Semantics Delivery Network: Rethinking Web Retrieval Infrastructure for LLM Agents

Peichun Hua
The Chinese University of Hong Kong, Shenzhen
Shenzhen, Guangdong, China
peichunhua@link.cuhk.edu.cn

Yunming Xiao*
The Chinese University of Hong Kong, Shenzhen
Shenzhen, Guangdong, China
yunmingxiao@cuhk.edu.cn

Abstract

Large language models (LLMs) increasingly ground their answers in proprietary data and live web content through both single-shot retrieval-augmented generation (RAG) and multi-turn agentic search. Yet the web path they use was built for human readers: search services return URLs and snippets ranked by generic relevance, and content delivery networks (CDNs) cache URL-addressed objects without knowing which passage an agent needs. LLMs instead consume short, semantically coherent passages—hereafter, *chunks*—selected for downstream utility rather than similarity alone, retrieve statefully across reasoning turns, and, as uncoordinated streams, repeatedly fetch, clean, chunk, and embed the same content.

We argue that *semantic chunk retrieval should become a first-class network-delivery abstraction*, and propose the *Semantics Delivery Network* (SemDN): an origin-authorized, hierarchical substrate that indexes, searches, and caches web content at chunk granularity, serving clients and origins alike. SemDN is more than finer-grained caching: it must estimate missing or stale evidence without an explicit miss signal, let clients choose retrieval representations that origins need not see, and share state among clients with the same representation. Preliminary results show that page-first retrieval processes two to three orders of magnitude more content than agents consume, that agentic queries exhibit task-local chunk reuse invisible to URL caches, and that chunk delivery improves answer quality per context token. We organize an open research agenda around shareable retrieval state, hierarchical caching and refresh under coverage risk, and deployability, including incentives, privacy, and verifiability. Our code is available at: https://github.com/sarendis56/SemDN_HotNets26.

*Corresponding author



This work is licensed under a Creative Commons Attribution 4.0 International License.

HotNets '26, Salt Lake City, UT, USA

© 2026 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-3029-0/2026/11
https://doi.org/10.1145/3847290.3859940

CCS Concepts

• **Networks** → **Network architectures**; **Network services**; *Overlay and other logical network structures*; **Network measurement**; • **Information systems** → **Web searching and information discovery**.

Keywords

Content Delivery Networks, Retrieval-Augmented Generation, Agentic Search, Semantic Caching, Vector Search

ACM Reference Format:
Peichun Hua and Yunming Xiao. 2026. Semantics Delivery Network: Rethinking Web Retrieval Infrastructure for LLM Agents. In *The 25th ACM Workshop on Hot Topics in Networks (HotNets '26), November 16-17, 2026, Salt Lake City, UT, USA.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3847290.3859940

1 Introduction

Machine-generated retrieval is becoming an important part of the web's read path. LLMs retrieve external content to ground answers, from single-shot RAG [65] to multi-turn *agentic search* that interleaves reasoning and retrieval [52, 73]. Increasingly, this retrieval targets the *live* web: browsing agents and freshness-sensitive benchmarks make static corpora insufficient [63, 69, 80, 113, 118, 120], while query-time fetcher traffic is rising sharply [29, 35, 132].

Yet these machine clients retrieve through infrastructure designed for human readers. Discovery APIs expose URL- and snippet-oriented results optimized for generic relevance; CDNs cache URL-addressed objects without knowing which passage an agent needs. An LLM instead consumes short semantic chunks[1] rather than pages [89, 131]. It cares about downstream *utility*, not similarity alone [16, 31, 73, 99]; retrieves statefully across related sub-queries; and belongs to an uncoordinated population that repeatedly downloads, cleans, and embeds the same content, pushing redundant work onto origins [35].

Today, this path is split across two systems (Fig. 1a). A search engine serves the client and decides *what* to read,

[1] A *chunk* broadly denotes a semantic unit; this may be a text passage [58], an image region [36, 90] or page screenshot [116], a table block [44], an audio clip [121], or a video segment [76].

but it returns only URLs. A CDN serves the origin with edge placement, offload, freshness management [2], and even origin logic at the edge [27, 68], but it never sees the query. We argue that *semantic chunk retrieval should become a first-class network-delivery abstraction.* We propose the *Semantics Delivery Network* (SemDN), a semantic delivery substrate that fuses both roles: like a search engine, it answers clients' queries; like a CDN, it delivers origin content from the edge. It encodes, indexes, searches, and caches web content at chunk granularity, returning the passages an LLM needs rather than pages it must download and post-process.

SemDN could be built atop an existing CDN or as a separate system, but it is more than finer-grained caching. Keying delivery on queries instead of URLs changes three things. (1) *Misses:* a URL lookup misses on an exact name, whereas top-k search always answers, so missing or stale evidence must be estimated. (2) *Tenancy:* origins still enroll for offload and protection, as with a CDN, but each client chooses the encoder and index it retrieves with, a choice origins need not see. (3) *State:* a cached page serves every client, whereas index and cache state serve only clients sharing a representation; only acquisition and normalization are shared by all. Each difference calls for careful design to attain performance and efficiency.

Two recent results make this new paradigm increasingly practical. First, recompute-based indexing, such as EdgeRAG and LEANN, shows that embeddings need not all be stored but can be regenerated on demand during search [95, 117]. An edge can therefore keep content blocks and materialize model-specific vectors only where demand justifies them. Second, agent query streams can be bursty and task-coherent, creating locality that an edge hierarchy may exploit for placement and reuse [32, 64].

Prior systems separately expose retrieved chunks as a service [1], build managed indexes over tenant corpora [26], tier retrieval across edge and cloud for one deployment [66], or reduce vector-index storage through on-demand recomputation [95, 117]; LLM-serving caches store responses or KV states per application [8, 53]. Concurrent visions have origins expose chunks behind a DNS-like semantic resolver [12] or distribute KV caches like CDN content [22, 92], and a major CDN now builds opt-in per-site AI indexes [24]. SemDN instead delivers content-level evidence ranked across origins, for clients as well as origins: it joins semantic selection with an origin-authorized hierarchical delivery and freshness path, sharing acquisition and normalized content across tenants while exposing semantic demand to placement and refresh decisions.

This paper makes three contributions. **(1)** We characterize today's LLM web-retrieval path and distill four structural mismatches between that workload and the search-engine/CDN stack, including its redundant, failure-prone preprocessing (§3). **(2)** We propose Semantics Delivery Network (SemDN), a new edge retrieval substrate, and contrast its data path with today's (§4, Fig. 1). **(3)** We organize its research agenda around shareable retrieval state, hierarchical caching and refresh under coverage risk, and two-sided, verifiable deployment (§5), and report preliminary evidence (§6): page-first retrieval processes two to three orders of magnitude more content than agents consume, agent queries exhibit task-local chunk reuse invisible to URL caches, and chunk delivery matches or beats page retrieval at far smaller budgets.

2 Background

2.1 Content Delivery Networks

A CDN caches origin content at edge points of presence (PoPs), keyed by URL and expired by a time-to-live (TTL) [2]. Its value is two-sided in effect but one-sided in billing: clients enjoy lower latency, while content owners, who pay the CDN, offload origin traffic, absorb flash crowds, and gain DoS protection. However, existing CDNs treat content as opaque bytes, placing and evicting objects by popularity and recency instead of semantic relationships; although CDNs now run more origin logic and defense [27, 68, 122, 123], placement remains URL-based and semantics-agnostic.

2.2 Retrieval for LLMs

RAG augments an LLM by retrieving external passages from proprietary sources or the live web, and placing them in the model's context [65]. A corpus is segmented into *chunks* (typically a few hundred tokens [89, 131]), each mapped to a dense vector by an *encoder*, e.g., E5 [115] or BGE [17]. A query is embedded by (typically) the same encoder, and an approximate nearest-neighbor search (ANNS) system returns the top-k chunks using a graph index such as HNSW [74, 77], a clustering-based index [9, 50], or low-bit representations [39, 47, 48, 129, 130].

Agentic search interleaves reasoning and retrieval over multiple turns: the agent identifies missing information, queries, incorporates returned passages, and repeats, turning one question into a task-coherent sequence [5, 52, 120]. Yet similar passages may carry nothing useful and can mislead reasoning [30], so the best retriever may be trained on downstream correctness and coupled with the agent [73, 99].

