for subsequent work.

Overall, our contributions are: providing a deployable unified terminology and typed framework; bringing correctness, faithfulness, positional robustness, and timeliness into a single evaluation coordinate system via "three regimes in parallel" and layered metrics; situating updating/unlearning within engineering governance, with reversible, auditable implementation guidelines and certificate-style reporting; and distilling a practice checklist of "external-memory-first—small-step editing—timestamp alignment—evidence grounding—audit trails." We hope these methodological baselines and governance points will supply a shared coordinate system for research and industrial deployment alike, advancing LLM memory research toward comparability, deployability, and governability.

References

- [1] Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc Le, and Thang Luong. Freshllms: Refreshing large language models with search engine augmentation, 2023.
- [2] Wenhu Chen, Xinyi Wang, and William Yang Wang. A dataset for answering time-sensitive questions, 2021.
- [3] Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273, 2022.
- [4] Elham Tabassi. Artificial intelligence risk management framework (ai rmf 1.0), 2023-01-26 05:01:00 2023.
- [5] OWASP Foundation. Owasp top 10 for large language model applications. Technical Report, 2023. Version 1.1 released in October 2023; updated in 2024 and 2025.
- [6] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. Language models as knowledge bases?, 2019.
- [7] Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. Kilt: a benchmark for knowledge intensive language tasks, 2021.
- [8] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories, 2021.
- [9] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.
- [10] Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, Roger Grosse, Sam McCandlish, Jared Kaplan, Dario Amodei, Martin Wattenberg, and Christopher Olah. Toy models of superposition, 2022.
- [11] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021.
- [12] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Realm: Retrieval-augmented language model pre-training, 2020.
- [13] Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162 of *Proceedings of Machine Learning Research*, pages 2206–2240. PMLR, 2022. RETRO.
- [14] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey, 2024.
- [15] Hailey Joren, Jianyi Zhang, Chun-Sung Ferng, Da-Cheng Juan, Ankur Taly, and Cyrus Rashtchian. Sufficient context: A new lens on retrieval augmented generation systems, 2025.
- [16] Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7):3580–3599, July 2024.
- [17] Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. Ares: An automated evaluation framework for retrieval-augmented generation systems, 2024.
- [18] Dongyu Ru, Lin Qiu, Xiangkun Hu, Tianhang Zhang, Peng Shi, Shuaichen Chang, Cheng Jiayang, Cunxiang Wang, Shichao Sun, Huanyu Li, Zizhao Zhang, Binjie Wang, Jiarong Jiang, Tong He, Zhiguo Wang, Pengfei Liu, Yue Zhang, and Zheng Zhang. Ragchecker: A fine-grained framework for diagnosing retrieval-augmented generation, 2024.
- [19] Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. Ragas: Automated evaluation of retrieval augmented generation, 2025.

- [20] Robert Friel, Masha Belyi, and Atindriyo Sanyal. Ragbench: Explainable benchmark for retrieval-augmented generation systems, 2025.
- [21] Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Daniel Gui, Ziran Will Jiang, Ziyu Jiang, Lingkun Kong, Brian Moran, Jiaqi Wang, Yifan Ethan Xu, An Yan, Chenyu Yang, Eting Yuan, Hanwen Zha, Nan Tang, Lei Chen, Nicolas Scheffer, Yue Liu, Nirav Shah, Rakesh Wanga, Anuj Kumar, Wen tau Yih, and Xin Luna Dong. Crag comprehensive rag benchmark, 2024.
- [22] Ionut-Teodor Sorodoc, Leonardo F. R. Ribeiro, Rexhina Blloshmi, Christopher Davis, and Adrià de Gispert. Garage: A benchmark with grounding annotations for rag evaluation, 2025.
- [23] Manan Suri, Puneet Mathur, Franck Dernoncourt, Kanika Goswami, Ryan A. Rossi, and Dinesh Manocha. Visdom: Multi-document qa with visually rich elements using multimodal retrieval-augmented generation, 2025.
- [24] Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. Retrieval-augmented generation with conflicting evidence, 2025.
- [25] Alexander R. Fabbri, Chien-Sheng Wu, Wenhao Liu, and Caiming Xiong. Qafacteval: Improved qa-based factual consistency evaluation for summarization, 2022.
- [26] Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation, 2023.
