

Hierarchical Document Retrieval for Efficient Question Answering with Large Language Models

Roza Armata

Department of Computer Science, University of New Hampshire, Durham, NH, USA.
rozaarmata94@unh.edu

Marcio Vives

Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, USA.
marcio1985@ku.edu

Oonagh Nolt

Department of Computer Science, University of Central Florida, Orlando, FL, USA.
oonagh.nolt@ucf.edu

Abstract

Large language models have become central to question answering systems, yet their effectiveness depends heavily on the quality, timeliness, and organization of external knowledge. Hierarchical document retrieval offers a systems-level strategy for reconciling the broad coverage of open-domain search with the precision required for grounded generation. This paper examines hierarchical document retrieval as an infrastructure for efficient question answering with large language models. It analyzes layered retrieval architectures that separate corpus stratification, query routing, coarse-grained candidate generation, fine-grained reranking, context assembly, and response synthesis. The discussion emphasizes structural trade-offs involving latency, cost, recall, precision, provenance, and governance rather than isolated algorithmic improvements. It argues that hierarchical retrieval is not merely a technical optimization but a socio-technical arrangement that shapes access to institutional knowledge, fairness across domains, and the sustainability of deployed systems. Through cross-domain comparisons in enterprise, legal, biomedical, and public-sector settings, the paper identifies recurring tensions between centralized control and local autonomy, between freshness and stability, and between retrieval breadth and generative restraint. It concludes that robust question answering systems require hierarchical retrieval designs that are auditable, energy-aware, policy-aligned, and resilient to heterogeneous source conditions.

Keywords

hierarchical retrieval; document retrieval; question answering; large language models; retrieval-augmented generation; systems architecture; governance; sustainability.

1. Introduction

Question answering with large language models has moved from closed-book generation toward retrieval-augmented architectures that condition responses on external documents. Early work demonstrated that retrieval can supply factual evidence for knowledge-intensive tasks, reducing reliance on parametric memory alone [1]. Subsequent dense retrieval methods showed that learned representations can outperform lexical matching in open-domain settings [2], while generative reading and fusion approaches illustrated how retrieved passages can be

integrated into answer synthesis [3]. These developments established retrieval as an essential layer in modern question answering, but they also exposed a scaling problem: as corpora grow in size, heterogeneity, and regulatory sensitivity, flat retrieval pipelines become inefficient and difficult to govern.

A flat retrieval architecture typically treats all documents as members of a single searchable space. This simplicity is attractive for prototyping, but it creates several system-level liabilities. First, broad indexes must serve queries of varying specificity, causing computational waste when a narrow domain query is compared against an entire enterprise corpus. Second, heterogeneous sources such as policies, technical manuals, clinical notes, legal opinions, and customer records have different update frequencies, access controls, and quality profiles. Treating them uniformly undermines both precision and compliance. Third, large language models can be sensitive to context ordering and redundancy, so indiscriminate passage inclusion may degrade answer quality even when relevant material is retrieved [4]. Hierarchical document retrieval responds to these pressures by organizing retrieval into multiple levels of granularity, authority, and domain scope.

The central claim of this paper is that hierarchical document retrieval should be understood as an infrastructure problem rather than a single-model problem. The design of retrieval hierarchies affects latency, operational cost, energy consumption, fairness, auditability, and the ability of institutions to maintain control over knowledge assets. Retrieval-augmented generation has already been extended to massive token collections and trillion-scale retrieval [5], and surveys have mapped the growing design space [6]. Self-reflective retrieval methods further indicate that systems can learn when to retrieve and how to critique evidence [7]. Yet the organizational and policy implications of hierarchical retrieval remain underdeveloped. This paper therefore examines architectural layers, deployment trade-offs, governance mechanisms, and cross-domain cases to articulate a systems agenda for efficient question answering with large language models.