2.3 How LLMs Retrieve From Web Today

A common production path for LLM web retrieval is a two-stage pipeline (Fig. 1a). A search engine results page (SERP) API (e.g., Serper [96], Brave [14], Exa [34]) maps a query to ranked URLs and snippets but does not return content. A separate reader (e.g., Firecrawl [37], Crawl4AI [110], Jina [55, 114], Tavily [104]) fetches those pages, strips boilerplate, and returns clean text. The application then chunks, embeds,

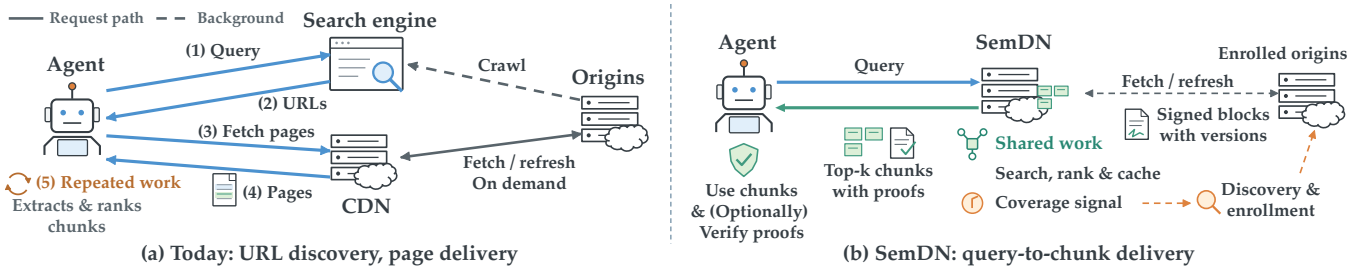


Figure 1: Today's retrieval path versus SemDN. (a) A search engine returns URLs; the agent fetches whole pages through a semantics-agnostic CDN and extracts chunks itself. (b) SemDN answers the query with top-k chunks from a shared index over enrolled origins, optionally with proofs; a coverage signal triggers discovery and enrollment.

ranks, reasons, and may issue another query [52, 73, 101]. Without cross-provider reuse or origin-integrated freshness, each application assembles its own pipeline or adopts a provider's.

3 Observations

We distill the gap between this workload and its infrastructure into four observations.

O1: Objective mismatch—similarity is not utility. Mainstream discovery services [14, 96] return results ranked by generic relevance (such as lexical similarity), while off-the-shelf encoders [94] optimize query–passage *similarity*; agents instead need passages that are *useful* for producing a correct answer. The two differ: similarity-trained retrievers underperform utility-aware ones [73, 135], and highly similar passages can even derail reasoning [30]. Search-R1 adapts a search policy to a fixed retriever with answer-level RL; Agentic-R also trains the retriever on answer correctness [52, 73]. This trend motivates tenant-specific utility ranking rather than treating shared similarity as the final objective. Content acquisition and canonical candidates can be shared; a tenant's utility function and outcome labels need not be. In our probes, similarity-ranked page-first retrieval leaves useful chunks unreached and is matched or outperformed by chunk-first retrieval (§6.3); whether utility-customized pipelines widen this gap is future work.

O2: Granularity mismatch—pages versus chunks. The web is authored, crawled, and delivered at the granularity of pages and their resource-heavy objects, whereas LLMs consume shorter passages or chunks [1, 89, 134]. An HTML-only fetch still transfers the full document; browser-rendering agents may additionally retrieve its subresources—the median desktop page is now ≈3 MB across roughly 70–77 requests [46], while the passage a model wants is typically a few kilobytes. Many production agents even render complete pages and operate over *screenshots* or other browser-level representations [4, 41, 59, 83, 84, 116].

Chunking is also a *budget* decision: stuffing whole pages into the model's context inflates inference cost and can degrade answer quality [72]. How content is chunked is itself improvable, from fixed-size windows and neural text segmentation [75, 89] to LLM-driven chunkers that split at logical boundaries or into self-contained propositions [20, 21, 33, 134], and the best granularity varies per query [128, 136]. Existing mitigations optimize this work on the client [13, 21, 71, 103] or convert whole pages to Markdown at the CDN [28]; we ask whether a shared edge substrate can instead perform it once, well, and amortize it (§4).

O3: Locality and policy mismatch—URL versus semantics. CDN caching keys on URL and recency; LLM retrieval keys on semantics and can exhibit locality that the current stack cannot see. AI scrapers and search agents already degrade URL caches: even a search agent draws 70–100% unseen URLs within a task [132], yet in our traces 32–55% of a new query's chunks recur within the task (§6.2). Reuse invisible to URLs reappears at chunk granularity. Query streams in RAG workloads are bursty and task-clustered [32, 64, 120]: within one task and across similar tasks, retrieved chunks recur across turns. Separating discovery from URL-object delivery prevents the delivery layer from directly exploiting this semantic locality for cross-client placement and eviction.

O4: Redundant, failure-prone semantic preprocessing. Turning a URL into model-ready chunks is a heavy, multi-stage pipeline that requires reaching the page (often past bot walls [85], JavaScript shells, and timeouts), extracting content from a noisy web, stripping boilerplate, segmenting, and embedding [21, 38, 102, 103, 114]. If each application assembles this pipeline itself (§2.3), popular pages may be processed independently by many operators. The work is both substantial and unreliable: across 1,430 live SERP result URLs, a cloud renderer (Cloudflare Browser Rendering [25]) obtains usable clean text from at most 75%, after which the text must still be de-boilerplated and embedded—work repeated across tenants within a single freshness window while origins absorb the resulting fetch storms [35].

4 Semantics Delivery Network (SemDN)

4.1 Architecture

SemDN serves two kinds of customers. Origins enroll, as with a CDN, for offload, freshness management, and protection from machine traffic. Clients, or *tenants*, choose how enrolled content is chunked, encoded, and ranked, within the transformations origins permit. In the data path (Fig. 1b), a client (e.g., a RAG application or an agent) sends a *query*, not a URL, to its nearest edge. We assume a mature SemDN already maintains a large origin-authorized corpus, populated by enrolled origins and prior acquisition, rather than building it per query. The edge runs three core functions: encode (map queries and content candidates to vectors), search (over canonical or registered model-specific indexes), and cache (store and evict normalized blocks, materialized chunks, and index state). Borrowing the hierarchy that makes CDNs scalable, an edge PoP holds hot content and index state, regional parents hold larger, colder shards, and a global tier maintains the distributed catalog; resource selection can avoid searching every shard [3].

This hierarchy changes the meaning of a cache miss. A URL cache, like an information-centric network that routes on exact content names [49], has a binary exact-match miss; semantic search replaces names with ranked queries and always returns top-k results, even when newer or better evidence lies outside its index. We define a *semantic coverage miss* to occur when the estimated risk that external discovery would add fresh evidence and change the returned top-k exceeds a policy threshold. The coverage controller could combine retrieval-score distributions and result coherence [100, 107, 108, 112], judgments of whether the retrieved evidence suffices to answer [19, 56], cross-tier agreement, and freshness metadata as imperfect signals of coverage risk. A judgment reading only the query and evidence, not the tenant's generator, can run once in the network before generation. Periodically, or on a high-risk query, it invokes an external discovery service (e.g., a SERP API), as Corrective RAG does for low-confidence retrieval [124]; results are filtered against the origin-enrollment registry before SemDN fetches or revalidates absent or stale content. FreshCache gates per-client semantic-cache reuse on an estimated staleness probability [78]; SemDN applies a similar risk test to a shared, origin-authorized corpus. Calibrating this risk under workload and content drift is an open problem.

4.2 Design Decisions

Share normalized content and canonical candidates; specialize ranking per tenant (O1). Different chunkers and encoders induce different objects and proximity graphs, and a single vector index cannot express every top-k ranking [119]. SemDN caches versioned normalized content blocks (e.g., DOM paragraphs, table blocks, image regions, and provenance) from which supported chunking policies compose passages. A shared canonical index retrieves candidates; a tenant-provided utility reranker or a pre-registered model-specific encoder refines them, with specialized index state materialized only where demand justifies it. Tenant utility functions and outcome labels remain private (§5.1).

Serve and cache at chunk granularity (O2). The edge returns passages, not pages. Normalized blocks remain the stable shared objects; hot chunks composed under supported policies can be materialized and cached at the granularity the workload actually reuses (§6.1, §6.2).

Place and evict in the query stream (O3). Because SemDN sees the agent query stream, it can place and evict based on observed semantic reuse rather than URL popularity alone. How much query state to share across tenants is open (§5).

Do shared work once and specialized work once per registered representation (O4). Acquisition, cleaning, and normalization run once per content version; canonical embedding and indexing are shared across tenants, while registered model-specific state is amortized among tenants that use it. This relieves origins of redundant fetches (§6.2).

5 Architectural Research Agenda

SemDN raises three coupled questions: what content and retrieval state is **shareable** across heterogeneous tenants; how a hierarchical substrate stays **manageable** in caching and refreshing under semantic coverage misses; and what interfaces, incentives, and guarantees make it **deployable**.