- [27] Xinze Li, Yixin Cao, Liangming Pan, Yubo Ma, and Aixin Sun. Towards verifiable generation: A benchmark for knowledge-aware language model attribution. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 493–516, Bangkok, Thailand, August 2024. Association for Computational Linguistics.
- [28] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.
- [29] Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. Large language models are not fair evaluators, 2023.
- [30] David Castillo-Bolado, Joseph Davidson, Finlay Gray, and Marek Rosa. Beyond prompts: Dynamic conversational benchmarking of large language models, 2024.
- [31] Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. Memgpt: Towards Ilms as operating systems, 2024.
- [32] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2023.
- [33] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don't stop pretraining: Adapt language models to domains and tasks, 2020.
- [34] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp, 2019.
- [35] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning, 2021.
- [36] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation, 2021.
- [37] Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering, 2021.
- [38] Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models, 2022.
- [39] Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen tau Yih. Dense passage retrieval for open-domain question answering, 2020.
- [40] Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert, 2020.
- [41] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models, 2021.
- [42] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt, 2023.

- [43] Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. Fast model editing at scale, 2022.
- [44] Eric Mitchell, Charles Lin, Antoine Bosselut, Christopher D. Manning, and Chelsea Finn. Memory-based model editing at scale, 2022.
- [45] Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer, 2023.
- [46] Xunjian Yin, Jin Jiang, Liming Yang, and Xiaojun Wan. History matters: Temporal knowledge editing in large language model, 2023.
- [47] Hao Peng, Xiaozhi Wang, Chunyang Li, Kaisheng Zeng, Jiangshan Duo, Yixin Cao, Lei Hou, and Juanzi Li. Event-level knowledge editing, 2024.
- [48] Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. Muse: Machine unlearning six-way evaluation for language models, 2024.
- [49] Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. Tofu: A task of fictitious unlearning for llms, 2024.
- [50] Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning, 2024.
- [51] Zhuoran Jin, Pengfei Cao, Chenhao Wang, Zhitao He, Hongbang Yuan, Jiachun Li, Yubo Chen, Kang Liu, and Jun Zhao. Rwku: Benchmarking real-world knowledge unlearning for large language models, 2024.
- [52] Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Adam Khoja, Zhenqi Zhao, Ariel Herbert-Voss, Cort B. Breuer, Samuel Marks, Oam Patel, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam A. Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Russell Kaplan, Ian Steneker, David Campbell, Brad Jokubaitis, Alex Levinson, Jean Wang, William Qian, Kallol Krishna Karmakar, Steven Basart, Stephen Fitz, Mindy Levine, Ponnurangam Kumaraguru, Uday Tupakula, Vijay Varadharajan, Ruoyu Wang, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. The wmdp benchmark: Measuring and reducing malicious use with unlearning, 2024.
- [53] Nora Belrose, Daniel Schneider-Joseph, Shauli Ravfogel, Ryan Cotterell, Edward Raff, and Stella Biderman. Leace: Perfect linear concept erasure in closed form. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.
- [54] Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan Cotterell. Linear adversarial concept erasure, 2024.
- [55] Shauli Ravfogel, Francisco Vargas, Yoav Goldberg, and Ryan Cotterell. Adversarial concept erasure in kernel space. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6034–6055, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.
- [56] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. Null it out: Guarding protected attributes by iterative nullspace projection, 2020.
- [57] Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650. USENIX Association, 2021.
- [58] Michael Duan, Anshuman Suri, Niloofar Mireshghallah, Sewon Min, Weijia Shi, Luke Zettlemoyer, Yulia Tsvetkov, Yejin Choi, David Evans, and Hannaneh Hajishirzi. Do membership inference attacks work on large language models?, 2024.
- [59] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. Quantifying memorization across neural language models, 2023.
- [60] Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models, 2022.
- [61] Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better, 2022.

- [62] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling, 2020.
- [63] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts, 2023.
- [64] Cheng-Yu Hsieh, Yung-Sung Chuang, Chun-Liang Li, Zifeng Wang, Long T. Le, Abhishek Kumar, James Glass, Alexander Ratner, Chen-Yu Lee, Ranjay Krishna, and Tomas Pfister. Found in the middle: Calibrating positional attention bias improves long context utilization, 2024.
- [65] Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. Ruler: What's the real context size of your long-context language models?, 2024.
- [66] Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. Longbench: A bilingual, multitask benchmark for long context understanding, 2024.
