言語モデルの検索拡張生成の研究動向

Trends of retrieval-augmented generation

中内 遼吾 [1*]

Ryogo Nakauchi[1*]

[1] 東京都立産業技術大学院大学 Advanced Institute of Industrial Technology
*Corresponding author: Ryogo Nakauchi, nakauchi-ryogo@aiit.ac.jp

Abstract Large language models (LLMs) possess vast amounts of knowledge, yet they often lack sufficient domain-specific expertise. To effectively integrate LLMs into business operations, it is essential to enable them to utilize domain knowledge in some capacity. Retrieval-Augmented Generation (RAG) has emerged as a widely adopted approach to address this challenge. However, achieving a practical level of effectiveness with RAG remains a complex task. Consequently, various methods have been proposed to enhance the effectiveness of RAG. This paper reviews the current research trends aimed at improving RAG's effectiveness.

Keywords large language models; retrieval-augmented generation

1 はじめに

大規模言語モデル（LLM）は自然言語処理の分野に大きな進歩を示し，産業界においても急速にその活用が模索されている．知識抽出器としての質問応答タスク，要約やセンチメント分析のような自然言語処理タスクのほか，その汎用性の高さを応用した機器制御[1]などの業務システム用コンポーネントとしても利用されるに至っている．OpenAI が公開した ChatGPT[2] は前提知識なしで利用できるチャット形式の UI を提供することで，言語モデルの会話能力の高さを示した．以降，多くのモデルがシステムプロンプトとユーザープロンプトを入力として受け取り，これに対する自然な会話形式の応答文を生成するように指示チューニングされた形で公開されるに至っている．この会話能力を直接的に用いた活用法として，ドメイン知識に特化させた質問応答タスクを業務用途で利用するユースケースが広くみられる．

言語モデルは事前学習を通じて大量の知識を内部に蓄積するが，専門分野など個別性が高い知識については不十分な水準にとどまるため，業務応用のためには何らかの追加的対応をとる必要がある．各企業が自社の業務に特化した情報を言語モデルに扱わせるためには追加学習によってモデル内部に知識追加を図るか，知識コーパスから検索した参照コンテキストを基に回答生成を行う検索拡張生成（Retrieval-Augmented Generation, RAG[3,4]）の仕組みを構築する必要がある．本稿では， RAG について概要を示したうえで，その精度改善を図る手法の研究動向について整理する．

2 RAG の概要

本章では RAG の概要を整理するため，基本構成例を示す．RAG は言語モデルを使って回答生成の役割を担う Reader モジュールと，知識コーパスから抽出した任意の件数のチャンクを参照コンテキストに統合して Reader に渡す役割を担う Retriever モジュールを組み合わせた Retriever-Reader モデルで構成する．Retriever は知識コーパス，類似度評価関数，ユーザークエリと知識コーパスをベクトル表現に埋め込むエンコーダなどから構成される．知識コーパスにベクトルデータベースを用いる場合，データベースは情報源から抽出したテキストデータを言語モデルのコンテキストウインドウに収まるようにスプリットしたチャンクと，そのベクトル表現をインデックスとしたレコードを格納する．基本的な手法である DPR[5,6]の場合，検索はベクトル変換されたユーザークエリと各インデックスの類似度をコサイン類似度やドット積などのメトリクスを使ってランキング評価し，スコアが高い任意の件数のレコードを上位から抽出する形で実行する．知識コーパスにはベクトルデータベースのほか，Web 検索や埋め込み表現によらない既存の業務用データベースなどが組み合わせられる場合もある．ここで抽出したチャンクを特定の形式に整形した参照コンテキストをシステムプロンプトに統合し，ユーザープロンプトと共に Reader に渡して回答生成を行う．ベクトルデータベースに各チャンクのソース情報も格納しておくことによって回答の根拠となるドキュメント名の提示を行うケースもある．

RAG は知識を外部化することで大規模な計算資源が必要な追加学習を必要とせずに回答生成を拡張できる点がメリットである．情報更新のしやすさや正確性，新しい言語モデルが登場した際のシステム更新のしやすさ，根拠の提示などが重要なユースケースで選好される．デメリットには，Retriever の構成によっては検索に時間がかかり応答時間が長くなることや，用途や知識コーパスによっては Retriever の精度向上の難易度が高いこと，複数のチャンクに断片化された情報の総合が難しいことが挙げられる．