5.1 What Retrieval State Is Shareable?

Representation. What stable *form* should the cached asset take? Beyond the versioned normalized blocks of §4, a canonical embedding or a Matryoshka representation, truncatable to smaller dimensions [60, 93], enables a shared first-stage candidate index but presumes an agreed encoder, or a bridge to it (§5.1). Even with recompute, the proximity graph is *metric-specific*: an HNSW graph encodes nearest-neighbor relations in one embedding space and does not transfer to another [77, 117], so a model-specific graph is shared only among tenants registered for the same representation.

Indexing and the storage-compute trade. SemDN's baseline uses a canonical index for candidate generation; sufficiently popular registered encoders may add model-specific indexes. For either option, storing embeddings versus recomputing them remains flexible. LEANN [117] and MHR [47] can reduce the storage requirement of graph-based indexes through lightweight structures [50], at the expense of additional GPU load. Where a real edge should sit on this surface and whether PoPs carry the assumed compute is left open.

Bridging encoders to the canonical index. A canonical index serves a tenant with its own encoder only if the two embedding spaces can be bridged. Compatible training makes a new encoder comparable to old embeddings [91, 98] but presumes control over how encoders are trained, which bring-your-own tenants deny. Query-side adapters instead project a tenant's query into the indexed space [23, 111], but an adapter fit to regress the tenant space onto the canonical one can undo much of the tenant's fine-tuning. Unsupervised translation reaches unrelated backbones without paired data [51], which also means tenant query embeddings are not opaque to the operator. All of these treat stored vectors as the only artifact. Because SemDN keeps normalized blocks and can recompute, it can instead decouple recall from ranking: the canonical index, possibly alongside an encoder-agnostic lexical index, need only place a tenant's true top-k within its top-K candidates ($k \ll K$), which the registered encoder re-encodes and ranks exactly. Re-encoded vectors of hot chunks are cached per encoder and grow into the model-specific index above where demand justifies it. How candidate recall degrades with encoder drift, and when re-encoding stops paying off, remain open.

5.2 How Should a Hierarchical SemDN Cache and Refresh?

Caching, locality, and coverage. This is the richest open area (O3). Where AI-aware web caches filter training scrapers and search agents alike as low-locality load [132], SemDN serves query-time retrieval, not training scrapes: an agentic task, or an organization's agents, forms a cohort whose reuse follows semantics. Agent trajectories reveal which chunks are reused together [64], while cross-tenant demand determines which content merits edge placement rather than a regional-only copy. Approximate reuse of retrieval results across similar queries [11, 43, 82] could extend hits beyond exact repeats.

The challenges include (1) *coverage skew* (only what is queried gets optimized, leaving the long tail under-served); (2) *cross-tenant leakage* (reweighting a shared structure from one tenant's trajectories can expose its task structure to others); and (3) *feedback entrenchment* (favoring previously successful chunks risks locking in stale answers). The controller must also distinguish an edge miss, which a regional or global parent can satisfy, from a system-wide semantic coverage miss that warrants external discovery: false confidence hides fresh evidence, and stale evidence mixed with current evidence degrades answers [86], while a conservative threshold restores redundant search and origin load.

Freshness. SemDN can adapt CDN-style freshness machinery to amortize *acquisition*. For example, a query that forces a re-fetch can refresh related chunks (spatial locality), and content warmed by one tenant's traffic serves the cohort. Versioned blocks also suit evolving documents, where amendments change few clauses but retrieval must select the right version [81]. However, freshness-critical content (news, prices) limits cache lifetimes; periodic TTL refresh provably leaves stale windows [67], graph indexes must absorb frequent in-place updates [70], and cold content may never be warmed by piggybacking. Therefore, we treat the achievable savings as something to estimate rather than assume.

5.3 What Interfaces, Incentives, and Guarantees Make SemDN Deployable?

Tenant interface. Today's agents retrieve from the web through SERP and reader APIs (§3) or managed indexes [26]. They return ranked snippets, cleaned text, or a synthesized answer with citations, as search engines' grounding APIs do [40], along with a bill, but generally expose limited control over *why* a passage was returned or how retrieval policy affects it. Yet the embedding model and chunking strategy materially change which passages are retrieved and how well the agent answers [15, 79], so a fixed, hidden choice can cap quality, and a per-query bill prices a process the client cannot inspect.

SemDN instead lets a client select a supported chunking policy over normalized blocks and either use canonical retrieval with a tenant utility reranker or register a model-specific encoder (§5.1). It returns selected passages with scores and provenance, making retrieval auditable and billable for what was consumed. Managed services already offer website ingestion, configurable chunking/models, reranking, citations, semantic query caching, and tenant isolation [26]; CDN operators add opt-in per-site AI indexes with change feeds [24]. Both are organized around the origin: one index and representation per site, federated by fan-out. SemDN instead ranks chunks across origins (§6.3) and shares origin-authorized acquisition and normalized content across independent retrieval clients, coordinating placement, freshness, and representation-specific state around client demand. For example, to compare two vendors' API limits, an agent today fetches and chunks five pages per turn; AI Search answers only from sites that deployed it, one index each; SemDN returns one top-k across both origins in the agent's chosen representation.

A two-sided service. A CDN is a reverse proxy paid by the origin; an enterprise web proxy is a forward proxy for its clients. **SemDN is both and can monetize two services.** To clients, including organizations whose agents share locality, it sells *retrieval quality and latency*: instead of fetching full pages and running the whole pipeline, RAG systems receive model-ready chunks, billed per chunk consumed and priced along the representation/accuracy surface (§5.1).

To content owners, it sells origin offload for machine traffic, analogous to a CDN: SemDN terminates agent fetches at the edge and serves only the relevant chunks rather than full pages, reducing both origin load and delivered bytes while retaining freshness management and DoS defense. An origin contract defines enrollment, permitted transformations, freshness/version bounds, purge and invalidation, access control (agent traffic is already identifiable at the TLS/HTTP layer [57]), and provenance; retrieval configuration is not part of it, being settled between SemDN and each client. The interface could expose aggregate machine-demand statistics while withholding individual queries. Pricing can also steer architecture: charging registered model-specific representations at their unshared cost while discounting a canonical representation gives tenants an incentive to use shareable state. Settlement is the main adoption hurdle: neither side pays merely because the other uses SemDN.

Verifiability. Acting for both sides creates conflicts a CDN has never faced: an operator paid by origins may rank paying origins higher, and a shared cache can leak tenants' queries through timing, as in shared prompt caches [42]. HTTPS authenticates only the endpoint and byte stream: a CDN already terminates TLS with the origin's delegated credentials [10], and a chunk is byte-identical to nothing the origin published, so a faulty edge can cherry-pick, suppress, or inject passages [137]. Clients must instead verify the *evidence-construction process*: that blocks come from their origin at a stated version (e.g., via HTTP message signatures [7]); that a chunk is composed of those blocks (via Merkle inclusion proofs); that the top-k is the honest result over a committed index, which also exposes pay-to-rank bias [6, 18, 88]; and that the index omits no enrolled origin, as transparency logs ensure for certificates [62]. Verifiable HNSW search takes 0.8–2 s per query [6]; amortizing audit-on-demand proofs [88] across cohorts and tiers is an open networking problem.

6 Preliminary Results

We report preliminary evidence along three axes. Two concern *cost*: the content volume an agent processes to obtain the text it consumes (§6.1; O2, O4), and the locality and cross-tenant amortization a chunk cache enables where a page cache cannot (§6.2; O3–O4). The third concerns answer *quality* (§6.3; O1–O2). Throughout, a *chunk* is a passage of ≈100–256 tokens and a *page* is its whole source web page; a *reranker* re-scores retrieved candidates.