2. Background and Related Work

Information retrieval has long grappled with the tension between recall and precision. Probabilistic relevance frameworks such as BM25 remain influential because they offer transparent, efficient, and robust lexical matching [8]. Classical retrieval theory emphasizes indexing, query representation, ranking, and evaluation as interconnected concerns [9]. Text classification and representation learning later expanded the feature space available to retrieval systems, enabling richer document characterization [10]. Dense passage retrieval shifted the emphasis from term overlap to semantic similarity, making it possible to retrieve passages that share meaning without sharing vocabulary [2]. Sentence-level embedding models further improved the ability to compare queries and passages in a shared vector space [11]. These advances are foundational, but they also increase infrastructure complexity because dense indexes require substantial memory, periodic retraining, and careful embedding version management.

Benchmarking efforts such as BEIR revealed that zero-shot retrieval performance varies substantially across domains and tasks [12]. This finding is important for hierarchical design because it suggests that no single retrieval model or index configuration is universally optimal. A system that performs well on web-style questions may perform poorly on legal or biomedical material. Language models as knowledge bases can encode factual associations, but they are unreliable for provenance and updating [13]. Consequently, retrieval layers are needed not only to improve accuracy but also to provide traceability and refreshability. At the

same time, the ethical and social risks of large language models include bias, misinformation, privacy leakage, and environmental costs [14], [15]. Hierarchical retrieval can mitigate some of these risks by segregating sensitive collections, enforcing access controls, and attaching source metadata to generated answers. However, it can also create new risks if hierarchy design reflects institutional power asymmetries or if lower-level caches bypass governance checks.

Evaluation traditions in information retrieval emphasize test collections, relevance judgments, and reproducible metrics [16]. Summarization research similarly shows that selecting and compressing source material is not neutral; it involves judgments about salience and omission [17]. In question answering, these judgments become policy decisions because the system decides which documents are visible and which are excluded. Transformer architectures underpin many contemporary retrieval and generation components through attention mechanisms that model long-range dependencies [19]. A recent framework for multi-source heterogeneous retrieval demonstrates that unified semantic spaces and attention-based fusion can improve alignment across diverse sources [18]. This work is relevant because it illustrates the technical feasibility of integrating heterogeneous retrieval signals, though it also raises governance questions about how source authority and access rights are represented. Foundation model research has shown that scale amplifies both capability and risk [20], making hierarchical retrieval a potential control surface for grounding, containment, and accountability.

3. Conceptual Architecture of Hierarchical Document Retrieval

A hierarchical document retrieval system can be understood as a layered pipeline that progressively narrows the search space while preserving opportunities for evidence recovery. The first layer is corpus stratification, in which documents are partitioned by domain, sensitivity, update frequency, language, authority, or ownership. This layer determines which collections are eligible for a given query and which are excluded by policy. The second layer is query routing, where the system classifies intent, detects domain cues, and dispatches the request to one or more sub-indexes. Routing may rely on lexical signals, learned classifiers, or metadata about the user and application context. The third layer is coarse retrieval, which retrieves a broad set of candidate documents or passages from selected sub-indexes. This stage prioritizes recall because later stages can filter aggressively. The fourth layer is fine-grained reranking, which uses more expensive models to score candidates against the query and against one another. The fifth layer is context assembly, where passages are selected, ordered, deduplicated, and compressed to fit the generative model's context window. The final layer is response synthesis, where the language model produces an answer with citations, caveats, and refusal conditions when evidence is insufficient.

This layered design creates multiple opportunities for efficiency. By routing queries to smaller sub-indexes, the system avoids comparing every query against the entire corpus. By using coarse retrieval before reranking, it limits the number of expensive semantic comparisons. By assembling context from a curated hierarchy, it can reduce redundant or low-value text in the prompt, which lowers inference cost and may improve answer focus. Dense retrieval methods provide the semantic foundation for several of these layers [2], while massive retrieval-augmented models show that scaling can be managed through retrieval rather than pure parameter growth [5]. Sentence embeddings support compact passage representations suitable for coarse and fine ranking [11]. Attention-based fusion can combine signals from multiple sources, as demonstrated in recent heterogeneous retrieval work [18].

The overall architecture therefore resembles a distributed information system more than a single search function.

