

Efficient Retrieval-Augmented Generation via Query-Adaptive Reranking and Lightweight Large Language Model Distillation

Luis Miguel Mayol

School of Electrical Engineering and Computer Science, Oregon State University, Corvallis,
OR, USA.

luismiguelwork@oregonstate.edu

Pirkko Sainio

Department of Computer Science, George Mason University, Fairfax, VA, USA.

pirkkowork@gmu.edu

Irm Klotz

Department of Computer Science, University of Alabama at Birmingham, Birmingham, AL,
USA.

irmiklotz511@uab.edu

Aisling MacVeagh

Department of Electrical Engineering and Computer Science, University of Missouri,
Columbia, MO, USA.

macveaghaisling@missouri.edu

Abstract

Retrieval-augmented generation has become a central architecture for knowledge-intensive language applications because it combines parametric language models with external evidence. Yet its practical value depends less on any single model component than on the system-level orchestration of retrieval, reranking, generation, latency control, and governance. This paper examines an efficient retrieval-augmented generation architecture built on query-adaptive reranking and lightweight large language model distillation. The central argument is that reranking should not be treated as a fixed and expensive stage applied uniformly to all queries. Instead, queries can be routed through adaptive policies that estimate ambiguity, domain specificity, temporal sensitivity, and risk before selecting the depth and type of reranking. Such adaptation improves relevance while controlling computational cost, especially when paired with distilled language models that preserve sufficient reasoning and instruction-following capacity for generation and judgment. The paper discusses architecture, infrastructure, deployment, robustness, fairness, sustainability, and policy implications. It also considers how distillation can reduce energy consumption and latency but may narrow model diversity and introduce artifacts inherited from teacher models. The analysis concludes that efficient retrieval-augmented generation is best understood as a socio-technical infrastructure requiring continuous evaluation, transparent data governance, and adaptive resource allocation rather than static optimization.

Keywords

retrieval-augmented generation; query-adaptive reranking; knowledge distillation; lightweight language models; information retrieval; system architecture; governance; sustainability.

1. Introduction

Retrieval-augmented generation has emerged as a dominant strategy for grounding language model outputs in external knowledge, reducing hallucination, and enabling updates without retraining the entire model. Early formulations combined dense retrieval with sequence-to-sequence generation to improve knowledge-intensive tasks [1], while dense passage retrieval established the effectiveness of learned representations for open-domain question answering [2]. Subsequent work showed that generative models can benefit from retrieved passages [3], and large-scale retrieval from trillions of tokens demonstrated that external memory can substantially expand the knowledge available to a language model [4]. Retrieval-augmented pre-training further integrated evidence access into the learning process itself [5]. These developments have made retrieval-augmented generation a general systems pattern rather than a narrow question-answering technique.

Despite these advances, deploying retrieval-augmented generation at scale exposes a persistent tension between quality and efficiency. A standard pipeline retrieves a broad candidate set, reranks passages with a more expensive model, and then passes the highest-scoring evidence to a generator. This approach can improve relevance, but it also increases latency, energy use, and operational complexity. Reranking is particularly costly because cross-encoder or large language model judges often evaluate many query-document pairs. If every query receives the same expensive treatment, the system wastes computation on simple factual requests while potentially under-serving ambiguous or high-stakes queries. The result is a mismatch between resource allocation and actual informational need.

This paper argues for query-adaptive reranking as a structural response to that mismatch. Rather than applying a uniform reranking policy, the system can estimate query characteristics and select an appropriate retrieval and reranking path. A simple lookup may require only lexical retrieval and a lightweight filter, while a comparative legal question may require multi-stage evidence evaluation. This adaptive logic aligns system behavior with query difficulty, risk, and domain context. It also creates new opportunities for lightweight large language model distillation, because distilled models can serve as efficient rerankers, query classifiers, or generators in resource-constrained settings. Distillation has long been used to compress neural networks [15], and transformer-specific methods have demonstrated that smaller models can retain much of the performance of larger teachers [16][17][18]. A survey of knowledge distillation provides a broader view of these trade-offs [19].