6.1 The Page Tax: Bytes Processed vs. Used

We send 100 real queries (50 curated intents across ten categories plus 50 from Natural Questions [61]) through the production agent path of Fig. 1a: a SERP API returns the top-5 URLs, which we then fetch and process. Five distinct

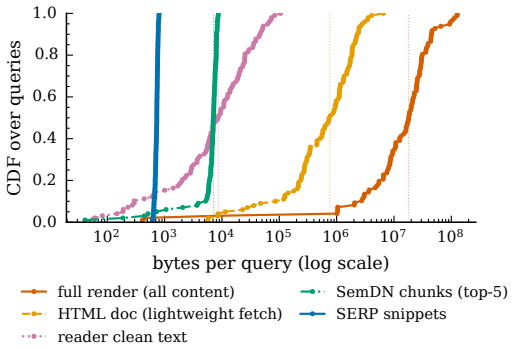


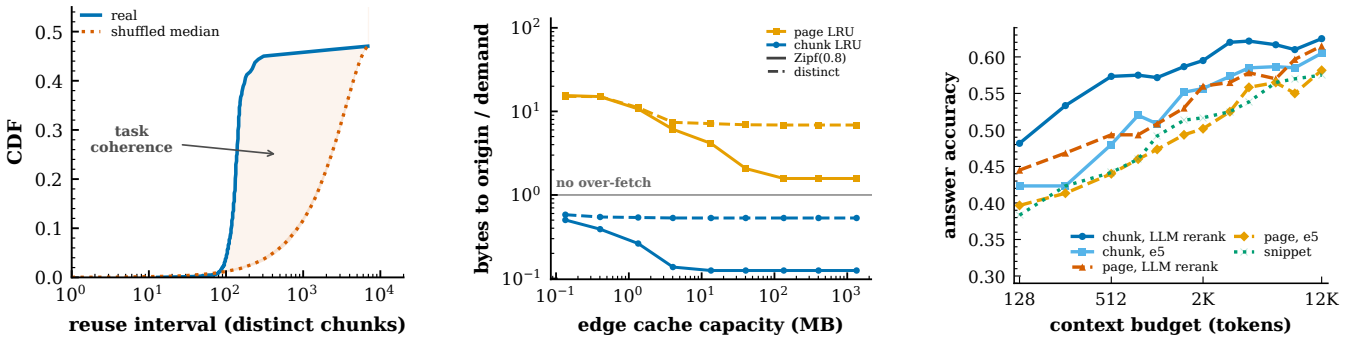
Figure 2: CDF of decompressed content bytes per query over the top-5 URLs (live web, 100 queries); values in the text are medians.

byte bands emerge (Fig. 2). At the median, the model consumes **7.1 KB** of selected chunks, while the lightweight no-JS baseline processes **761 KB** of decompressed HTML to obtain them (**107**×); a full browser render yields **18.1 MB** of decompressed content (**2,532**×, or 5.75 MB on the wire). Fetching is also brittle: at least 39% of the 500 sampled URLs lose over half their text without JavaScript and ≈19% are bot-blocked, making fetch-clean-chunk-embed heavy and failure-prone when performed independently by each client (O4).

6.2 Agentic Locality

Observed session locality. We run Search-R1 [52] on Qwen2.5-7B [125] over wiki-18 [54, 58] (21M chunks, 3.2M pages) with the fixed e5-base-v2 [115] backend to which its search policy was adapted [52]. The locality (O3) is large and *task-driven*: within a task, 30.5% of post-first-turn queries are byte-identical repeats, consistent with the redundant retrieval independently reported for Search-R1 [97], and even after deleting every repeat, 32–55% of a new query's chunks were already retrieved earlier in the task. Compared with popularity-preserving, task-shuffled controls, the real stream reuses chunks after 1–2 orders of magnitude fewer intervening chunks—structure a URL/TTL cache cannot key on (Fig. 3a).

Trace-driven cache-byte savings. We replay the stream against simulated PoP caches. In the evaluated URL-object cache, an entry is a source article sized as the sum of its chunk bytes, and a miss fetches the entire article. This favors the URL-object cache by excluding markup and subresources. A chunk-granularity LRU cache of just **1.34 MB** serves **73.8%** of chunk requests (87.6% at 13.4 MB). The URL-object LRU under page-first delivery reaches a comparable hit ratio only near 100 MB and fetches **6.9–15.5**× more bytes than the agent consumes; at equal cache capacity, the chunk cache sends **13–41**× fewer bytes to origin (Fig. 3b). The study replicates with the same agent over a realistic CRAG web corpus,



(a) Reuse interval (b) Cache efficiency (c) Answer quality

Figure 3: Agentic locality (a,b) and answer quality (c). (a) Reuse-interval CDF for distinct chunks; shaded band shows 5–95% over 50 popularity-preserving, task-shuffled controls. (b) Bytes to origin per demand byte vs. edge cache capacity. (c) Answer quality vs. reader context budget (CRAG Task-3, n=600).

where within-task reuse is stronger (63–79%) and the chunk cache sends 25–70× fewer origin bytes at equal size.

Modeled cross-tenant amortization. The top-10 domains draw 34% of results across our live set, suggesting potential overlap but not establishing that tenants request the same pages or chunks. We therefore feed the measured per-page and embedding costs (392 chunks/s per A100 GPU with e5-base-v2) into a Zipfian tenant-demand model, which estimates ≈6× less repeated fetch-clean-chunk-embed work at 50 tenants. Embedding and index work amortize only among tenants sharing a representation; a private encoder stays at 1×, though acquisition and normalized blocks remain shareable—the incentive for encoder-aware pricing (§5.3).

6.3 Answer Quality

On a fixed corpus (wiki-18) with five QA sets (HotpotQA [127], 2WikiMultiHopQA [45], MuSiQue [109], Bamboogle [87], and NQ [61]), we compare SemDN's chunks with a page-first client that downloads the top pages and passes the reader their best-scoring chunks. Both share the ranker (Search-R1's e5, alone or followed by Qwen3-Reranker-4B [133]) and the Qwen2.5-7B reader, and a Llama-3.1-70B judge [105] grades the generated answers with CRAG's metric [126]. Chunks match or beat the client even when it downloads ten pages (accuracy 0.456 vs. 0.427 with e5, 0.500 vs. 0.493 with the reranker; with a Qwen3-32B reader [106], 0.497 vs. 0.486 and 0.562 vs. 0.547), which moves ≈1,200× more bytes (3.8 MB of HTML vs. 3.2 KB, measured on 250 questions), or ≈3,000× when the reranker reads 30 candidate pages. On 300 time-stable HotpotQA questions, 16–29% of SemDN's global top-5 chunks lie on pages that a whole-page ranker does not select, even with 20 downloaded pages, which page-first designs would miss [12].

On the noisier, web-derived CRAG benchmark [126], with the same judge, we answer each question once with a Qwen3-32B reader [106] and vary only how the retrieved content is packed into the reader's context. With the same LLM reranker (Qwen3-Reranker-4B [133]) scoring chunks and whole pages, chunks beat pages by 3.5–8.2 accuracy points from 256 to 6,144 tokens and beat SERP snippets at every evaluated token budget (Fig. 3c). They even beat pages ordered with answer-bearing first by 6–9 points up to 768 tokens: a page spends budget on unrelated text [72]. The gap narrows above 8K tokens, where the context already holds most of each question's supporting evidence and further content mostly adds only noise.

7 Conclusion

LLM agents consume task-relevant passages, yet today's web stack discovers and delivers URL-addressed objects. We argue that semantic selection should become a first-class network-delivery abstraction, and we propose SemDN, a substrate that joins a search engine's query interface with a CDN's origin-authorized edge delivery and could be built atop an existing CDN or as a separate network. It is more than finer-grained caching because semantic delivery changes three things: misses must be estimated rather than observed, clients choose retrieval representations that origins need not see, and only acquisition and normalization, not indexes or caches, are shared by all. Our probes show that page-first retrieval processes far more content than agents consume, that agent queries exhibit task-local chunk reuse invisible to URL caches, and that chunk delivery improves quality per context token. Open challenges in shareable retrieval state, hierarchical caching and freshness under coverage risk, and deployability, including privacy, incentives, and verifiable evidence construction, define a networking research agenda.

References

[1] Shawqi Al-Maliki, Ammar Gharaibeh, Mohamed Rahouti, Mohammad Ruhul Amin, Mohamed Abdallah, Junaid Qadir, and Ala Al-Fuqaha. 2026. Budget-Constrained Online Retrieval-Augmented Generation: The Chunk-as-a-Service Model. *IEEE Transactions on Artificial Intelligence* (2026). doi:10.1109/TAI.2026.3666170

[2] Waris Ali, Chao Fang, and Akmal Khan. 2025. A survey on the state-of-the-art CDN architectures and future directions. *J. Netw. Comput. Appl.* 236 (2025), 104106. doi:10.1016/J.JNCA.2025.104106

[3] Robin Aly, Djoerd Hiemstra, and Thomas Demeester. 2013. Taily: Shard Selection Using the Tail of Score Distributions. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 673–682. doi:10.1145/2484028.2484033

[4] Anthropic. 2024. Introducing computer use, a new Claude 3.5 Sonnet, and Claude 3.5 Haiku. Anthropic Blog. Accessed: 2026-06-08. https://www.anthropic.com/news/3-5-models-and-computer-use

[5] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net. https://openreview.net/forum?id=hSyW5go0v8

[6] Nikolay Avramov, Hidde Lycklama, Alexander Viand, and Anwar Hithnawi. 2026. Atlas: Efficient Verifiable Semantic Search. arXiv:2609.11841 [cs.CR]

[7] Annabelle Backman, Justin Richer, and Manu Sporny. 2024. HTTP Message Signatures. *RFC* 9421 (2024), 1–95. doi:10.17487/RFC9421

[8] Fu Bang. 2023. GPTCache: An Open-Source Semantic Cache for LLM Applications Enabling Faster Answers and Cost Savings. In *Proceedings of the 3rd Workshop for Natural Language Processing Open Source Software (NLP-OSS 2023)*, Liling Tan, Dmitrijs Milajevs, Geeticka Chauhan, Jeremy Gwinnup, and Elijah Rippeth (Eds.). Association for Computational Linguistics, Singapore, 212–218. doi:10.18653/v1/2023.nlposs-1.24