3 RAG の研究動向

前章に示したようなベーシックな構成だけでは回答精度が実用レベルにまで達しないことが多い．このため，Retriever-Reader モデルを構成するコンポーネントごとに様々な精度改善手法が提案されている．本章ではこうした手法の研究動向について項目別に整理する．

チャンキング

チャンキングは膨大な情報源を必要な長さに切り出すことで Reader 側の言語モデルが適切に情報を扱えるようにする役割を持つ．一方で，文章の切断位置によっては重要な文脈情報が失われるリスクがあるため，切断位置やチャンクサイズの設計方法を巡って様々な議論がある．

Xia Y ら[7]は，チャンクサイズが長い方が文脈情報を多く保持できる反面，コサイン類似度での検索精度は短い方が優れる

Received: 2024-10-30; Published online: 2025-02-28

というトレードオフを指摘し，長いチャンク（親チャンク）と短いチャンク（子チャンク）を組み合わせて併用する Parent Document Retriever[8]の利用を提案している．これは，端的に必要情報が格納されていることが期待される子チャンクを検索に用い，それを包含するパラグラフ単位の親チャンクを Reader に渡して回答生成を行うことで双方のデメリットを補う仕組みとなっている．

Jiang Z ら[9]は，関連性の高いチャンク同士をグループ化した集合チャンクを参照コンテキストに統合することによって，必要な情報が複数のチャンクに分散していても Recall が低下しないようにする手法を提案している．この手法では言語モデルに渡される参照コンテキストは比較的長文になるが，Liu NF ら[10]は，長文を渡された場合に言語モデルはその中間付近に位置する情報を見逃しやすい傾向にあることを指摘している．こうした問題を重視する Yu W ら[11]や Fangyuan X ら[12]は，Reader 側に配置する回答生成用の言語モデルとは別に，Retriever 側に情報集約と取捨選択を担う言語モデルを配置し，長文に渡る参照コンテキストから重要な情報だけをまとめた文章を作成して Reader に渡す手法を提案する．

チャンキングの改善においては地道な手法の積み重ねも重視される傾向にある．近年開催された RAG の精度を競うコンペティションにおいて上位受賞した解法[13,14]では，正規表現などルールベースによるジャンク文字列の排除や文法規則，文書形式の特徴によるチャンキングといった手法を多数併用することで精度向上が図られている．Yuan Y ら[15]は，文頭または文末の疑問詞の有無を指標として各センテンスが質問文か否かを識別し，問題提起文とそれに続く文を連結することで文脈情報が切断されることを防ぐ方法を提案する．

インデキシング

検索に用いるベクトルデータベースのインデックスには，知識コーパスを何らかの手法で埋め込んだベクトル表現が利用されるが，埋め込み手法よって検索時の特性が変わるため，様々な手法が提案されている．

ユーザークエリとインデックスの文字列一致度によるアプローチでは TF-IDF[16]や BM25[6,17]といった伝統的な疎ベクトルが用いられる．意味的類似度によるアプローチでは深層学習モデルを使ったエンコーダによって埋め込んだ密ベクトルを用い，ユーザークエリとのコサイン類似度やドット積でインデックスの類似度評価が行われる．明確なキーワードが存在するユースケースでは疎ベクトルに利点があるが，わずかでも表記揺れが生じうるようなユースケースでは密ベクトルによるアプローチに利点が生じやすい．他方で，ユーザークエリが極端に短い場合などに密ベクトルが適さないケースもある．[18]では，こうした性質の異なる複数の手法をアンサンブルして欠点を補う Reciprocal Rank Fusion (RRF) が提案され，ハイブリッド検索が単独の手法による検索よりも精度に優れることが検証されている．複数の異なる密ベクトルを用いたアンサンブルを採用するアプローチ[13,19]では，複数のランキングモデルによるスコアリングを重み付きでアンサンブルする手法が提案されている．