- [67] Howard Yen, Tianyu Gao, Minmin Hou, Ke Ding, Daniel Fleischer, Peter Izsak, Moshe Wasserblat, and Danqi Chen. Helmet: How to evaluate long-context language models effectively and thoroughly, 2025.
- [68] Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Khai Hao, Xu Han, Zhen Leng Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. ∞bench: Extending long context evaluation beyond 100k tokens, 2024.
- [69] Uri Shaham, Elad Segal, Maor Ivgi, Avia Efrat, Ori Yoran, Adi Haviv, Ankit Gupta, Wenhan Xiong, Mor Geva, Jonathan Berant, and Omer Levy. Scrolls: Standardized comparison over long language sequences, 2022.
- [70] Zican Dong, Tianyi Tang, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. Bamboo: A comprehensive benchmark for evaluating long text modeling capacities of large language models, 2024.
- [71] Tao Yuan, Xuefei Ning, Dong Zhou, Zhijie Yang, Shiyao Li, Minghui Zhuang, Zheyue Tan, Zhuyu Yao, Dahua Lin, Boxun Li, Guohao Dai, Shengen Yan, and Yu Wang. Lv-eval: A balanced long-context benchmark with 5 length levels up to 256k, 2024.
- [72] Hengyi Wang, Haizhou Shi, Shiwei Tan, Weiyi Qin, Wenyuan Wang, Tunyu Zhang, Akshay Nambi, Tanuja Ganu, and Hao Wang. Multimodal needle in a haystack: Benchmarking long-context capability of multimodal large language models, 2025.
- [73] Di Wu, Hongwei Wang, Wenhao Yu, Yuwei Zhang, Kai-Wei Chang, and Dong Yu. Longmemeval: Benchmarking chat assistants on long-term interactive memory, 2025.
- [74] Alexis Huet, Zied Ben Houidi, and Dario Rossi. Episodic memories generation and evaluation benchmark for large language models, 2025.
- [75] Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Peter Kairouz, Kangwook Lee, Brendan McMahan, Ari Morcos, Gaurav Pandey, Kunal Talwar, Florian Tramèr, and Nicholas Carlini. Machine unlearning. In 2021 IEEE Symposium on Security and Privacy (SP), pages 141–159. IEEE, 2021.
- [76] Jiabao Ji, Yujian Liu, Yang Zhang, Gaowen Liu, Ramana Rao Kompella, Sijia Liu, and Shiyu Chang. Reversing the forget-retain objectives: An efficient llm unlearning framework from logit difference, 2024.
- [77] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context, 2019.
- [78] Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, and Timothy P. Lillicrap. Compressive transformers for long-range sequence modelling, 2019.
- [79] Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks, 2024.
- [80] Tsendsuren Munkhdalai, Manaal Faruqui, and Siddharth Gopal. Leave no context behind: Efficient infinite context transformers with infini-attention, 2024.
- [81] Shouyuan Chen, Sherman Wong, Liangjian Chen, and Yuandong Tian. Extending context window of large language models via positional interpolation, 2023.
- [82] Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023.
- [83] Ofir Press, Noah A. Smith, and Mike Lewis. Train short, test long: Attention with linear biases enables input length extrapolation, 2022.

- [84] Hanlin Tang, Yang Lin, Jing Lin, Qingsen Han, Shikuan Hong, Yiwu Yao, and Gongyi Wang. Razorattention: Efficient ky cache compression through retrieval heads, 2024.
- [85] Junhao Hu, Wenrui Huang, Weidong Wang, Haoyi Wang, Tiancheng Hu, Qin Zhang, Hao Feng, Xusheng Chen, Yizhou Shan, and Tao Xie. Epic: Efficient position-independent caching for serving large language models, 2025.
- [86] Jang-Hyun Kim, Jinuk Kim, Sangwoo Kwon, Jae W. Lee, Sangdoo Yun, and Hyun Oh Song. Kvzip: Queryagnostic kv cache compression with context reconstruction, 2025.
- [87] Jie Hu, Shengnan Wang, Yutong He, Ping Gong, Jiawei Yi, Juncheng Zhang, Youhui Bai, Renhai Chen, Gong Zhang, Cheng Li, and Kun Yuan. Efficient long-context llm inference via kv cache clustering, 2025.
- [88] Heejun Lee, Geon Park, Jaduk Suh, and Sung Ju Hwang. Infinitehip: Extending language model context up to 3 million tokens on a single gpu, 2025.