The hierarchy also has important structural trade-offs. Deeper hierarchies can improve precision and policy compliance, but they introduce routing errors, coordination overhead, and index synchronization challenges. If a query is routed to the wrong domain, relevant evidence may never enter the candidate set. If sub-indexes update at different rates, the system may provide inconsistent answers across domains. If context assembly favors higher-authority sources, it may suppress valuable local knowledge. These tensions suggest that hierarchy design must be treated as an ongoing governance process. The system should expose routing decisions, source eligibility rules, and confidence estimates so that operators and auditors can diagnose failures. In this sense, hierarchical retrieval is not merely a method for finding documents; it is an institutional mechanism for deciding which knowledge counts, which sources are current, and which answers can be responsibly generated.

4. Efficiency, Latency, and Infrastructure Trade-Offs

Efficiency in hierarchical retrieval is multidimensional. The most visible dimension is latency, which includes query understanding, routing, index lookup, reranking, context assembly, and generation. A system may reduce retrieval latency by using smaller sub-indexes, approximate nearest-neighbor search, or caching. However, these optimizations can reduce recall if the hierarchy is poorly tuned. Lexical retrieval remains valuable for low-latency and high-precision matching in specialized vocabularies [8]. Dense retrieval can improve semantic recall but often requires more memory and computation [2]. Reranking with cross-attention or large embedding models can improve precision but adds significant cost. Context assembly must balance the number of passages against the language model's prompt capacity and the risk of distraction. The system architect must therefore decide where to spend computational budget and where to accept bounded uncertainty.

A second dimension is operational cost. Hierarchical systems require index construction, embedding generation, storage, replication, monitoring, and periodic re-embedding when models change. These costs are unevenly distributed. High-authority or high-sensitivity collections may require dedicated infrastructure, audit logs, and access controls, while low-risk public collections can use more elastic resources. Caching can reduce repeated retrieval, but caches must respect access rights and data freshness. If a cached answer contains information that later becomes restricted or outdated, the hierarchy has failed its governance objectives. Evaluation frameworks developed for information retrieval emphasize reproducible measurement [16], and hierarchical systems should extend such measurement to routing accuracy, source coverage, freshness, and policy compliance. Without these additional metrics, efficiency gains may obscure systemic degradation.

A third dimension is sustainability. Large language models and dense retrieval systems consume energy during training, indexing, and inference [20]. Hierarchical retrieval can reduce energy use by narrowing candidate sets and shortening prompts, but it can also increase energy use by maintaining many specialized indexes and rerankers. The net effect depends on query volume, cache hit rates, source update frequency, and hardware utilization. Green design therefore requires lifecycle analysis rather than isolated benchmarks. Systems should consider whether a smaller model with hierarchical retrieval can replace a larger monolithic model, whether embeddings can be reused across applications, and whether cold collections can be indexed on demand. Recent heterogeneous retrieval frameworks point toward better fusion across sources [18], but their sustainability depends on deployment

choices. Efficiency is thus not a purely technical metric; it is a socio-technical outcome shaped by institutional priorities and infrastructure governance.

5. Robustness, Fairness, and Governance

Robustness in hierarchical retrieval concerns the system's ability to maintain acceptable performance under distribution shift, adversarial inputs, source outages, and corrupted documents. Because queries are routed across sub-indexes, a single misclassification can exclude the correct domain. Because context is assembled from multiple sources, conflicting evidence can confuse generation. Because language models can hallucinate, retrieval systems must support refusal and uncertainty expression. Self-reflective retrieval approaches attempt to critique retrieved evidence and decide when more retrieval is needed [7]. However, reflection alone does not resolve governance questions about which sources are authoritative. A robust system needs redundancy across retrieval paths, fallback strategies when a sub-index is unavailable, and anomaly detection for unusual routing patterns.

Fairness is equally important. Hierarchical retrieval can improve fairness by ensuring that underrepresented domains are indexed and routed appropriately. It can also worsen fairness if hierarchy design privileges dominant languages, institutions, or formats. If a legal aid system retrieves only formal statutes and ignores community legal education materials, it may disadvantage users who need practical guidance. If a biomedical system prioritizes English-language journals over local clinical guidelines, it may produce answers that are less relevant to regional practice. Benchmark studies show that retrieval models often perform unevenly across domains [12]. Ethical analyses of language models warn that bias and exclusion can be amplified at scale [14], [15]. Hierarchical retrieval must therefore include source diversity audits, coverage reporting, and mechanisms for affected communities to contest missing or misrepresented material.