密ベクトルを用いるベーシックな手法である DPR ではインデキシングにバイエンコーダが用いられるが，クロスエンコーダを用いる場合には，ユーザークエリとドキュメントを連結した結合埋め込みベクトルが計算されることで，両者間のより複雑な相互作用が期待される．クロスエンコーダはバイエンコーダに比べて精度が向上しやすいが，計算コストが大きくなるデメリットがある．その欠点を補う手法として，ColBERT[20,21]や SPLADE[22,23]などが提案されている．これらの手法が採用する後期相互作用メカニズムでは，クエリとドキュメントを個別にエンコードした後に，トークンごとのベクトル間でコサイン類似度やドット積を計算して最終的な検索結果が決定される．クエリとドキュメントのエンコードが独立であるため，データベース側のベクトルは事前に作成しておくことができ，検索時にはクエリ側のベクトルのみを新たにエンコードすれば済むため伝統的なクロスエンコーダに比べて応答速度を改善できる利点がある．

Jégou H ら[24]は，ベクトルインデックスの探索空間をボロノイ領域に分割し，ユーザークエリとの類似度の高い領域に局限して個別ベクトルとの類似度評価を行うことで，ベクトルデータベースが大規模な場合にも計算量と検索時間を短縮する手法を提案している．

リランキング

前節でみたような各種のインデキシング手法は，より複雑な手法ほど精度を高めやすい一方で計算コストが増大しやすくなるトレードオフがあるため，複数の手法を段階によって使い分ける手法が議論されている．

Glass M ら[25]や Nogueira R ら[26]は，BM25 などの計算コストの軽量な手法で広めに検索対象を絞り込んだあと，次の段階でより精度の高い手法を使って類似度評価を行い，参照コンテキストに用いるチャンクを決定する手法を提案している．これらの手法は，類似度のランキングを多段階に渡って作成して評価を行うことからリランキングと呼ばれる．リランキングに言語モデルを利用することで，検索上位に抽出するレコードの多様性を高めるなど，目的に応じた評価基準の調整を柔軟に行えるようにする手法[27]も提案されている．

Yang R ら[28]は，知識グラフからの情報取得に類似度，回答拡張，周辺関連性最大化（MMR）の 3 つの評価を組み合わせてリランキングを行うことで，情報取得の信頼性向上を行うフレームワークを提案している．

ノイズ軽減

Retriever 側で回答生成に必要な情報だけに完全に絞り込んだ情報抽出を行うことは現実的には困難なため，Reader 側に渡される参照コンテキストには多くの場合，必要情報と共に回答に関係しないノイズ情報も含まれる．このため RAG の精度はこうした雑多性のある情報から言語モデルが必要な情報を適切に取捨選択する能力に左右されることになるが，この能力自体は言語モデルにとって自明のものではなく，その対処法をめぐって議論がある．

Hsia J ら[29]は，モデルごとに参照コンテキストの分量やノ

イズへの頑健性に大きな差があることを検証し，一般的なリーダーボードだけを基にモデルの性能を判断するのではなく，RAG 専用の評価フレームワークを用いてノイズ耐性の高いモデルを選択することを提案している．Chen J ら[30]はモデルごとのノイズ耐性を比較するためのベンチマークを提案している．

Shi F ら[31]や Weston J ら[32]の研究では，必要な情報が含まれていなかったり間違った情報が含まれていたりする場合でも，参照コンテキストが渡されると言語モデルは学習済み知識よりもその内容を優先して回答しようとする傾向が顕著であることが示されている．このため，タスクによってはむしろ Retriever を経由することによって回答の精度が低下する可能性がある．この問題を軽減するため，Yuan Y ら[15]は回答に必要な知識が最新情報の検索が必要な程度にリアルタイム性を持つ内容であるかを分類予測する機械学習モデルを別途構築し，Retriever の利用が不要と判定された場合には情報取得をスキップする仕組みを提案している．Wu K ら[33]は Retriever から与えられた参照コンテキストを考慮した回答とそうでない回答のトークン確率に着目し，後者の方が確率の高い場合には参照コンテキストを使わずに生成した回答を採用する機構を導入することによって，ノイズ情報が与えられた際の精度低下を防ぐ手法を提案している．Asai A ら[34]は，参照コンテキストの証拠性をスコアリングする予測モデルを別途構築し，高いスコアが付与されたチャンクだけを選別して Reader に提供する手法を提案している．また，回答生成にあたって Retriever を呼び出す必要があるか否かの判断自体を言語モデルが行えるようにファインチューニングし，不要と判断した場合には Retriever の利用をスキップするフレームワーク[35]も提案されている．

Retriever が選択した参照コンテキストからノイズを除去する方向性も議論されており，ルールベースでの除外ができないノイズを識別するフィルタリングモデルを学習させて併用する手法[36]や，トークンレベルでこれを実践する手法[37]も提案されている．