- [89] Minsoo Kim, Kyuhong Shim, Jungwook Choi, and Simyung Chang. Infinipot: Infinite context processing on memory-constrained llms, 2024.
- [90] Linhao Luo, Thuy-Trang Vu, Dinh Phung, and Gholamreza Haffari. Systematic assessment of factual knowledge in large language models, 2023.
- [91] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads, 2022.
- [92] Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? investigations with linear models. In *International Conference on Learning Representations* (*ICLR*), 2023.
- [93] Johannes von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent, 2023.
- [94] Matanel Oren, Michael Hassid, Nir Yarden, Yossi Adi, and Roy Schwartz. Transformers are multi-state rnns, 2024.
- [95] Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. Generalization through memorization: Nearest neighbor language models, 2020.
- [96] Alireza Salemi and Hamed Zamani. Evaluating retrieval quality in retrieval-augmented generation, 2024.
- [97] Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior, 2023.
- [98] Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Randy Zhong, Juntong Song, and Tong Zhang. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models, 2024.
- [99] Yubin Ge, Salvatore Romeo, Jason Cai, Raphael Shu, Monica Sunkara, Yassine Benajiba, and Yi Zhang. Tremu: Towards neuro-symbolic temporal reasoning for llm-agents with memory in multi-session dialogues, 2025.
- [100] Zidi Xiong, Yuping Lin, Wenya Xie, Pengfei He, Jiliang Tang, Himabindu Lakkaraju, and Zhen Xiang. How memory management impacts llm agents: An empirical study of experience-following behavior, 2025.
- [101] Huichi Zhou, Yihang Chen, Siyuan Guo, Xue Yan, Kin Hei Lee, Zihan Wang, Ka Yiu Lee, Guchun Zhang, Kun Shao, Linyi Yang, and Jun Wang. Memento: Fine-tuning llm agents without fine-tuning llms, 2025.
- [102] Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. Hipporag: Neurobiologically inspired long-term memory for large language models, 2025.
- [103] Hongjin Qian, Zheng Liu, Peitian Zhang, Kelong Mao, Defu Lian, Zhicheng Dou, and Tiejun Huang. Memorag: Boosting long context processing with global memory-enhanced retrieval augmentation, 2025.
- [104] Jia-Chen Gu, Hao-Xiang Xu, Jun-Yu Ma, Pan Lu, Zhen-Hua Ling, Kai-Wei Chang, and Nanyun Peng. Model editing harms general abilities of large language models: Regularization to the rescue, 2024.
- [105] Domenic Rosati, Robie Gonzales, Jinkun Chen, Xuemin Yu, Melis Erkan, Yahya Kayani, Satya Deepika Chavatapalli, Frank Rudzicz, and Hassan Sajjad. Long-form evaluation of model editing, 2024.
- [106] Cheng-Hsun Hsueh, Paul Kuo-Ming Huang, Tzu-Han Lin, Che-Wei Liao, Hung-Chieh Fang, Chao-Wei Huang, and Yun-Nung Chen. Editing the mind of giants: An in-depth exploration of pitfalls of knowledge editing in large language models, 2024.

- [107] Mengqi Zhang, Xiaotian Ye, Qiang Liu, Pengjie Ren, Shu Wu, and Zhumin Chen. Uncovering overfitting in large language model editing, 2025.
- [108] Qi Li, Xiang Liu, Zhenheng Tang, Peijie Dong, Zeyu Li, Xinglin Pan, and Xiaowen Chu. Should we really edit language models? on the evaluation of edited language models, 2024.
- [109] Yuxin Jiang, Yufei Wang, Chuhan Wu, Wanjun Zhong, Xingshan Zeng, Jiahui Gao, Liangyou Li, Xin Jiang, Lifeng Shang, Ruiming Tang, Qun Liu, and Wei Wang. Learning to edit: Aligning llms with knowledge editing, 2024.
- [110] Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. Wise: Rethinking the knowledge memory for lifelong model editing of large language models, 2024.
- [111] Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. True: Re-evaluating factual consistency evaluation, 2022.
- [112] Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. Evaluating very long-term conversational memory of llm agents, 2024.
- [113] Yuanzhe Hu, Yu Wang, and Julian McAuley. Evaluating memory in llm agents via incremental multi-turn interactions, 2025.