The contribution of this paper is system-level and interdisciplinary. It does not propose a single algorithm or report a narrow benchmark result. Instead, it analyzes how query-adaptive reranking and distillation interact across architecture, infrastructure, governance, fairness, robustness, and sustainability. The discussion draws on retrieval benchmarks [6], reranking methods [7][8][9], large language model rerankers [10][11], query transformation techniques [12][13], and recent collaborative reranking workflows [14]. It also connects these technical choices to augmented language model surveys [20], self-reflective generation [21], and broader critiques of large-scale language models [22][23][24]. Finally, it considers energy and policy concerns in deep learning [25]. The resulting perspective treats efficient retrieval-augmented generation as a socio-technical infrastructure whose performance is inseparable from its governance and deployment context.

2. Background and Related Work

Retrieval-augmented generation sits at the intersection of information retrieval, natural language processing, and systems engineering. Classical retrieval relied heavily on lexical matching, but dense retrieval transformed the field by mapping queries and documents into shared embedding spaces [2]. Hybrid approaches often combine lexical precision with semantic recall, and benchmarks such as BEIR have highlighted the difficulty of zero-shot generalization across heterogeneous tasks [6]. In practice, retrieval quality is not a stable property of a model alone; it depends on corpus freshness, chunking strategy, index maintenance, metadata, and the distribution of user queries. A system that performs well on one domain may fail in another because the meaning of relevance shifts.

Reranking emerged as a way to improve precision after initial retrieval. BERT-based passage reranking demonstrated that deep language understanding could substantially improve ranking quality [7]. ColBERT introduced late interaction to balance efficiency and effectiveness [8], while ColBERTv2 improved storage and retrieval efficiency through residual compression and denoised supervision [9]. These methods show that reranking is not merely a final sorting step; it is a mechanism for resolving ambiguity, injecting context, and correcting retrieval errors. More recently, large language models have been used as rerankers, with studies examining their zero-shot search capabilities [10] and listwise reranking strategies [11]. Query rewriting and hypothetical document generation further expand the design space by transforming the query before retrieval [12][13].

A particularly relevant development is the use of multi-role collaborative workflows for reranking. Instead of relying on a single judge, such workflows assign different roles to model instances or prompts, enabling filtering, comparison, adjudication, and validation to occur in sequence or in parallel [14]. This approach reflects a broader insight: complex ranking decisions benefit from structured disagreement and staged evidence evaluation. It also raises system-level questions about cost, orchestration, and failure modes. If each role uses a large model, the pipeline becomes prohibitively expensive for interactive applications. If roles are distilled into smaller models, the system may become more deployable but may lose subtle judgment. The design challenge is therefore not simply to choose a better reranker but to assemble a portfolio of rerankers and route queries among them.

The generation side of retrieval-augmented systems has also evolved. Augmented language models can retrieve, reason, and act, but they require careful control over when retrieval is invoked and how evidence is integrated [20]. Self-RAG illustrates how a model can learn to retrieve, generate, and critique its own outputs through self-reflection [21]. These approaches suggest that efficiency depends on knowing when not to retrieve and when not to use the largest available model. Distillation supports that goal by transferring behavior from larger teachers to smaller students [15]. DistilBERT, TinyBERT, and MiniLM show that transformer compression can preserve substantial capability while reducing inference cost [16][17][18]. However, distillation is not neutral. It can amplify teacher biases, reduce diversity, and create brittle behavior outside the training distribution [19]. For retrieval-augmented generation, these risks matter because the distilled model may be responsible not only for fluent generation but also for judging evidence.

3. Query-Adaptive Reranking Architecture

A query-adaptive reranking architecture begins with the recognition that queries are not homogeneous. Some are short, popular, and unambiguous, while others are long, context-

dependent, and high-stakes. A system that applies the same reranking depth to all queries will over-invest in easy cases and under-invest in difficult ones. The first architectural layer is therefore query analysis. This layer can estimate query type, domain, temporal sensitivity, ambiguity, and risk. It can also use historical interaction data to predict whether a query is likely to require multi-pass retrieval. The output is a routing decision, not a final answer. The system then selects a retrieval and reranking path that matches the estimated need.

The second layer is candidate generation. A hybrid retriever can produce a broad candidate set using dense, sparse, or learned sparse methods. The size of this set should also be adaptive. For a simple factual query, fewer candidates may suffice. For a comparative or exploratory query, a larger candidate pool provides more material for later judgment. This stage must be integrated with index freshness and metadata filtering. If the corpus includes documents with different access permissions, the retrieval layer must enforce those permissions before reranking. Otherwise, the reranker may score documents that the user is not allowed to see, creating both security and fairness problems.