Reader に用いる言語モデル自体のノイズ耐性に改善を加えるアプローチとしては，ノイズ情報を無視する能力を向上させるファインチューニング手法[38]，モデルの学習コーパスに意図的にノイズの多い情報を含めることによって低品質の参照コンテキストが渡された際の頑健性を高める手法[39]がそれぞれ提案されている．Yuan Y ら[15]は，プロンプティングによる文脈内学習によってその能力を高める手法を提案している．

データオーグメンテーション

ユーザークエリとユーザーが実際に求めている知識を含むチャンクのそれぞれが類似度による関連付けを行うにあたって常に十分な表現を持つわけではなく，ここにギャップが生じることがある．この点を改善するため，言語モデルを使ってそれぞれの表現をオーグメンテーションしたうえで検索する手法が議論されている．

Wu M ら[40]は，知識コーパスの側を言語モデルを使ってオーグメンテーションする手法を提案している．通常はユーザークエリとチャンクの類似度は 1 対 1 で評価されるが，この手法では予め各チャンクのソースとなっているドキュメントの全文から，その内容を解釈した疑似タイトルと複数の説明文からなる疑似クエリを言語モデルに生成させる．そのうえで，疑似タイトル・疑似クエリ・チャンク本体からなる集合とユーザークエリの類似度評価を行うことで，多面的な文脈からユーザーの求める情報を抽出することを図るものとなっている．

Ma X ら[41]は，ユーザークエリの側を言語モデルを使ってオーグメンテーションする手法を提案している．入力されたユーザークエリを解釈する役割の言語モデルを Retriever 側に配置し，文脈補間やパラフレーズにより複数のパッセージを生成することによって，元のユーザークエリと類似度が一致していなくても目的に合致する情報がある場合に必要な関連付けを行える可能性を高めようとするものである．

回答生成

検索プロセスが完了したあと，抽出された参照コンテキストを最終的に Reader に渡して回答生成するステップでは，参照コンテキストの構造化や後処理などの手法が議論されている．

Vu T ら[42]は，Retriever が取得した複数のチャンクを目的に応じて作り込んだプロンプトテンプレートに統合し，その情報をどのように利用して回答生成すべきか Few-shot 文脈内学習を行うことで，モデルが参照コンテキストを効果的に処理して回答生成することを促す手法を提案している．

Asai A ら[35]は，回答生成用モデルとは別に，生成された回答文が Retriever から与えられた参照コンテキストに裏付けられた内容になっているかを評価する言語モデルを配置することで，参照コンテキストと回答の関連付けを高める手法を提案している．

チャンキングと同じく回答生成の改善にあたっても地道な手法の積み重ねが重視されており，回答生成の段階において Retriever が抽出した参照コンテキストに対して文脈圧縮や要約などの後処理を実施する手法[43,44]も提案されている．

4 おわりに

本稿では，言語モデルの検索拡張生成に関する研究動向をまとめた．Retriever-Reader モデルは複数のコンポーネントから構成されるため，各部分について精度改善手法の議論が存在する．その中でも，近年提案される手法には，回答生成用とは別に言語モデルや分類系の機械学習モデルを Retriever 側に配置することで，より高度なデータ前処理や制御を目指す傾向がみられる．回答文の作成過程を多段階に分節することでその品質を引き上げようとする試みは推論スケーリングなどにも通底する発想である．ユーザーが入力するプロンプトに問題の複合性が含まれることは多くあるため，これを必要に応じてサブタスクに分解した上で，最適な個別手法で部分解を作成して最終的なアウトプットに統合するアプローチは今後も進展していく可能性がある．

近年小規模な言語モデルにおいても性能向上が顕著であるものの，依然として追加学習は計算コストが高額になりやすく，

ユースケースや導入規模によっては敷居が高くなりがちである．それに対して RAG は相対的に小規模なシステムからでも開発することができ，企業が自社ドメイン知識への特化用途で言語モデルを活用するための第一歩としやすい選択肢である．言語モデルによるデータオーグメンテーションの研究がでてきたことで，データ不足に直面する企業においても導入の敷居が下がることが期待できる．