- [114] Alex Wang, Kyunghyun Cho, and Mike Lewis. Asking and answering questions to evaluate the factual consistency of summaries, 2020.
- [115] Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, Benjamin Newman, Binhang Yuan, Bobby Yan, Ce Zhang, Christian Cosgrove, Christopher D. Manning, Christopher Ré, Diana Acosta-Navas, Drew A. Hudson, Eric Zelikman, Esin Durmus, Faisal Ladhak, Frieda Rong, Hongyu Ren, Huaxiu Yao, Jue Wang, Keshav Santhanam, Laurel Orr, Lucia Zheng, Mert Yuksekgonul, Mirac Suzgun, Nathan Kim, Neel Guha, Niladri Chatterji, Omar Khattab, Peter Henderson, Qian Huang, Ryan Chi, Sang Michael Xie, Shibani Santurkar, Surya Ganguli, Tatsunori Hashimoto, Thomas Icard, Tianyi Zhang, Vishrav Chaudhary, William Wang, Xuechen Li, Yifan Mai, Yuhui Zhang, and Yuta Koreeda. Holistic evaluation of language models, 2023.
- [116] Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. Dynabench: Rethinking benchmarking in nlp, 2021.
- [117] Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. Enabling large language models to generate text with citations, 2023.
- [118] Chirag Agarwal, Sree Harsha Tanneru, and Himabindu Lakkaraju. Faithfulness vs. plausibility: On the (un)reliability of explanations from large language models, 2024.
- [119] Bhavana Dalvi, Peter Jansen, Oyvind Tafjord, Zhengnan Xie, Hannah Smith, Leighanna Pipatanangkura, and Peter Clark. Explaining answers with entailment trees, 2022.
- [120] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. Ms marco: A human generated machine reading comprehension dataset, 2018.
- [121] Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. Benchmarking large language models in retrieval-augmented generation, 2023.
- [122] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks, 2019.
- [123] Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models, 2023.
- [124] Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness, 2022.
- [125] Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning, 2023.
- [126] Da Ma, Lu Chen, Situo Zhang, Yuxun Miao, Su Zhu, Zhi Chen, Hongshen Xu, Hanqi Li, Shuai Fan, Lei Pan, and Kai Yu. Compressing ky cache for long-context llm inference with inter-layer attention similarity, 2025.

- [127] Alon Jacovi, Andrew Wang, Chris Alberti, Connie Tao, Jon Lipovetz, Kate Olszewska, Lukas Haas, Michelle Liu, Nate Keating, Adam Bloniarz, Carl Saroufim, Corey Fry, Dror Marcus, Doron Kukliansky, Gaurav Singh Tomar, James Swirhun, Jinwei Xing, Lily Wang, Madhu Gurumurthy, Michael Aaron, Moran Ambar, Rachana Fellinger, Rui Wang, Zizhao Zhang, Sasha Goldshtein, and Dipanjan Das. The facts grounding leaderboard: Benchmarking llms' ability to ground responses to long-form input, 2025.
- [128] Nandan Thakur, Suleman Kazi, Ge Luo, Jimmy Lin, and Amin Ahmad. Mirage-bench: Automatic multilingual benchmark arena for retrieval-augmented generation systems, 2025.
- [129] Xinyu Hu, Zijun Wu, Zhiyuan Liu, and Maosong Sun. Evoke: Evoking critical thinking abilities in llms via reviewer-author prompt editing. *arXiv preprint*, October 2023.
- [130] Akshat Gupta and Gopala Krishna Anumanchipalli. Rebuilding rome: Resolving model collapse during sequential model editing. In *Conference on Empirical Methods in Natural Language Processing*, 2024.
- [131] Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 12:283–298, 2023.
- [132] Peter Carragher, Abhinand Jha, R Raghav, and Kathleen M. Carley. Quantifying memorization and parametric response rates in retrieval-augmented vision-language models, 2025.
- [133] Yufei Tao, Adam Hiatt, Erik Haake, Antonie J. Jetter, and Ameeta Agrawal. When context leads but parametric memory follows in large language models, 2024.
- [134] Anil Kumar Shukla. Large language model evaluation in 2025: Smarter metrics that separate hype from trust. Technical report, TechRxiv, jun 2025. Preprint.
- [135] Maor Ivgi, Uri Shaham, and Jonathan Berant. Efficient long-text understanding with short-text models, 2022.