Governance mechanisms should make the hierarchy auditable. Each document or passage should carry provenance, access level, update timestamp, jurisdiction, and authority metadata. Routing rules should be documented and versioned. Context assembly should log which sources were considered and which were excluded. Generated answers should include citations and confidence indicators. These requirements are not merely compliance overhead; they are functional necessities for debugging and trust. Foundation model research emphasizes the broad societal risks of large-scale systems [20], and retrieval layers can serve as control points for containment and accountability. At the same time, governance must avoid creating rigid hierarchies that freeze knowledge in outdated forms. A good system balances stability with contestability, allowing corrections, appeals, and source updates to propagate through the hierarchy without requiring complete re-architecture.

6. Deployment, Sustainability, and Policy Implications

Deployment of hierarchical retrieval can take several forms. Centralized deployments offer consistent policy enforcement and easier auditing but may create bottlenecks, single points of failure, and data concentration risks. Federated deployments distribute indexes across institutions or regions, improving data sovereignty and resilience but complicating ranking, provenance, and latency. Hybrid deployments use central coordination for routing and policy while leaving sensitive collections local. Each model has different implications for cloud cost, edge computation, and legal compliance. Retrieval-augmented pretraining showed that models can benefit from large external memory [4], while trillion-token retrieval demonstrated the feasibility of very large-scale retrieval [5]. However, feasibility at research

scale does not automatically translate into responsible deployment. Policy frameworks must address data residency, cross-border transfer, retention, and the right to explanation.

Sustainability is closely tied to deployment topology. Centralized systems can achieve economies of scale and high hardware utilization, which may reduce energy per query. Federated systems may duplicate indexes and increase network traffic, but they can avoid moving sensitive data and reduce unnecessary replication. On-device retrieval may reduce network energy but increase local computation and complicate updates. The choice depends on workload characteristics and regulatory constraints. Classical retrieval theory reminds us that indexing and query processing are resource-intensive activities [9]. Language models as knowledge bases can reduce retrieval needs for some stable facts, but they are poor at provenance and updating [13]. Summarization techniques can compress context and reduce prompt length [17], but compression may omit qualifying details. Hierarchical retrieval therefore requires a lifecycle perspective that accounts for data ingestion, embedding refresh, index maintenance, query serving, and eventual decommissioning.

Policy implications extend beyond energy and cost. If hierarchical retrieval becomes a gatekeeper for institutional knowledge, then its design determines who can ask what and receive reliable answers. Public-sector deployments may require transparency about source selection, especially in benefits, immigration, health, and education. Private-sector deployments may treat the hierarchy as proprietary, raising concerns about lock-in and accountability. A balanced policy approach would require minimum standards for citation, provenance, access logging, and error correction. It would also encourage interoperability so that institutions can migrate between vendors without losing governance metadata. The hierarchy should not be treated as a black box; it should be documented as an information policy instrument with measurable obligations.

7. Evaluation and Cross-Domain Considerations

Evaluating hierarchical retrieval requires metrics beyond answer accuracy. Retrieval evaluation traditionally measures recall, precision, mean average precision, and normalized discounted cumulative gain using curated test collections [16]. Question answering evaluation adds answer exactness, fluency, citation quality, and refusal calibration. Hierarchical systems need additional metrics for routing accuracy, source coverage, freshness, policy compliance, and context efficiency. A system may achieve high answer accuracy while consistently excluding a marginalized source, or it may reduce latency while increasing hallucination due to over-compression. Evaluation should therefore combine automatic measures with expert review, user studies, and adversarial testing. Benchmark suites such as BEIR provide a starting point for cross-domain retrieval assessment [12], but they do not capture institutional access rules or update dynamics.