参考文献

1. 三菱電機とソラコム・松尾研究所「IoT × GenAI Lab」が，IoT と生成 AI を応用した空調機器制御の実証実験を実施 - ニュース. In: 株式会社ソラコム コーポレートサイト [Internet]. [cited 20 Oct 2024]. Available: https://soracom.com/ja/news/20240711-iot-genai-poc-report
2. ChatGPT. [cited 20 Oct 2024]. Available: https://chatgpt.com/
3. Chen D, Fisch A, Weston J, Bordes A. Reading Wikipedia to Answer Open-Domain Questions. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2017. pp. 1870–1879. doi:10.18653/v1/P17-1171
4. Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. arXiv [cs.CL]. 2020. Available: http://arxiv.org/abs/2005.11401
5. Karpukhin V, Oğuz B, Min S, Lewis P, Wu L, Edunov S, et al. Dense passage retrieval for open-domain question answering. arXiv [cs.CL]. 2020. Available: http://arxiv.org/abs/2004.04906
6. Izacard G, Grave E. Leveraging passage retrieval with generative models for open domain question answering. arXiv [cs.CL]. 2020. Available: http://arxiv.org/abs/2007.01282
7. Xia Y, Chen J, Gao J. Winning Solution For Meta KDD Cup' 24. 2024 KDD Cup Workshop for Retrieval Augmented Generation. 2024. Available: https://openreview.net/pdf?id=oWNPeoP1uC
8. How to use the Parent Document Retriever. [cited 29 Oct 2024]. Available: https://python.langchain.com/docs/how_to/parent_document_retriever/
9. Jiang Z, Ma X, Chen W. LongRAG: Enhancing retrieval-augmented generation with long-context LLMs. arXiv [cs.CL]. 2024. Available: http://arxiv.org/abs/2406.15319
10. Liu NF, Lin K, Hewitt J, Paranjape A, Bevilacqua M, Petroni F, et al. Lost in the middle: How language models use long contexts. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2307.03172
11. Yu W, Zhang H, Pan X, Ma K, Wang H, Yu D. Chain-of-note: Enhancing robustness in retrieval-augmented language models. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2311.09210
12. Fangyuan X, Weijia S, Eunsol C. RECOMP: Improving retrieval-augmented LMs with compression and selective augmentation. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2310.04408
13. Wang Q, Yuan J, Huang X, Yan J, Xiao R, Ding X. Hybrid Retrieval Systems Based on LLMs Embedding and Enhancement. KDD 2024 OAG-Challenge Cup. 2024. doi:10.1145/nnnnnnn.nnnnnnn
14. Ouyang J, Luo Y, Cheng M, Wang D, Yu S, Liu Q, et al. Revisiting the solution of Meta KDD cup 2024: CRAG. arXiv [cs.IR]. 2024. Available: http://arxiv.org/abs/2409.15337
15. Yuan Y, Liu C, Yuan J, Sun G, Li S, Zhang M. A hybrid RAG system with comprehensive enhancement on complex reasoning. arXiv [cs.CL]. 2024. Available: http://arxiv.org/abs/2408.05141
16. Sparck Jones K. A statistical interpretation of term specificity and its application in retrieval. J Doc. 1972;28: 11–21. doi:10.1108/eb026526
17. The Probabilistic Relevance Framework: BM25 and Beyond. In: ResearchGate [Internet]. [cited 25 Oct 2024]. Available: https://www.researchgate.net/publication/220613776_The_Probabilistic_Relevance_Framework_BM25_and_Beyond
18. Cormack GV, Clarke CLA, Buettcher S. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. New York, NY, USA: ACM; 2009. doi:10.1145/1571941.1572114
19. Deotte C, Onodera K, Puget J-F, Schifferer B, Titericz G. Winning Amazon KDD Cup'23. Amazon KDD Cup 2023 Workshop. 2023. Available: https://openreview.net/pdf?id=J3wj55kK5t
20. Khattab O, Zaharia M. ColBERT: Efficient and effective passage search via contextualized late interaction over BERT. arXiv [cs.IR]. 2020. Available: http://arxiv.org/abs/2004.12832
21. Santhanam K, Khattab O, Saad-Falcon J, Potts C, Zaharia M. ColBERTv2: Effective and efficient retrieval via lightweight late interaction. arXiv [cs.IR]. 2021. Available: http://arxiv.org/abs/2112.01488
22. Formal T, Piwowarski B, Clinchant S. SPLADE: Sparse lexical and expansion model for first stage ranking. arXiv [cs.IR]. 2021. Available: http://arxiv.org/abs/2107.05720