- [136] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.
- [137] Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y. Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. Rarr: Researching and revising what language models say, using language models, 2023.
- [138] Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. Evaluating the factual consistency of abstractive text summarization, 2019.
- [139] Yue Yu, Wei Ping, Zihan Liu, Boxin Wang, Jiaxuan You, Chao Zhang, Mohammad Shoeybi, and Bryan Catanzaro. Rankrag: Unifying context ranking with retrieval-augmented generation in llms, 2024.
- [140] Hengyi Wang, Haizhou Shi, Shiwei Tan, Weiyi Qin, Wenyuan Wang, Tunyu Zhang, Akshay Nambi, Tanuja Ganu, and Hao Wang. Multimodal needle in a haystack: Benchmarking long-context capability of multimodal large language models. *arXiv preprint arXiv:2406.11230*, 2024.
- [141] Jan Strich, Enes Kutay Isgorur, Maximilian Trescher, Chris Biemann, and Martin Semmann. T²-ragbench: Text-and-table benchmark for evaluating retrieval-augmented generation, 2025.
- [142] Yixuan Tang and Yi Yang. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries. *arXiv preprint arXiv:2401.15391*, 2024.
- [143] Jintao Liu, Ruixue Ding, Linhao Zhang, Pengjun Xie, and Fie Huang. Cofe-rag: A comprehensive full-chain evaluation framework for retrieval-augmented generation with enhanced data diversity. *arXiv preprint arXiv:2410.12248*, 2024.
- [144] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From local to global: A graph rag approach to query-focused summarization, 2025.
- [145] Michael J Ryan, Danmei Xu, Chris Nivera, and Daniel Campos. Enronqa: Towards personalized rag over private documents. *arXiv preprint arXiv:2505.00263*, 2025.
- [146] Mirco Bonomo and Simone Bianco. Visual rag: Expanding mllm visual knowledge without fine-tuning. *arXiv* preprint arXiv:2501.10834, 2025.
- [147] Costas Mavromatis, Soji Adeshina, Vassilis N Ioannidis, Zhen Han, Qi Zhu, Ian Robinson, Bryan Thompson, Huzefa Rangwala, and George Karypis. Byokg-rag: Multi-strategy graph retrieval for knowledge graph question answering. *arXiv preprint arXiv:2507.04127*, 2025.
- [148] Jimeng Shi, Sizhe Zhou, Bowen Jin, Wei Hu, Shaowen Wang, Giri Narasimhan, and Jiawei Han. Hypercube-rag: Hypercube-based retrieval-augmented generation for in-domain scientific question-answering. *arXiv* preprint *arXiv*:2505.19288, 2025.

- [149] Xuejiao Zhao, Siyan Liu, Su-Yin Yang, and Chunyan Miao. Medrag: Enhancing retrieval-augmented generation with knowledge graph-elicited reasoning for healthcare copilot. In *Proceedings of the ACM on Web Conference* 2025, pages 4442–4457, 2025.
- [150] Zhikun Xu, Yinghui Li, Ruixue Ding, Xinyu Wang, Boli Chen, Yong Jiang, Hai-Tao Zheng, Wenlian Lu, Pengjun Xie, and Fei Huang. Let llms take on the latest challenges! a chinese dynamic question answering benchmark. *arXiv preprint arXiv:2402.19248*, 2024.
- [151] Xiaodong Yu, Hao Cheng, Xiaodong Liu, Dan Roth, and Jianfeng Gao. Reeval: Automatic hallucination evaluation for retrieval-augmented large language models via transferable adversarial attacks. *arXiv preprint arXiv:2310.12516*, 2023.
- [152] Mohita Chowdhury, Yajie Vera He, Jared Joselowitz, Aisling Higham, and Ernest Lim. Astrid–an automated and scalable triad for the evaluation of rag-based clinical question answering systems. arXiv preprint arXiv:2501.08208, 2025.
- [153] David Rau, Hervé Déjean, Nadezhda Chirkova, Thibault Formal, Shuai Wang, Vassilina Nikoulina, and Stéphane Clinchant. Bergen: A benchmarking library for retrieval-augmented generation. arXiv preprint arXiv:2407.01102, 2024.
- [154] Matin Mortaheb, Mohammad A Amir Khojastepour, Srimat T Chakradhar, and Sennur Ulukus. Rag-check: Evaluating multimodal retrieval augmented generation performance. *arXiv preprint arXiv:2501.03995*, 2025.