Cross-domain comparisons reveal recurring patterns. In enterprise settings, hierarchical retrieval often separates public marketing material, internal policies, technical documentation, and customer records. The hierarchy must respect role-based access while allowing cross-departmental questions. In legal settings, the hierarchy may separate statutes, regulations, case law, and secondary commentary. Authority and jurisdiction become central ranking features. In biomedical settings, the hierarchy may separate clinical guidelines, systematic reviews, trial registries, and patient records. Freshness and evidence quality are critical, and errors can have direct safety consequences. In public-sector settings, the hierarchy may separate official notices, service descriptions, and community-generated information. Fairness requires

attention to language access and digital literacy. These domains share a need for provenance, but they differ in tolerance for latency, cost, and uncertainty.

Methodologically, hierarchical retrieval can benefit from combining lexical and dense approaches. Lexical retrieval is efficient and transparent for exact terminology [8], while dense retrieval handles paraphrase and semantic similarity [2]. Reranking can improve precision, and generative fusion can synthesize evidence across passages [3]. Self-reflection can help systems decide when evidence is insufficient [7]. Recent work on multi-source heterogeneous retrieval suggests that unified semantic spaces and attention-based fusion can improve integration across diverse source types [18]. However, evaluation must test whether these methods preserve policy constraints and source authority. A system that fuses sources without respecting access rights is not robust; it is merely powerful. Cross-domain evaluation should therefore include governance scenarios, such as conflicting sources, restricted documents, stale guidelines, and adversarial queries.

8. Case Illustrations and Forward-Looking Perspectives

Consider a large university deploying a question answering assistant for students, faculty, and staff. A flat index would mix course catalogs, financial aid policies, research compliance rules, and internal memoranda. A hierarchical design could route student queries to academic and financial aid sub-indexes, faculty queries to research and grant sub-indexes, and staff queries to human resources and facilities sub-indexes. Access control would filter results before generation. Caching would be limited to public information, while sensitive answers would require real-time retrieval. Evaluation would track routing errors, unanswered queries, and source staleness. This case illustrates how hierarchy improves efficiency and compliance simultaneously, but it also shows the need for clear escalation paths when users dispute answers or request human review.

Consider a hospital system using retrieval-augmented question answering for clinical decision support. A hierarchy might separate drug formularies, clinical guidelines, institutional protocols, and patient-specific records. The system must prioritize current guidelines and flag conflicts. If a guideline is updated, the hierarchy must propagate the change to all relevant caches and indexes. If a patient record is accessed, audit logs must record the query and response. A dense retrieval model may improve semantic matching [2], but lexical retrieval remains important for drug names and codes [8]. Reranking must be calibrated to avoid over-weighting recently indexed but low-authority sources. Text classification can help tag documents by topic and sensitivity [10]. The system's safety depends less on any single model than on the disciplined interaction of hierarchy, metadata, access control, and human oversight.

Looking forward, hierarchical retrieval is likely to become more adaptive. Systems may learn routing policies from user feedback, adjust hierarchy depth based on query complexity, and use attention-based fusion to combine evidence from structured databases, unstructured documents, and multimodal sources [18]. They may also use smaller specialized models for routing and reranking, reserving large language models for final synthesis. This division of labor could improve efficiency and sustainability, but it increases the number of components that must be governed. Interoperability standards for provenance, access, and source authority will be essential. Evaluation must become continuous rather than episodic, with monitoring for drift, bias, and policy violations. The most successful systems will not be those with the largest models, but those that organize knowledge hierarchies in ways that are transparent, contestable, and aligned with institutional purpose.

9. Conclusion

Hierarchical document retrieval offers a practical and conceptually rich foundation for efficient question answering with large language models. By stratifying corpora, routing queries, retrieving candidates at multiple granularities, reranking evidence, assembling context, and synthesizing cited responses, hierarchical systems can reduce computational waste while improving precision and governance. The approach is not a simple replacement for flat retrieval; it is an architectural commitment that reshapes how knowledge is organized, accessed, updated, and audited. The analysis in this paper emphasizes that efficiency, robustness, fairness, and sustainability are intertwined. Latency improvements that bypass access controls are not true gains. Recall improvements that ignore source authority may increase risk. Cost reductions that shift energy consumption elsewhere may not be sustainable. A well-designed hierarchy must therefore be evaluated as a socio-technical infrastructure, not merely as a search pipeline.