[9] Dmitry Baranchuk, Artem Babenko, and Yury Malkov. 2018. Revisiting the Inverted Indices for Billion-Scale Approximate Nearest Neighbors. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XII (Lecture Notes in Computer Science, Vol. 11216)*, Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss (Eds.). Springer, 209–224. doi:10.1007/978-3-030-01258-8_13

[10] Richard Barnes, Subodh Iyengar, Nick Sullivan, and Eric Rescorla. 2023. *Delegated Credentials for TLS and DTLS*. RFC 9345. RFC Editor. doi:10.17487/RFC9345

[11] Shai Bergman, Anne-Marie Kermarrec, Diana Petrescu, Rafael Pires, Mathis Randl, Martijn de Vos, and Ji Zhang. 2025. Leveraging Approximate Caching for Faster Retrieval-Augmented Generation. In *Proceedings of the 26th International Middleware Conference, MIDDLEWARE 2025, Vanderbilt University, Nashville, TN, USA, December 15-19, 2025*, Mohammad Sadoghi and Lydia Y. Chen (Eds.). ACM, 340–353. doi:10.1145/3721462.3770776

[12] Muhammad Bilal, Zafar Qazi, and Marco Canini. 2025. Toward an AI-Native Internet: Rethinking the Web Architecture for Semantic Retrieval. arXiv:2511.18354

[13] Arth Bohra, Manvel Saroyan, Danil Melkozerov, Vahe Karufanyan, Gabriel Maher, Pascal Weinberger, Artem Harutyunyan, and Giovanni Campagna. 2025. WebLists: Extracting Structured Information From Complex Interactive Websites Using Executable LLM Agents. arXiv:2504.12682 [cs.AI]

[14] Brave. 2026. Brave Search API. https://brave.com/search/api/. Accessed: 2026-06-23.

[15] Laura Caspari, Kanishka Ghosh Dastidar, Saber Zerhoudi, Jelena Mitrovic, and Michael Granitzer. 2024. Beyond Benchmarks: Evaluating Embedding Model Similarity for Retrieval Augmented Generation Systems. In *Proceedings of the Workshop Information Retrieval's Role in RAG Systems (IR-RAG 2024) co-located with the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2024), Washington DC, USA, 07 18, 2024 (CEUR Workshop Proceedings, Vol. 3784)*, Fabio Petroni, Federico Siciliano, Fabrizio Silvestri, and Giovanni Trappolini (Eds.). CEUR-WS.org, 62–70.

[16] Manish Chandra, Debasis Ganguly, and Iadh Ounis. 2026. LURE-RAG: Lightweight Utility-driven Reranking for Efficient RAG. arXiv:2601.19535 [cs.IR]

[17] Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-Embedding: Multi-Linguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*. Association for Computational Linguistics, Bangkok, Thailand, 2318–2335. doi:10.18653/v1/2024.findings-acl.137

[18] Qian Chen, Haibo Hu, and Jianliang Xu. 2013. Authenticating Top-k Queries in Location-based Services with Confidentiality. *Proc. VLDB Endow.* 7, 1 (2013), 49–60. doi:10.14778/2732219.2732224

[19] Suting Chen, Peichun Hua, and Yunming Xiao. 2026. Relevance Is Not Sufficient Evidence: Detecting Evidence Gaps Before Generation in RAG. arXiv:2609.37469 [cs.CL]

[20] Tong Chen, Hongwei Wang, Sihao Chen, Wenhao Yu, Kaixin Ma, Xinran Zhao, Hongming Zhang, and Dong Yu. 2024. Dense X Retrieval: What Retrieval Granularity Should We Use?. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, 15159–15177. doi:10.18653/V1/2024.EMNLP-MAIN.845

[21] Yihan Chen, Benfeng Xu, Xiaorui Wang, and Zhendong Mao. 2025. An Index-based Approach for Efficient and Effective Web Content Extraction. arXiv:2512.06641

[22] Yihua Cheng, Kuntai Du, Jiayi Yao, and Junchen Jiang. 2024. Do Large Language Models Need a Content Delivery Network? arXiv:2409.13761

[23] Jun Woo Chung and Weijie Zhao. 2026. On Reranking Space for Multi-Tenant Retrieval with Adapted Queries. In *Proceedings of the 20th ACM Conference on Recommender Systems (RecSys '26)*. ACM, 400–409. doi:10.1145/3773078.3831781

[24] Cloudflare. 2025. An AI Index for All Our Customers. https://blog.cloudflare.com/an-ai-index-for-all-our-customers/. Accessed: 2026-09-27.

[25] Cloudflare. 2026. Browser Rendering: Markdown endpoint. https://developers.cloudflare.com/browser-rendering/rest-api/markdown-endpoint/.

[26] Cloudflare. 2026. Cloudflare AI Search. https://developers.cloudflare.com/ai-search/. Accessed: 2026-06-23.

[27] Cloudflare. 2026. Examples · Cloudflare Workers docs. Cloudflare Developer Docs. Accessed: 2026-06-08. https://developers.cloudflare.com/workers/examples/

[28] Cloudflare. 2026. Introducing Markdown for Agents. https://blog.cloudflare.com/markdown-for-agents/. Accessed: 2026-09-27.

[29] Cloudflare. 2026. No hallucinations here: track the latest AI trends with expanded insights on Cloudflare Radar. Cloudflare Radar. https://blog.cloudflare.com/expanded-ai-insights-on-cloudflare-radar/

[30] Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The Power of Noise: Redefining Retrieval

for RAG Systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024*. ACM, 719–729. doi:10.1145/3626772.3657834

[31] Lu Dai, Yijie Xu, Jinhui Ye, Hao Liu, and Hui Xiong. 2025. SePer: Measure Retrieval Utility Through The Lens Of Semantic Perplexity Reduction. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net. https://openreview.net/forum?id=ixMBnOhFGd

[32] Yongheng Deng, Tianyuan Jiang, Zhenya Ma, Hao Wu, Yongjian Fu, Hao Pan, Sheng Yue, and Ju Ren. 2026. Accelerating Graph-Based RAG Retrieval via Locality-Aware Device-Cloud Collaboration. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, KDD 2026, Jeju Island, Korea, August 9-13, 2026*. ACM, 839–850. doi:10.1145/3770855.3817674

[33] André V. Duarte, João D. S. Marques, Miguel Graça, Miguel Freire, Lei Li, and Arlindo L. Oliveira. 2024. LumberChunker: Long-Form Narrative Document Segmentation. In *Findings of the Association for Computational Linguistics: EMNLP 2024, Miami, Florida, USA, November 12-16, 2024 (Findings of ACL, Vol. EMNLP 2024)*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, 6473–6486. doi:10.18653/V1/2024.FINDINGS-EMNLP.377

[34] Exa. 2026. Exa: Web Search API, AI Search Engine, and Website Crawler. https://exa.ai/. Accessed: 2026-06-23.

[35] Fastly. 2025. Q2 2025 Threat Insights Report: The Rise of AI Crawlers and Fetchers. Fastly Threat Research. https://www.fastly.com/blog/ai-bots-q2-2025-trends-fastlys-threat-insights-report

[36] Manuel Faysse, Hugues Sibille, Tony Wu, Bilel Omrani, Gautier Viaud, Céline Hudelot, and Pierre Colombo. 2024. ColPali: Efficient Document Retrieval with Vision Language Models. arXiv:2407.01449 [cs.IR]

[37] Firecrawl. 2026. Firecrawl: Turn Websites into LLM-Ready Data. https://firecrawl.dev/. Accessed: 2026-06-23.

[38] Murrough Foley. 2026. WCXB: A Multi-Type Web Content Extraction Benchmark. arXiv:2605.21097

[39] Tiezheng Ge, Kaiming He, Qifa Ke, and Jian Sun. 2013. Optimized Product Quantization for Approximate Nearest Neighbor Search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2946–2953. doi:10.1109/CVPR.2013.379

[40] Google. 2026. Grounding with Google Search, Gemini API. https://ai.google.dev/gemini-api/docs/google-search. Accessed: 2026-10-18.