- [155] Zehan Qi, Rongwu Xu, Zhijiang Guo, Cunxiang Wang, Hao Zhang, and Wei Xu. Long ² rag: Evaluating long-context & long-form retrieval-augmented generation with key point recall. arXiv preprint arXiv:2410.23000, 2024.
- [156] Hailey Joren, Jianyi Zhang, Chun-Sung Ferng, Da-Cheng Juan, Ankur Taly, and Cyrus Rashtchian. Sufficient context: A new lens on retrieval augmented generation systems. *arXiv* preprint arXiv:2411.06037, 2024.
- [157] Jia-Huei Ju, Suzan Verberne, Maarten de Rijke, and Andrew Yates. Controlled retrieval-augmented context evaluation for long-form rag. *arXiv preprint arXiv:2506.20051*, 2025.
- [158] Ilias Driouich, Hongliu Cao, and Eoin Thomas. Diverse and private synthetic datasets generation for rag evaluation: A multi-agent framework. *arXiv preprint arXiv:2508.18929*, 2025.
- [159] Quentin Romero Lauro, Shreya Shankar, Sepanta Zeighami, and Aditya Parameswaran. Rag without the lag: Interactive debugging for retrieval-augmented generation pipelines. *arXiv preprint arXiv:2504.13587*, 2025.
- [160] Yixiao Zeng, Tianyu Cao, Danqing Wang, Xinran Zhao, Zimeng Qiu, Morteza Ziyadi, Tongshuang Wu, and Lei Li. Rare: Retrieval-aware robustness evaluation for retrieval-augmented generation systems. *arXiv* preprint *arXiv*:2506.00789, 2025.
- [161] Kepu Zhang, Zhongxiang Sun, Weijie Yu, Xiaoxue Zang, Kai Zheng, Yang Song, Han Li, and Jun Xu. Qe-rag: A robust retrieval-augmented generation benchmark for query entry errors. *arXiv* preprint arXiv:2504.04062, 2025.
- [162] Jie Ouyang, Tingyue Pan, Mingyue Cheng, Ruiran Yan, Yucong Luo, Jiaying Lin, and Qi Liu. Hoh: A dynamic benchmark for evaluating the impact of outdated information on retrieval-augmented generation. *arXiv* preprint *arXiv*:2503.04800, 2025.
- [163] Yuanjie Lyu, Zhiyu Li, Simin Niu, Feiyu Xiong, Bo Tang, Wenjin Wang, Hao Wu, Huanyong Liu, Tong Xu, and Enhong Chen. Crud-rag: A comprehensive chinese benchmark for retrieval-augmented generation of large language models. *ACM Transactions on Information Systems*, 43(2):1–32, 2025.
- [164] Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. Retrieval-augmented generation with conflicting evidence. *arXiv preprint arXiv:2504.13079*, 2025.
- [165] Shuai Wang, Ekaterina Khramtsova, Shengyao Zhuang, and Guido Zuccon. Feb4rag: Evaluating federated search in the context of retrieval augmented generation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 763–773, 2024.
- [166] Shuting Wang, Jiongnan Liu, Shiren Song, Jiehan Cheng, Yuqi Fu, Peidong Guo, Kun Fang, Yutao Zhu, and Zhicheng Dou. Domainrag: A chinese benchmark for evaluating domain-specific retrieval-augmented generation. *arXiv preprint arXiv:2406.05654*, 2024.
- [167] Yi Liu, Lianzhe Huang, Shicheng Li, Sishuo Chen, Hao Zhou, Fandong Meng, Jie Zhou, and Xu Sun. Recall: A benchmark for llms robustness against external counterfactual knowledge. arXiv preprint arXiv:2311.08147, 2023.
- [168] Mathis Pink, Vy A. Vo, Qinyuan Wu, Jianing Mu, Javier S. Turek, Uri Hasson, Kenneth A. Norman, Sebastian Michelmann, Alexander Huth, and Mariya Toneva. Testing memory capabilities in large language models with the sequential ordered recall task. In *Latinx in AI @ NeurIPS 2024*, 2024.

- [169] Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning, 2023.
- [170] Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models, 2023.
- [171] Petr Anokhin, Nikita Semenov, Artyom Sorokin, Dmitry Evseev, Andrey Kravchenko, Mikhail Burtsev, and Evgeny Burnaev. Arigraph: Learning knowledge graph world models with episodic memory for llm agents, 2025.