The third layer is adaptive reranking. Here, the system can choose among lightweight cross-encoders, distilled language model judges, or multi-role collaborative workflows [14]. A lightweight judge may be sufficient for straightforward relevance, while a multi-role workflow may be reserved for ambiguous or consequential queries. The system can also vary the number of passages considered, the granularity of comparison, and the use of listwise versus pairwise judgment. The key is that reranking becomes a resource allocation problem. The system spends more computation where it is likely to change the final answer or reduce risk. This logic resembles cascading architectures in information retrieval, but it is extended to include language model judgment and governance constraints.

The fourth layer is generation. The generator receives a selected evidence set and produces an answer. If the generator is a distilled model, it may be faster and cheaper, but it may also be less capable of synthesizing conflicting evidence. The architecture can compensate by structuring the prompt, providing evidence summaries, or using a larger model only for final validation. This creates a layered division of labor. Smaller models handle routine queries, while larger models are reserved for complex synthesis. The pipeline can also include citation enforcement, uncertainty disclosure, and refusal policies. These are not merely user-interface features; they are architectural constraints that determine what the system can safely do.

The trade-offs in this architecture are significant. Adaptive routing adds complexity and can introduce classification errors. If a query is misclassified as easy, the system may provide a shallow answer. If it is misclassified as hard, the system may waste resources. Governance requirements can also conflict with efficiency. For example, a high-stakes domain may require exhaustive reranking even for simple queries. The design must therefore make policy explicit. It should define which query classes require which reranking depth, how uncertainty is handled, and how users can appeal or escalate. This is where technical architecture meets institutional design. An efficient system is not necessarily one that minimizes computation; it is one that allocates computation in a way that is defensible, equitable, and aligned with the purpose of the application.

4. Lightweight Large Language Model Distillation in Retrieval-Augmented Generation

Distillation is a natural complement to query-adaptive reranking because it provides smaller models that can serve as routers, rerankers, and generators. The classical distillation framework transfers knowledge from a cumbersome teacher to a smaller student by training

the student to mimic the teacher’s output distribution [15]. In transformer-based systems, DistilBERT, TinyBERT, and MiniLM demonstrated that careful distillation can reduce model size while preserving much of the original performance [16][17][18]. A survey of knowledge distillation further shows that distillation can take many forms, including response-based, feature-based, and relation-based transfer [19]. In retrieval-augmented generation, these methods can be applied at multiple points.

One application is query classification. A distilled model can predict whether a query is simple, ambiguous, comparative, temporal, or risky. This classification determines the reranking path. The advantage is low latency and low cost. The risk is that the classifier may inherit the teacher’s blind spots. If the teacher was trained on a corpus dominated by one language or domain, the distilled classifier may misroute queries from underrepresented groups. This is not only a technical issue but also a fairness issue. A query-adaptive system must be evaluated across domains, languages, and user populations to ensure that routing does not systematically disadvantage some users.

A second application is reranking. A distilled language model can act as a relevance judge, either alone or as part of a multi-role workflow. It can compare passages, assign scores, or generate rationales. Because reranking often requires only relative judgments, a smaller model may be sufficient if it is trained on high-quality preference data. However, distillation can compress subtle distinctions. A teacher model may recognize that a passage is relevant because it addresses a narrow caveat, while a student may focus on surface keywords. This can lead to systematic errors in legal, medical, or technical domains where precision matters. The solution is not necessarily to avoid distillation but to combine it with uncertainty estimation and selective escalation. If the distilled reranker is uncertain, the system can call a larger model.

A third application is generation. Distilled generators can produce fluent answers at lower cost, but they may struggle with long evidence sets, conflicting sources, or multi-step reasoning. Retrieval-augmented generation can mitigate this by providing concise evidence and structured prompts. It can also use a larger model for final validation or for queries that require high-stakes synthesis. This layered approach is consistent with augmented language model surveys that emphasize the importance of controlling retrieval and reasoning [20]. It also aligns with self-reflective methods that allow models to critique their own outputs [21]. In practice, the most efficient system may be a cascade: a distilled generator produces a draft, a lightweight checker verifies citations, and a larger model resolves only the cases that remain uncertain.