[41] Google DeepMind. 2024. Introducing Gemini 2.0: our new AI model for the agentic era. Google Blog. Accessed: 2026-06-08. https://blog.google/innovation-and-ai/models-and-research/google-deepmind/google-gemini-ai-update-december-2024/

[42] Chenchen Gu, Xiang Lisa Li, Rohith Kuditipudi, Percy Liang, and Tatsunori Hashimoto. 2025. Auditing Prompt Caching in Language Model APIs. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025 (Proceedings of Machine Learning Research, Vol. 267)*, Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (Eds.). PMLR / OpenReview.net. https://proceedings.mlr.press/v267/gu25b.html

[43] Peizhen Guo, Bo Hu, Rui Li, and Wenjun Hu. 2018. FoggyCache: Cross-Device Approximate Computation Reuse. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking, MobiCom 2018, New Delhi, India*. ACM, 19–34. doi:10.1145/3241539.3241557

[44] Jonathan Herzig, Thomas Müller, Syrine Krichene, and Julian Eisenschlos. 2021. Open Domain Question Answering over Tables via Dense Retrieval. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*. Association for Computational Linguistics, 512–519. doi:10.18653/v1/2021.naacl-main.43

[45] Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing A Multi-hop QA Dataset for Comprehensive Evaluation of Reasoning Steps. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*. International Committee on Computational Linguistics, 6609–6625. doi:10.18653/V1/2020.COLING-MAIN.580

[46] HTTP Archive. 2026. *Page Weight*. Accessed: 2026-06-08. https://httparchive.org/reports/page-weight

[47] Peichun Hua and Yunming Xiao. 2026. Matryoshka Hash Representations for Model-Aware Compact Semantic Retrieval. arXiv:2609.07276 [cs.IR]

[48] Peichun Hua and Yunming Xiao. 2026. Spruce: Scalable Private Outsourced Retrieval Using Compact Embeddings. arXiv:2609.03376 [cs.CR]

[49] Van Jacobson, Diana K. Smetters, James D. Thornton, Michael F. Plass, Nicholas H. Briggs, and Rebecca Braynard. 2009. Networking named content. In *Proceedings of the 2009 ACM Conference on Emerging Networking Experiments and Technology, CoNEXT 2009, Rome, Italy, December 1-4, 2009*, Jörg Liebeherr, Giorgio Ventre, Ernst W. Biersack, and Srinivasan Keshav (Eds.). ACM, 1–12. doi:10.1145/1658939.1658941

[50] Hervé Jégou, Matthijs Douze, and Cordelia Schmid. 2011. Product Quantization for Nearest Neighbor Search. *IEEE Trans. Pattern Anal. Mach. Intell.* 33, 1 (2011), 117–128. doi:10.1109/TPAMI.2010.57

[51] Rishi Jha, Collin Zhang, Vitaly Shmatikov, and John X. Morris. 2025. Harnessing the Universal Geometry of Embeddings. arXiv:2505.12540 [cs.LG]

[52] Bowen Jin, Hansi Zeng, Zhenrui Yue, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-R1: Training LLMs to Reason and Leverage Search Engines with Reinforcement Learning. arXiv:2503.09516

[53] Chao Jin, Zili Zhang, Xuanlin Jiang, Fangyue Liu, Shufan Liu, Xuanzhe Liu, and Xin Jin. 2026. RAGCache: Efficient Knowledge Caching for Retrieval-Augmented Generation. *ACM Trans. Comput. Syst.* 44, 1 (2026), 2:1–2:27. doi:10.1145/3768628

[54] Jiajie Jin, Yutao Zhu, Zhicheng Dou, Guanting Dong, Xinyu Yang, Chenghao Zhang, Tong Zhao, Zhao Yang, and Ji-Rong Wen. 2025. FlashRAG: A Modular Toolkit for Efficient Retrieval-Augmented Generation Research. In *Companion Proceedings of the ACM on Web Conference 2025, WWW 2025, Sydney, NSW, Australia*. ACM, 737–740. doi:10.1145/3701716.3715313

[55] Jina AI. 2026. Reader API: Convert any URL to an LLM-friendly input. https://jina.ai/reader/. Accessed: 2026-06-23.

[56] Hailey Joren, Jianyi Zhang, Chun-Sung Ferng, Da-Cheng Juan, Ankur Taly, and Cyrus Rashtchian. 2025. Sufficient Context: A New Lens on Retrieval Augmented Generation Systems. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net. https://openreview.net/forum?id=Jjr2Odj8DJ

[57] Dayeon Kang, Hyejun Jeong, Jade Sheffey, Pubali Datta, and Amir Houmansadr. 2026. Whose Agent Are You? Multi-Layer Fingerprinting and Attribution of Autonomous Web Agents. arXiv:2606.20910

[58] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 6769–6781. doi:10.18653/v1/2020.emnlp-main.550

[59] Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Chong Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov, and Daniel Fried. 2024. VisualWebArena: Evaluating Multimodal Agents on Realistic Visual Web Tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*. Association for Computational Linguistics, 881–905. doi:10.18653/V1/2024.ACL-LONG.50

[60] Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham M. Kakade, Prateek Jain, and Ali Farhadi. 2022. Matryoshka Representation Learning. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (Eds.). http://papers.nips.cc/paper_files/paper/2022/hash/c32319f4868da7613d78af9993100e42-Abstract-Conference.html

[61] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: a Benchmark for Question Answering Research. *Trans. Assoc. Comput. Linguistics* 7 (2019), 452–466. doi:10.1162/TACL_A_00276

[62] Ben Laurie, Eran Messeri, and Rob Stradling. 2021. Certificate Transparency Version 2.0. *RFC* 9162 (2021), 1–53. doi:10.17487/RFC9162

[63] Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Tomas Kocisky, Sebastian Ruder, Dani Yogatama, Kris Cao, Susannah Young, and Phil Blunsom. 2021. Mind the Gap: Assessing Temporal Generalization in Neural Language Models. In *Advances in Neural Information Processing Systems (NeurIPS)*. arXiv:2102.01951

[64] Seungjin Lee, Faraz Ahmed, Diman Zad Tootaghaj, Hardik Soni, Steven Swanson, and Puneet Sharma. 2026. Characterizing Locality in Large-Scale Vector Search Workloads of Agentic AI Systems. In *VLDB 2026 Workshop: The 2nd Workshop on Vector Databases*.

[65] Patrick S. H. 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. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (Eds.). https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html

[66] Jiaxing Li, Chi Xu, Lianchen Jia, Feng Wang, Cong Zhang, and Jiangchuan Liu. 2024. EACO-RAG: Edge-Assisted and Collaborative RAG with Adaptive Knowledge Update. arXiv:2410.20299

[67] Rui Li, Shuang Cao, Ruihua Liu, and Alexandre Duprey. 2026. EviDex: Provenance-Weighted Evidence-Path Indexing for Fresh and Auditable Retrieval under Continuous Updates. *Proc. ACM Meas. Anal. Comput. Syst.* 10, 2 (2026), 38:1–38:23. doi:10.1145/3805636

[68] Shihan Lin, Suting Chen, Yunming Xiao, Yanqi Gu, Aleksandar Kuzmanovic, and Xiaowei Yang. 2025. PreAcher: Secure and Practical Password Pre-Authentication by Content Delivery Networks. In *22nd USENIX Symposium on Networked Systems Design and Implementation (NSDI 25)*. USENIX Association, Philadelphia, PA, 1399–1419.

[69] Adam Liska, Tomás Kociský, Elena Gribovskaya, Tayfun Terzi, Eren Sezener, Devang Agrawal, Cyprien de Masson d'Autume, Tim Scholtes, Manzil Zaheer, Susannah Young, Ellen Gilsenan-McMahon, Sophia Austin, Phil Blunsom, and Angeliki Lazaridou. 2022. StreamingQA: A Benchmark for Adaptation to New Knowledge over Time in Question Answering Models. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA (Proceedings of Machine Learning Research, Vol. 162)*. PMLR, 13604–13622. https://proceedings.mlr.press/v162/liska22a.html

[70] Haotian Liu, Yujun He, and Bo Tang. 2026. Efficient and Effective In-place Graph-based Vector Index Updates. arXiv:2607.15576

[71] Mengjie Liu, Jiahui Peng, Wenchang Ning, Pei Chu, Jiantao Qiu, Ren Ma, He Zhu, Rui Min, Lindong Lu, Linfeng Hou, Kaiwen Liu, Yuan Qu, Zhenxiang Li, Chao Xu, Zhongying Tu, Wentao Zhang, and Conghui He. 2026. Dripper: Token-Efficient Main HTML Extraction with a Lightweight LM. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, KDD 2026, Jeju Island, Korea, August 9-13, 2026*. ACM, 3258–3269. doi:10.1145/3770855.3817915

[72] Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the Middle: How Language Models Use Long Contexts. *Trans. Assoc. Comput. Linguistics* 12 (2024), 157–173. doi:10.1162/TACL_A_00638

[73] Wenhan Liu, Xinyu Ma, Yutao Zhu, Yuchen Li, Daiting Shi, Dawei Yin, and Zhicheng Dou. 2026. Agentic-R: Learning to Retrieve for Agentic Search. arXiv:2601.11888

[74] Yi Liu, Fei Fang, and Chen Qian. 2025. Efficient Vector Search on Disaggregated Memory with d-HNSW. In *Proceedings of the 17th ACM Workshop on Hot Topics in Storage and File Systems, HotStorage 2025, Boston, MA, USA, July 10-11, 2025*. ACM, 1–8. doi:10.1145/3736548.3737822

[75] Michal Lukasik, Boris Dadachev, Kishore Papineni, and Gonçalo Simões. 2020. Text Segmentation by Cross Segment Attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, 4707–4716. doi:10.18653/V1/2020.EMNLP-MAIN.380

[76] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2022. CLIP4Clip: An empirical study of CLIP for end to end video clip retrieval and captioning. *Neurocomputing* 508 (2022), 293–304. doi:10.1016/J.NEUCOM.2022.07.028

[77] Yu A Malkov and Dmitry A Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE transactions on pattern analysis and machine intelligence* 42, 4 (2018), 824–836.