- [172] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization, 2020.
- [173] Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Shi Jie, Xiang Wang, Xiangnan He, and Tat seng Chua. Alphaedit: Null-space constrained knowledge editing for language models, 2025.
- [174] Da Ma, Lu Chen, Situo Zhang, Yuxun Miao, Su Zhu, Zhi Chen, Hongshen Xu, Hanqi Li, Shuai Fan, Lei Pan, and Kai Yu. Compressing kv cache for long-context llm inference with inter-layer attention similarity. *arXiv* preprint arXiv:2412.02252, 2024.
- [175] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. Unsupervised dense information retrieval with contrastive learning, 2022.
- [176] Haoran Tan, Zeyu Zhang, Chen Ma, Xu Chen, Quanyu Dai, and Zhenhua Dong. Membench: Towards more comprehensive evaluation on the memory of llm-based agents, 2025.
- [177] Mathis Pink, Vy A. Vo, Qinyuan Wu, Jianing Mu, Javier S. Turek, Uri Hasson, Kenneth A. Norman, Sebastian Michelmann, Alexander Huth, and Mariya Toneva. Assessing episodic memory in llms with sequence order recall tasks. *arXiv preprint*, 2024.
- [178] Dong Gong, Lingqiao Liu, Vuong Le, Budhaditya Saha, Moussa Reda Mansour, Svetha Venkatesh, and Anton van den Hengel. Memorizing normality to detect anomaly: Memory-augmented deep autoencoder for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1705–1714, 2019.
- [179] Yanai Elazar, Shauli Ravfogel, Alon Jacovi, and Yoav Goldberg. Amnesic probing: Behavioral explanation with amnesic counterfactuals, 2021.

AppendicesA Minimal MRD YAML template

```
mrd_version: "0.1"
study_id: "<short-identifier>"
claim_type: ["capability_change", "freshness_related"] # pick one or both
temporal_governance:
 train:
    window: "YYYY-MM-DD..YYYY-MM-DD"
    snapshot_date: "YYYY-MM-DD"
    sources: ["<corpus1>", "<corpus2>"]
 index_or_session_store:
   snapshot_date: "YYYY-MM-DD"
    dedup_strategy: "<e.g., MinHash@J=0.85, URL+title exact>"
    update:
     last_time: "YYYY-MM-DDThh:mm:ssZ"
      frequency: "<e.g., daily | weekly | ad-hoc>"
  test:
    window: "YYYY-MM-DD..YYYY-MM-DD"
    snapshot_date: "YYYY-MM-DD"
    contamination_notes: "<potential benchmark leakage & mitigation>"
leakage_overlap_auditing:
 methods: ["<e.g., SimHash, BM25-topk cross-match>"]
  thresholds: {"near_dup_jaccard": 0.85}
  exclusion_criteria: "<rules applied>"
 impact_fraction:
   train∩test: 0.012
   train∩index: 0.034
    index∩test: 0.007
implementation_resources:
 model:
   name: "<model-family>"
    checkpoint: "<hash/tag>"
    decoding_hparams: {temperature: 0.7, top_p: 0.95, max_new_tokens: 512}
   random_seeds: [2024, 2025, 2026]
    script_versions: {"runner": "v1.3.2", "evaluator": "v0.9.1"}
 retrieval:
   retriever_type: "<e.g., BM25 | DPR | ColBERT | hybrid>"
   fusion_strategy: "<e.g., RRF@60 | concat-then-rerank>"
   reranker_type: "<e.g., Cross-Encoder-msmarco>"
    core_params: {"max_passages": 50}
 hardware_cost:
    accelerators: {"A100_80G": 8}
    total_gpu_hours: 120.5
   per_unit_cost_usd: 1.8
    token_budget: {"prompt_tokens": 1.2e8, "completion_tokens": 6.5e7}
regimes: # ensure shared slice/time-window when comparing
  - name: "PO"
    shared_slice_with: ["Offline", "Online"]
  - name: "Offline"
  - name: "Online"
limitations_and_robustness:
 undisclosed_fields: ["<e.g., exact costs>"]
 reason: "<compliance | NDA | privacy>"
 sensitivity_analyses: ["<bri>brief pointer to what was varied and the effect>"]
```