Distillation also has infrastructure benefits. Smaller models require less memory, consume less energy, and can run on edge devices or within private environments. This can improve privacy and reduce dependence on centralized cloud services. However, distillation can create a false sense of safety. A smaller model may appear to behave like the teacher in benchmark settings but fail under distribution shift. It may also reproduce harmful stereotypes or factual errors from the teacher. For these reasons, distillation must be accompanied by evaluation, documentation, and monitoring. The goal is not to replace large models entirely but to place them strategically within a system that knows when their capacity is needed.

5. Governance, Robustness, Fairness, and Policy

Efficient retrieval-augmented generation is not only a technical pipeline; it is a governance object. The system retrieves documents, ranks them, and generates text that may influence

decisions. It therefore requires policies for data provenance, access control, auditability, and accountability. If the corpus includes copyrighted material, personal data, or proprietary information, the retrieval layer must respect legal and organizational constraints. If the reranker learns from user interactions, it must avoid reinforcing existing biases. If the generator produces answers without citations, it may undermine trust. Governance must be embedded in the architecture rather than added after deployment.

Robustness is a central concern. Retrieval-augmented systems can fail when the index is stale, when the query is adversarial, or when the reranker is manipulated by keyword stuffing. Query-adaptive routing can reduce some risks by escalating suspicious queries, but it can also create new attack surfaces. An adversary might craft a query that is classified as simple so that it bypasses rigorous reranking. A poisoned document might be retrieved and then promoted by a distilled reranker that lacks the capacity to detect manipulation. Robustness therefore requires adversarial testing, anomaly detection, and continuous index validation. It also requires transparency about system limitations.

Fairness is closely tied to retrieval and reranking. If the corpus underrepresents certain perspectives, the generator will produce answers that reflect that imbalance. If the reranker favors documents from dominant sources, minority viewpoints may be marginalized. Distillation can amplify these effects because the student model may learn a simplified version of the teacher’s preferences. A fair system must measure outcomes across groups, domains, and query types. It must also provide mechanisms for correction, appeal, and redress. This is especially important in public-sector applications, healthcare, education, and legal assistance, where retrieval-augmented generation may shape access to information and services.

Policy implications extend beyond individual systems. Foundation models and their applications raise broad questions about concentration of power, labor, and environmental impact [23]. Taxonomies of language model risks identify harms such as misinformation, discrimination, privacy violations, and malicious use [24]. Critiques of large-scale language models emphasize that scale alone does not guarantee social benefit and may exacerbate existing inequalities [22]. Retrieval-augmented generation can partially address these concerns by grounding outputs in external evidence and enabling updates. However, it does not eliminate them. The system still depends on data, infrastructure, and institutions. Policy should therefore focus on transparency, data rights, model documentation, and accountability. It should also support public interest evaluation and independent auditing.

The governance challenge is compounded by the adaptive nature of the system. If the reranking depth changes based on query characteristics, then the level of scrutiny also changes. This can be efficient, but it can also create inconsistent levels of due process. A user asking a high-stakes question may receive a different quality of evidence depending on how the system classifies the query. Governance must define minimum standards for different risk tiers. It must also ensure that users understand when they are receiving a lightweight answer and when they can request deeper review. In this sense, query-adaptive reranking is not just an optimization technique; it is a policy mechanism. It determines who gets careful attention and who receives a faster, less resource-intensive response.

6. Deployment, Sustainability, and Infrastructure

Deployment context shapes the feasibility of retrieval-augmented generation. Cloud-based systems can scale elastically but may raise privacy, sovereignty, and cost concerns. On-

premises systems offer more control but require hardware and expertise. Edge deployments can reduce latency and improve privacy but are constrained by memory and power. Query-adaptive reranking and distillation are particularly valuable in edge and on-premises settings because they reduce the need for large models. A distilled router can run locally, a lightweight reranker can filter candidates, and a larger model can be called only when necessary. This hybrid architecture can balance performance, cost, and control.

Sustainability is an increasingly important dimension. Large language models require substantial energy for training and inference [25]. Retrieval adds additional computation, and reranking can multiply that cost if applied indiscriminately. Query-adaptive reranking reduces waste by matching computational effort to query difficulty. Distillation reduces the energy required per inference. Together, they can lower the carbon intensity of retrieval-augmented generation. However, sustainability must be measured across the full lifecycle. This includes index construction, embedding updates, model training, hardware manufacturing, and data center cooling. A system that is efficient at inference but requires frequent full reindexing may not be sustainable overall. Similarly, a distilled model that is deployed on inefficient hardware may consume more energy than a larger model on optimized infrastructure.