[78] Muhammad Mansoor, Tahir Ahmad, and Yeo-Chan Yoon. 2026. Risk-Constrained Freshness-Aware Semantic Caching for Open-Web Retrieval-Augmented LLMs. arXiv:2607.04281

[79] Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023. MTEB: Massive Text Embedding Benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*. doi:10.18653/v1/2023.eacl-main.148

[80] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. 2021. WebGPT: Browser-assisted question-answering with human feedback. arXiv:2112.09332 [cs.CL]

[81] Youngeun Nam, Joeun Kim, Hwanjun Song, Susik Yoon, Jae-Gil Lee, and Byung Suk Lee. 2026. TimelyRAG: Semantic-Temporal Hybrid Retrieval for Time-Critical Question Answering in Overlapping-Evolving Documents. arXiv:2609.11572 [cs.IR]

[82] Sukjoon Oh, Minki Kang, Dohyun Kim, Baotong Lu, Jing Liu, Qianxi Zhang, Qi Chen, and Youjip Won. 2026. Aker: Density-Aware Approximate Caching for Vector Search (Extended Version).

arXiv:2609.03712 [cs.DB]

[83] OpenAI. 2025. Computer-Using Agent. OpenAI Blog. Accessed: 2026-06-08. https://openai.com/index/computer-using-agent/

[84] OpenAI. 2025. Introducing ChatGPT Agent: Bridging Research and Action. OpenAI Blog. Accessed: 2026-06-08. https://openai.com/index/introducing-chatgpt-agent/

[85] Behzad Ousat, Nikita Turkmen, Lalchandra Rampersaud, Dillan Bailey, and Amin Kharraz. 2026. Broken Gates: Re-evaluating Web Bot Defenses in the Age of LLM Agents. arXiv:2607.18659

[86] Jie Ouyang, Tingyue Pan, Mingyue Cheng, Ruiran Yan, Yucong Luo, Jiaying Lin, and Qi Liu. 2025. HoH: A Dynamic Benchmark for Evaluating the Impact of Outdated Information on Retrieval-Augmented Generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria*. Association for Computational Linguistics, 6036–6063. doi:10.18653/V1/2025.ACL-LONG.301

[87] Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. 2023. Measuring and Narrowing the Compositionality Gap in Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023 (Findings of ACL, Vol. EMNLP 2023)*. Association for Computational Linguistics, 5687–5711. doi:10.18653/V1/2023.FINDINGS-EMNLP.378

[88] Zipeng Qiu, Wenjie Qu, Jiaheng Zhang, and Binhang Yuan. 2026. V3DB: Audit-on-Demand Zero-Knowledge Proofs for Verifiable Vector Search over Committed Snapshots. arXiv:2603.03065 [cs.CR]

[89] Renyi Qu, Ruixuan Tu, and Forrest Sheng Bao. 2025. Is Semantic Chunking Worth the Computational Cost?. In *Findings of the Association for Computational Linguistics: NAACL 2025, Albuquerque, New Mexico, USA, April 29 - May 4, 2025 (Findings of ACL)*, Luis Chiruzzo, Alan Ritter, and Lu Wang (Eds.). Association for Computational Linguistics, 2155–2177. doi:10.18653/V1/2025.FINDINGS-NAACL.114

[90] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML) (Proceedings of Machine Learning Research, Vol. 139)*. PMLR, 8748–8763.

[91] Vivek Ramanujan, Pavan Kumar Anasosalu Vasu, Ali Farhadi, Oncel Tuzel, and Hadi Pouransari. 2022. Forward Compatible Training for Large-Scale Embedding Retrieval Systems. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 19386–19395. doi:10.1109/CVPR52688.2022.01878

[92] Siddhant Ray, Nick Feamster, and Junchen Jiang. 2026. An Internet for the KV Cache: Rethinking Classical Infrastructure Boundaries in the LLM Inference Age. arXiv:2608.01526

[93] Aniket Rege, Aditya Kusupati, Sharan Ranjit S, Alan Fan, Qingqing Cao, Sham M. Kakade, Prateek Jain, and Ali Farhadi. 2023. AdANNS: A Framework for Adaptive Semantic Search. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine (Eds.). http://papers.nips.cc/paper_files/paper/2023/hash/f062da1973ac9ac61fc6d44dd7fa309f-Abstract-Conference.html

[94] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, Hong Kong, China, 3982–3992. doi:10.18653/v1/D19-1410

[95] Korakit Seemakhupt, Sihang Liu, and Samira Manabi Khan. 2024. EdgeRAG: Online-Indexed RAG for Edge Devices. arXiv:2412.21023

[96] Serper. 2026. Serper: The World's Fastest and Cheapest Google Search API. https://serper.dev/. Accessed: 2026-06-23.

[97] Abhinav Sharma, Brian Zhang, Deepti Guntur, Zhiyang Zuo, Shreyas Chaudhari, Wenlong Zhao, Franck Dernoncourt, Puneet Mathur, Ryan Anthony Rossi, and Nedim Lipka. 2026. Test-Time Strategies for More Efficient and Accurate Agentic RAG. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop), ACL 2026, San Diego, California, United States, July 2-7, 2026*. Association for Computational Linguistics, 463–469. doi:10.18653/V1/2026.ACL-SRW.41

[98] Yantao Shen, Yuanjun Xiong, Wei Xia, and Stefano Soatto. 2020. Towards Backward-Compatible Representation Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 6368–6377. doi:10.1109/CVPR42600.2020.00640

[99] Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. REPLUG: Retrieval-Augmented Black-Box Language Models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, Kevin Duh, Helena Gómez-Adorno, and Steven Bethard (Eds.). Association for Computational Linguistics, 8371–8384. doi:10.18653/V1/2024.NAACL-LONG.463

[100] Aparajita Sinha and Kunal Chakma. 2026. Adaptive Query Performance Prediction for Retrieval-Augmented Generation: Bridging Retrieval Quality and Generation Relevance. *ACM Transactions on Information Systems* (2026). doi:10.1145/3827605

[101] Huatong Song, Jinhao Jiang, Wenqing Tian, Zhipeng Chen, Yuhuan Wu, Jiahao Zhao, Yingqian Min, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. 2025. R1-Searcher++: Incentivizing the Dynamic Knowledge Acquisition of LLMs via Reinforcement Learning. arXiv:2505.17005

[102] Aaron Steiner, Ralph Peeters, and Christian Bizer. 2026. MCP vs RAG vs NLWeb vs HTML: A Comparison of the Effectiveness and Efficiency of Different Agent Interfaces to the Web. In *Proceedings of the ACM Web Conference 2026, WWW 2026, Dubai, United Arab Emirates, originally scheduled for April 13-17, 2026, rescheduled for June 29 - July 3, 2026*, Hakim Hacid, Yoelle Maarek, Francesco Bonchi, Ido Guy, and Emine Yilmaz (Eds.). ACM, 8493–8496. doi:10.1145/3774904.3792893

[103] Jiejun Tan, Zhicheng Dou, Wen Wang, Mang Wang, Weipeng Chen, and Ji-Rong Wen. 2025. HtmlRAG: HTML is Better Than Plain Text for Modeling Retrieved Knowledge in RAG Systems. In *Proceedings of the ACM on Web Conference 2025, WWW 2025, Sydney, NSW, Australia, 28 April 2025- 2 May 2025*, Guodong Long, Michale Blumestein, Yi Chang, Liane Lewin-Eytan, Zi Helen Huang, and Elad Yom-Tov (Eds.). ACM, 1733–1746. doi:10.1145/3696410.3714546

[104] Tavily. 2026. Tavily: The Web Access Layer for AI Agents. https://tavily.com/. Accessed: 2026-06-23.