Cross-domain cases in enterprise, legal, biomedical, and public-sector settings show that hierarchical retrieval must accommodate different update rates, authority structures, and regulatory requirements. The technical literature provides strong foundations in lexical and dense retrieval, reranking, fusion, and self-reflection [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20]. The next challenge is to integrate these methods into deployable systems with clear governance, interoperable metadata, and continuous evaluation. Future research should develop metrics for routing fairness, source coverage, policy compliance, and lifecycle sustainability. It should also study how institutions negotiate hierarchy design when multiple stakeholders have conflicting knowledge claims. If these challenges are addressed, hierarchical retrieval can help large language models move from impressive but opaque answer generation toward accountable, efficient, and institutionally grounded question answering.

References

1. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems* (Vol. 33, pp. 9459–9474).
2. Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., & Yih, W.-t. (2020). Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing* (pp. 6769–6781). Association for Computational Linguistics.
3. Izacard, G., & Grave, E. (2021). Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume* (pp. 874–880). Association for Computational Linguistics.
4. Guu, K., Lee, K., Tung, Z., Pasupat, P., & Chang, M.-W. (2020). Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning* (Vol. 119, pp. 3929–3938). PMLR.
5. Borgeaud, S., Mensch, A., Hoffmann, J., Cai, T., Rutherford, E., Millican, K., van den Driessche, G., Lespiau, J.-B., Damoc, B., Clark, A., de Las Casas, D., Guy, A., Menick, J., Ring, R., Driessche, G., et al. (2022). Improving language models by retrieving from

trillions of tokens. In Proceedings of the 39th International Conference on Machine Learning (Vol. 162, pp. 2206–2240). PMLR.

6. Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., & Wang, H. (2023). Retrieval-augmented generation for large language models: A survey. arXiv. <https://arxiv.org/abs/2312.10997>
7. Asai, A., Wu, Z., Wang, Y., Sil, A., & Hajishirzi, H. (2023). Self-RAG: Learning to retrieve, generate, and critique through self-reflection. arXiv. <https://arxiv.org/abs/2310.11511>
8. Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4), 333–389.
9. Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press.
10. Kowsari, K., Jafari Meimandi, K., Heidarysafa, M., Mendu, S., Barnes, L., & Brown, D. (2019). Text classification algorithms: A survey. *Information*, 10(4), 150.
11. Reimers, N., & Gurevych, I. (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* (pp. 3982–3992). Association for Computational Linguistics.
12. Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., & Gurevych, I. (2021). BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks* (Vol. 1).
13. Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., & Miller, A. (2019). Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing* (pp. 2463–2473). Association for Computational Linguistics.
14. Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency* (pp. 610–623). ACM.
15. Weidinger, L., Mellor, J., Rauh, M., Griffin, C., Uesato, J., Huang, P.-S., Cheng, M., Glaese, A., Balle, B., Kasirzadeh, A., Kenton, Z., Brown, S., Hawkins, W., Stepleton, T., Biles, C., Birhane, A., Haas, J., Rimell, L., Hendricks, L. A., Isaac, W., Gilbert, M., Legassick, S., Irving, G., & Gabriel, I. (2021). Ethical and social risks of harm from language models. arXiv. <https://arxiv.org/abs/2112.04359>
16. Sanderson, M. (2010). Test collection based evaluation of information retrieval systems. *Foundations and Trends in Information Retrieval*, 4(4), 247–375.
17. Nenkova, A., & McKeown, K. (2012). A survey of text summarization techniques. In *Mining text data* (pp. 43–76). Springer.
18. Deng, Q., Liu, S., Qiu, H., Long, N., Lin, W., & Chen, S. (2026, April). Unified Semantic Space and Attention-Based Fusion Framework for Multi-Source Heterogeneous

Retrieval-Augmented Generation. In 2026 6th International Conference on Artificial Intelligence and Industrial Technology Applications (AIITA) (pp. 591-597). IEEE.

19. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In Advances in Neural Information Processing Systems (Vol. 30, pp. 5998–6008).
20. Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, J., Creel, K., Davis, J. Q., Demszky, D., ... Liang, P. (2021). On the opportunities and risks of foundation models. arXiv. <https://arxiv.org/abs/2108.07258>