Infrastructure design must also address reliability and maintainability. Retrieval indexes must be updated as documents change. Embedding models must be versioned. Reranker and generator models must be evaluated after updates. Query-adaptive policies must be monitored for drift. This requires observability, logging, and automated testing. It also requires organizational capacity. Many institutions can build a prototype but struggle to maintain it. A sustainable deployment must plan for model retirement, data retention, and incident response. It must also consider vendor lock-in and interoperability. Open standards and modular components can reduce long-term risk.

The economics of deployment are similarly nuanced. A distilled model may reduce per-query cost, but it may also increase the number of queries by making the system more accessible. A cloud-based reranker may be cheap at low volume but expensive at scale. A locally hosted model may have high fixed costs but low marginal costs. Query-adaptive routing can optimize these trade-offs by directing traffic to the most cost-effective path. It can also support tiered service levels, where routine queries use lightweight models and premium queries use larger models. This raises fairness concerns if premium access is limited by payment. Policy design must ensure that essential information services remain accessible regardless of ability to pay.

7. Evaluation, Case Illustrations, and Trade-offs

Evaluating query-adaptive retrieval-augmented generation requires more than standard accuracy metrics. Retrieval quality can be measured with recall, precision, and ranking metrics. Generation quality can be measured with answer accuracy, faithfulness, and citation correctness. But a system-level evaluation must also consider latency, cost, energy, robustness, and fairness. It must test whether adaptive routing improves outcomes for difficult queries without degrading simple ones. It must examine whether distillation preserves performance across domains. It must assess whether the system behaves consistently under distribution shift. Benchmarks such as BEIR provide a useful starting point for zero-shot retrieval evaluation [6], but they do not cover all deployment conditions. Evaluation must be contextual and ongoing.

Consider enterprise search. Employees ask questions about policies, projects, and technical systems. A query-adaptive system might use a lightweight retriever for simple policy lookups

and a multi-role reranker for complex compliance questions. A distilled generator can draft answers with citations. A larger model can review high-risk responses. The benefits are lower latency and cost. The risks are misrouting and stale indexes. Governance must ensure that sensitive documents are not exposed and that answers are auditable. In this setting, the most important metric may not be answer fluency but citation fidelity and access control.

Consider healthcare. Clinicians may ask about drug interactions or treatment guidelines. Retrieval must be precise and up to date. Reranking must prioritize authoritative sources. Distillation may reduce cost, but the system must be conservative about uncertainty. A query-adaptive approach can escalate ambiguous or high-risk queries to a larger model and require human review. However, the system must also avoid alert fatigue and workflow disruption. Fairness concerns include language access and representation of rare conditions. Policy concerns include liability, privacy, and clinical validation. The architecture must therefore support audit trails, versioned evidence, and clear escalation paths.

Consider legal research. Lawyers ask nuanced questions about precedent and statutory interpretation. Retrieval must capture jurisdiction, date, and hierarchy of authority. Reranking must distinguish binding from persuasive sources. Distillation may be useful for initial filtering, but final judgment may require a larger model or human expert. A multi-role workflow can separate relevance, authority, and recency [14]. The trade-off is cost and complexity. The system must also manage confidentiality and privilege. These cases illustrate that efficiency is not a universal good. In high-stakes domains, the optimal system may deliberately choose to spend more computation to reduce risk.

The broader trade-off is between adaptability and accountability. Query-adaptive reranking improves efficiency by treating queries differently. But differential treatment can be difficult to explain and govern. Distillation improves deployability but can obscure the source of errors. The most responsible approach is to make the policy explicit, document the routing logic, and provide mechanisms for oversight. Evaluation should include stress tests, red-teaming, and participatory design with affected communities. The goal is not to eliminate trade-offs but to make them visible and contestable.

8. Conclusion

Efficient retrieval-augmented generation is a system-level challenge that cannot be solved by scaling a single component. This paper has argued that query-adaptive reranking and lightweight large language model distillation offer a coherent strategy for balancing relevance, latency, cost, and governance. Query-adaptive reranking treats computation as a scarce resource and allocates it according to query difficulty, domain, and risk. Distillation provides smaller models that can serve as routers, rerankers, and generators, enabling deployment in cloud, on-premises, and edge environments. Together, they can reduce energy consumption and improve responsiveness while preserving access to larger models for complex or consequential cases.