[105] Llama Team. 2024. The Llama 3 Herd of Models. arXiv:2407.21783

[106] Qwen Team. 2025. Qwen3 Technical Report. arXiv:2505.09388

[107] Fangzheng Tian, Jinyuan Fang, Debasis Ganguly, Zaiqiao Meng, and Craig Macdonald. 2025. Am I on the Right Track? What Can Predicted Query Performance Tell Us about the Search Behaviour of Agentic RAG. In *Proceedings of the Workshop on Information Retrieval's Role in RAG Systems (IR-RAG 2025) co-located with the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2025), Padova, Italy, July 17, 2025 (CEUR Workshop Proceedings, Vol. 4199)*, Negar Arabzadeh, Ziheng Chen, Fabio Petroni, Federico Siciliano, Fabrizio Silvestri, and Giovanni Trappolini (Eds.).

CEUR-WS.org, 22–41. https://ceur-ws.org/Vol-4199/paper3.pdf

[108] Fangzheng Tian, Debasis Ganguly, and Craig Macdonald. 2026. Predicting Retrieval Utility and Answer Quality in Retrieval-Augmented Generation. In *Advances in Information Retrieval - 48th European Conference on Information Retrieval, ECIR 2026, Delft, The Netherlands, March 29 - April 2, 2026, Proceedings, Part I (Lecture Notes in Computer Science, Vol. 16483)*, Ricardo Campos, Adam Jatowt, Yanyan Lan, Mohammad Aliannejadi, Christine Bauer, Sean MacAvaney, Avishek Anand, Zhaochun Ren, Suzan Verberne, Nan Bai, and Masoud Mansoury (Eds.). Springer, 368–385. doi:10.1007/978-3-032-21289-4_24

[109] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. MuSiQue: Multihop Questions via Single-hop Question Composition. *Trans. Assoc. Comput. Linguistics* 10 (2022), 539–554. doi:10.1162/TACL_A_00475

[110] UncleCode. 2026. Crawl4AI: Open-Source LLM-Friendly Web Crawler & Scraper. https://github.com/unclecode/crawl4ai. Accessed: 2026-10-18.

[111] Harshil Vejendla. 2025. Drift-Adapter: A Practical Approach to Near Zero-Downtime Embedding Model Upgrades in Vector Databases. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, 15938–15949. doi:10.18653/v1/2025.emnlp-main.805

[112] Maria Vlachou and Craig Macdonald. 2024. Coherence-based Query Performance Measures for Dense Retrieval. In *Proceedings of the 2024 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2024, Washington, DC, USA, 13 July 2024*, Harrie Oosterhuis, Hannah Bast, and Chenyan Xiong (Eds.). ACM, 15–24. doi:10.1145/3664190.3672518

[113] Tu Vu, Mohit Iyyer, Xuezhi Wang, Noah Constant, Jerry W. Wei, Jason Wei, Chris Tar, Yun-Hsuan Sung, Denny Zhou, Quoc V. Le, and Thang Luong. 2024. FreshLLMs: Refreshing Large Language Models with Search Engine Augmentation. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024 (Findings of ACL, Vol. ACL 2024)*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, 13697–13720. doi:10.18653/V1/2024.FINDINGS-ACL.813

[114] Feng Wang, Zesheng Shi, Bo Wang, Nan Wang, and Han Xiao. 2025. ReaderLM-v2: Small Language Model for HTML to Markdown and JSON. arXiv:2503.01151

[115] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text Embeddings by Weakly-Supervised Contrastive Pre-training. arXiv:2212.03533

[116] Yichuan Wang, Zhifei Li, Zirui Wang, Paul Teiletche, Lesheng Jin, Matei Zaharia, Joseph E. Gonzalez, and Sewon Min. 2026. PIXEL-RAG: Web Screenshots Beat Text for Retrieval-Augmented Generation. arXiv:2606.28344

[117] Yichuan Wang, Shu Liu, Zhifei Li, Yongji Wu, Ziming Mao, Yilong Zhao, Xiao Yan, Zhiying Xu, Yang Zhou, Ion Stoica, Sewon Min, Matei Zaharia, and Joseph E. Gonzalez. 2025. LEANN: A Low-Storage Vector Index. arXiv:2506.08276 [cs.DB]

[118] Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford, Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese. 2025. BrowseComp: A Simple Yet Challenging Benchmark for Browsing Agents. arXiv:2504.12516 [cs.CL]

[119] Orion Weller, Michael Boratko, Iftekhar Naim, and Jinhyuk Lee. 2025. On the Theoretical Limitations of Embedding-Based Retrieval. arXiv:2508.21038

[120] Jialong Wu, Wenbiao Yin, Yong Jiang, Zhenglin Wang, Zekun Xi, Runnan Fang, Linhai Zhang, Yulan He, Deyu Zhou, Pengjun Xie, and Fei Huang. 2025. WebWalker: Benchmarking LLMs in Web Traversal. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (Eds.). Association for Computational Linguistics, 10290–10305. https://aclanthology.org/2025.acl-long.508/

[121] Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Marianna Nezhurina, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. 2023. Large-scale Contrastive Language-Audio Pretraining with Feature Fusion and Keyword-to-Caption Augmentation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 1–5.

[122] Yunming Xiao, Yanqi Gu, Yibo Zhao, Sen Lin, and Aleksandar Kuzmanovic. 2025. Enabling Anonymous Online Streaming Analytics at the Network Edge. *ACM Trans. Comput. Syst.* 43, 4 (2025), 13:1–13:39. doi:10.1145/3746130

[123] Yunming Xiao, Yibo Zhao, Sen Lin, and Aleksandar Kuzmanovic. 2024. Snatch: Online Streaming Analytics at the Network Edge. In *Proceedings of the Nineteenth European Conference on Computer Systems, EuroSys 2024, Athens, Greece, April 22-25, 2024*. ACM, 349–369. doi:10.1145/3627703.3629577

[124] Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective Retrieval Augmented Generation. arXiv:2401.15884

[125] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 Technical Report. arXiv:2412.15115

[126] 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. 2024. CRAG – Comprehensive RAG Benchmark. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*. arXiv:2406.04744

[127] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. arXiv:1809.09600

[128] Woongyeong Yeo, Kangsan Kim, Soyeong Jeong, Jinheon Baek, and Sung Ju Hwang. 2026. UniversalRAG: Retrieval-Augmented Generation over Corpora of Diverse Modalities and Granularities. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States*. Association for Computational Linguistics, 3843–3871. doi:10.18653/V1/2026.ACL-LONG.177

[129] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2021. Jointly Optimizing Query Encoder and Product Quantization to Improve Retrieval Performance. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*. Association for Computing Machinery, 2487–2496. doi:10.1145/3459637.3482358

[130] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2022. Learning Discrete Representations via Constrained Clustering for Effective and Efficient Dense Retrieval. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*. Association for Computing Machinery, 1328–1336.

doi:10.1145/3488560.3498443

[131] Jintao Zhang, Guoliang Li, and Jinyang Su. 2025. SAGE: A Framework of Precise Retrieval for RAG. In *41st IEEE International Conference on Data Engineering, ICDE 2025, Hong Kong, May 19-23, 2025*. IEEE, 1388–1401. doi:10.1109/ICDE65448.2025.00108

[132] Yazhuo Zhang, Jinqing Cai, Avani Wildani, and Ana Klimovic. 2025. Rethinking Web Cache Design for the AI Era. In *Proceedings of the 2025 ACM Symposium on Cloud Computing, SoCC 2025, Online, USA, November 19-21, 2025*. ACM, 535–542. doi:10.1145/3772052.3772255

[133] Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. arXiv:2506.05176

[134] Jihao Zhao, Zhiyuan Ji, Pengnian Qi, Simin Niu, Bo Tang, Feiyu Xiong, and Zhiyu Li. 2024. Meta-Chunking: Learning Efficient Text Segmentation via Logical Perception. arXiv:2410.12788

[135] Tong Zhao, Yutao Zhu, Yucheng Tian, and Zhicheng Dou. 2026. R^3AG: Retriever Routing for Retrieval-Augmented Generation. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2026, San Diego, California, United States*. Association for Computational Linguistics, 20506–20522. doi:10.18653/V1/2026.ACL-LONG.939

[136] Zijie Zhong, Hanwen Liu, Xiaoya Cui, Xiaofan Zhang, and Zengchang Qin. 2025. Mix-of-Granularity: Optimize the Chunking Granularity for Retrieval-Augmented Generation. In *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025, Abu Dhabi, UAE*. Association for Computational Linguistics, 5756–5774. https://aclanthology.org/2025.coling-main.384/

[137] Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. 2025. PoisonedRAG: Knowledge Corruption Attacks to Retrieval-Augmented Generation of Large Language Models. In *34th USENIX Security Symposium, USENIX Security 2025, Seattle, WA, USA, August 13-15, 2025*, Lujo Bauer and Giancarlo Pellegrino (Eds.). USENIX Association, 3827–3844. https://www.usenix.org/conference/usenixsecurity25/presentation/zou-poisonedrag