23. Formal T, Lassance C, Piwowarski B, Clinchant S. SPLADE v2: Sparse lexical and expansion model for Information Retrieval. arXiv [cs.IR]. 2021. Available: http://arxiv.org/abs/2109.10086
24. Jégou H, Douze M, Schmid C. Product quantization for nearest neighbor search. IEEE Trans Pattern Anal Mach Intell. 2011;33: 117–128. doi:10.1109/TPAMI.2010.57
25. Glass M, Rossiello G, Chowdhury MFM, Naik AR, Cai P, Gliozzo A. Re2G: Retrieve, Rerank, Generate. arXiv [cs.CL]. 2022. Available: http://arxiv.org/abs/2207.06300
26. Nogueira R, Cho K. Passage re-ranking with BERT. arXiv [cs.IR]. 2019. Available: http://arxiv.org/abs/1901.04085
27. Gao J, Chen B, Zhao X, Liu W, Li X, Wang Y, et al. LLM-enhanced Reranking in Recommender Systems. arXiv [cs.IR]. 2024. Available: http://arxiv.org/abs/2406.12433
28. Yang R, Liu H, Marrese-Taylor E, Zeng Q, Ke YH, Li W, et al. KG-Rank: Enhancing large language models for medical QA with knowledge graphs and ranking techniques. arXiv [cs.CL]. 2024. Available: http://arxiv.org/abs/2403.05881
29. Hsia J, Shaikh A, Wang Z, Neubig G. RAGGED: Towards informed design of retrieval augmented generation systems. arXiv [cs.CL]. 2024. Available: http://arxiv.org/abs/2403.09040
30. Chen J, Lin H, Han X, Sun L. Benchmarking large language models in retrieval-Augmented Generation. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2309.01431
31. Shi F, Chen X, Misra K, Scales N, Dohan D, Chi EH, et al. Large Language Models Can Be Easily Distracted by Irrelevant Context. International Conference on Machine Learning. PMLR; 2023. pp. 31210–31227. Available: https://proceedings.mlr.press/v202/shi23a.html
32. Weston J, Sukhbaatar S. System 2 Attention (is something you might need too). arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2311.11829
33. Wu K, Wu E, Zou J. ClashEval: Quantifying the tug-of-war between an LLM's internal prior and external evidence. arXiv [cs.CL]. 2024. Available: http://arxiv.org/abs/2404.10198
34. Asai A, Gardner M, Hajishirzi H. Evidentiality-guided generation for knowledge-intensive NLP tasks. arXiv [cs.CL]. 2021. Available: http://arxiv.org/abs/2112.08688
35. Asai A, Wu Z, Wang Y, Sil A, Hajishirzi H. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2310.11511
36. Wang Z, Araki J, Jiang Z, Parvez MR, Neubig G. Learning to filter context for retrieval-augmented generation. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2311.08377
37. Berchansky M, Izsak P, Caciularu A, Dagan I, Wasserblat M. Optimizing retrieval-augmented reader models via token elimination. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2310.13682
38. RAFT: Adapting Language Model to Domain Specific RAG. [cited 2 Oct 2024]. Available: https://arxiv.org/html/2403.10131v2

39. Yoran O, Wolfson T, Ram O, Berant J. Making retrieval-augmented language models robust to irrelevant context. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2310.01558
40. Wu M, Cao S. LLM-augmented retrieval: Enhancing retrieval models through language models and doc-level embedding. arXiv [cs.IR]. 2024. Available: http://arxiv.org/abs/2404.05825
41. Ma X, Gong Y, He P, Zhao H, Duan N. Query rewriting for retrieval-augmented large Language Models. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2305.14283
42. Vu T, Iyyer M, Wang X, Constant N, Wei J, Wei J, et al. FreshLLMs: Refreshing large language models with search engine augmentation. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2310.03214
43. Arefeen MA, Debnath B, Chakradhar S. LeanContext: Cost-efficient domain-specific question answering using LLMs. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2309.00841
44. Liu J, Li L, Xiang T, Wang B, Qian Y. TCRA-LLM: Token compression retrieval augmented large language model for inference cost reduction. arXiv [cs.CL]. 2023. Available: http://arxiv.org/abs/2310.15556