However, these techniques introduce new risks. Adaptive routing can misclassify queries, distillation can narrow model diversity, and layered architectures can obscure accountability. Governance must therefore be embedded in the design. Systems need transparent routing policies, robust access control, continuous evaluation, and mechanisms for appeal. Sustainability must be measured across the full lifecycle, not only at inference. Fairness must be assessed across domains, languages, and user groups. Policy must address data rights, transparency, and accountability without stifling innovation.

The future of retrieval-augmented generation will likely involve a portfolio of models and retrieval strategies rather than a single monolithic pipeline. The most effective systems will know when to retrieve, when to rerank, when to distill, and when to escalate. They will be judged not only by benchmark scores but by their robustness, fairness, sustainability, and institutional legitimacy. Efficient retrieval-augmented generation is therefore both an engineering problem and a socio-technical one. Its success depends on aligning architecture with governance and computation with public purpose.

References

1. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33, 9459–9474.
2. Zhou, D., & Yi, X. (2026). DISTRIBUTED AI-ASSISTED RISK ASSESSMENT FOR ZERO-TRUST MICROSERVICES: A TAXONOMY AND CONCEPTUAL FRAMEWORK. *AI and Data Science Journal*, 3(2), 1-13.
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. 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., Dziugaite, K., ... Sifre, L. (2022). Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning* (pp. 2206–2240). PMLR.
5. 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* (pp. 3929–3938). PMLR.
6. Thakur, N., Reimers, N., Rüclé, A., Srivastava, A., & Gurevych, I. (2021). BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.
7. Nogueira, R., & Cho, K. (2019). Passage re-ranking with BERT. *arXiv preprint arXiv:1901.04085*.
8. Khattab, O., & Zaharia, M. (2020). ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 39–48). Association for Computing Machinery.
9. Santhanam, K., Khattab, O., Saad-Falcon, J., Potts, C., & Zaharia, M. (2022). ColBERTv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 3715–3734). Association for Computational Linguistics.

10. Sun, W., Yan, L., Ma, X., Wang, S., Ren, P., Chen, Z., Yin, D., & Ren, Z. (2023). Is ChatGPT good at search? Investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* (pp. 14918–14937). Association for Computational Linguistics.
11. Pradeep, R., Sharifymoghaddam, S., & Lin, J. (2023). RankZephyr: Effective and robust zero-shot listwise reranking is a breeze! *arXiv preprint arXiv:2312.02724*.
12. Ma, X., Gong, Y., He, P., Zhao, H., & Duan, N. (2023). Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* (pp. 5303–5315). Association for Computational Linguistics.
13. Gao, L., Ma, X., Lin, J., & Callan, J. (2023). Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1762–1777). Association for Computational Linguistics.
14. Jin, Can, et al. "Rankflow: A multi-role collaborative reranking workflow utilizing large language models." *Companion Proceedings of the ACM on Web Conference 2025*. 2025.
15. Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.
16. Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.
17. Jiao, X., Yin, Y., Shang, L., Jiang, X., Chen, X., Li, L., Wang, F., & Liu, Q. (2020). TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020* (pp. 4163–4174). Association for Computational Linguistics.
18. Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M. (2020). MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems*, 33, 5776–5788.
19. Gou, J., Yu, B., Maybank, S. J., & Tao, D. (2021). Knowledge distillation: A survey. *International Journal of Computer Vision*, 129, 1789–1819.
20. Mialon, G., Dessi, R., Lomeli, M., Nalpantis, C., Pasunuru, R., Raileanu, R., Rozière, B., Schick, T., Dwivedi-Yu, J., Celikyilmaz, A., Grave, E., LeCun, Y., & Scialom, T. (2023). Augmented language models: A survey. *Transactions on Machine Learning Research*.
21. Asai, A., Wu, Z., Wang, Y., Sil, A., & Hajishirzi, H. (2024). Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.
22. 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). Association for Computing Machinery.
23. Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., ... Liang, P. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

24. Weidinger, L., Uesato, J., Rauh, M., Griffin, C., Huang, P.-S., Mellor, J., ... Gabriel, I. (2022). Taxonomy of risks posed by language models. In Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency (pp. 214–229). Association for Computing Machinery.
25. Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 3645–3650). Association for Computational Linguistics.