Graph RAG における GNN 検索器の埋め込み次元スケーリングの検討

清 恵人 [1]　Sunil Kumar Maurya[2]　小橋 洋平 [2]
[1] 東京都市大学　[2] 東京大学
g2581429@tcu.ac.jp　yohei.kobashi@weblab.t.u-tokyo.ac.jp

概要

RAG reduces LLM hallucinations by grounding answers in external knowledge, and Graph RAG enables multi-hop reasoning via knowledge graphs. GNN-based retrievers (GNN-RAG) offer accurate, low-cost retrieval, but performance drops on complex long-hop questions and scaling rules are unclear. We analyze how scaling the GNN retriever affects retrieval in Graph RAG. Scaling only the linear-layer width often saturates or degrades under limited supervision, while increasing the effective training signal stabilizes scaling. These results indicate that robust scaling requires jointly increasing model capacity and training data for multi-hop KGQA.

1　はじめに

現在，ChatGPT をはじめとする大規模言語モデル（LLM）を活用したサービスが急速に普及し，研究・開発のみならず企業システムにも広く導入されている．特に，社内文書やドメイン知識を取り込んだチャットボットなど，外部知識を検索して回答に反映する Retrieval-Augmented Generation（RAG) や Graph RAG の研究が進み，学習時に含まれない最新情報や企業固有の知識に対しても回答可能な枠組みが整ってきた.

一方で，Graph RAG における検索・推論性能は，部分グラフ上でどれだけ正確に根拠ノード／パスを抽出できるかに強く依存する．特に GNN は，マルチホップの質問には優れいてるものの 1-hopRAG の性能は性能が低下しやすい. また近年は推論能力の高い LLM の登場により，LLM 主導探索との比較において GNN の優位性（低コスト・安定性）を維持するための設計がより重要になっており GNN-RAG が提案されている [1].

本稿では Graph RAG における推論器として GNN に着目し，モデル規模とデータ規模のスケーリングが性能に与える影響を体系的に検証する．具体的には，(i) GNN のパラメータ数の増加による性能向上の限界を確認した上で，(ii) 学習データ量の拡大や質問パラフレーズによる実質的なデータ増強が性能に与える寄与を比較し，(iii) それらを踏まえた Graph RAG における有効なスケール戦略を提示する.

2　関連研究

2.1　RAG

Retrieval Augmented Generation (RAG) は,LLM の学習に含まれない最新情報や専門的な知識を正確に解答することを目的とし, ハルシネーションを抑制する手法として提案された.RAG はユーザーの入力に対して, 外部の知識ベースから関連する情報を検索する. 次に検索した情報をコンテキストとしてプロンプトに組み込む,In-context learning により正確な解答を生成する. これによりモデルを再学習させることなく, 専門知識や最新情報に正確に解答することができる.

2.2　Graph RAG

ベクトル検索に基づく RAG は, 局所的な情報の検索には優れているものの, 複数のドキュメント間といった大域的な情報には優れていない. 複数のエンティティ間にまたがる検索や複雑な関係性を捉える検索などに,GraphRAG が提案されている.GraphRAG は, 外部データベースを Knowledge Graph(KG) にすることで RAG の問題点を解消しようと試みている.KG は, ノード (エンティティ) とエッジ (リレーション) の集合を $G = (V, E)$ として表現することでテキスト間の構造的な依存関係を保持する. そのため, 複雑な関係性や大域的な情報の検索に優れている.

2.3 GNN-RAG

GNN-RAG は，知識グラフ上の検索を LLM ではなく学習済みの GNN で行い，質問に関連するエンティティやリレーションを高精度に抽出する Graph RAG の一種である．従来の Graph RAG では，LLM が KG を逐次探索して根拠パスを組み立てるため，多くのトークンを消費し，探索の揺らぎによって誤った根拠を参照することがあった．これに対し GNN-RAG では，質問 q と質問に対応する部分グラフ G_q を入力として，GNN が各ノードのスコア $s(v|q)$ を推定し，上位のノード／パスのみを LLM に渡すことで，トークン消費を抑えつつ安定した検索を実現する．以下では，GNN 検索器 [2] の推論手順を Algorithm 1 に示す．

Algorithm 1 GNN-RAG の GNN アーキテクチャ

入力: 質問 q，KG 部分グラフ G_q，seed エンティティ $\{e\}_q$，ハイパーパラメータ T, K, L.
初期化: K 個の命令（instruction）$\{i^{(k)}\}_{k=1}^{K}$，ノード表現 H^{in}.
for $t = 1$ **to** T **do**
  for $l = 1$ **to** L **do**
    for $k = 1$ **to** K **do**
      $h_v^{(k,l)} = \phi\left(\left\{m_{v' \to v}^{(l)} : v' \in \mathcal{N}(v|i^{(k)})\right\}\right)$.
    end for
    $h_v^{(l)} = \psi\left(h_v^{(l-1)}, \left\{\tilde{h}_v^{(k,l)}\right\}_{k=1}^{K}\right)$.
  end for
  $H^{\text{out}} \leftarrow H^{(L)}$，$H^{\text{in}} \leftarrow H^{(L)}$ とする．
  H^{out} を用いて $\{i^{(k)}\}_{k=1}^{K}$ を更新する．
end for
出力: h_v^{out} に基づき，ノード v を解（answer）/非解（non-answer）として分類する．

2.4 Scaling 則

2.4.1 スケーリング則

大規模言語モデルでは，モデル規模 N や学習トークン数 D を増やすことで，損失がべき乗則に従って減少することが報告されている [3].

2.4.2 データボトルネック

Transformer を始めとする LLM の学習では，固定データ量のままモデルのみを拡大すると under-training となり非効率になることが指摘されている．Hoffmann らは，計算資源が固定された条件下で，モデル規模の増加に応じて学習トークン数も増やすことが性能向上に重要であることを示した [4]. 同様に GNN 検索器でも，Knowledge Graph Question Anwer のデータ量が不足したままモデルサイズのみを増やすと，性能が劣化する可能性がある．

2.4.3 学習率・バッチサイズのスケーリング

大規模学習では，学習率 η とバッチサイズ B の同時調整が性能・収束安定性に影響する．Goyal らは，大バッチ化に伴い学習率を比例的に増やす線形スケーリング則と，初期のウォームアップが有効であることを示した [5]．ここで (η_0, B_0) は基準設定である．

$$\eta(B) = \eta_0 \frac{B}{B_0}, \tag{1}$$

3 提案手法

3.1 パラメータ増加の方針

本稿では，関連研究で示した GNN 検索器（Algorithm 1）の構造（層数 L，反復回数 T，instruction 数 K など）を基本的に固定し，隠れ次元（幅）d のみを増加させることでモデル規模をスケールさせる．ここで d は，各層のノード更新やメッセージ計算に用いる中間表現の次元である．一方で，エンティティ埋め込みおよびリレーション埋め込みの次元は固定し，埋め込みを層内で変換する線形層のパラメータのみを増加させる．同一の推論手順, ハイパーパラメータ設定の下で，モデル容量の増加が検索性能に与える影響を明確に評価できる．

一方で，GNN は層数を過度に増やすと表現が均質化する over-smoothing が生じうるため，深さ方向（L の増加）によるスケールは本稿では扱わず，幅方向（d）のスケーリングに焦点を当てる．

3.2 転移学習によるデータ量増加

モデル規模（幅 d）を拡大すると，目的データのみではデータ量が不足し，汎化性能の伸びが頭打ちになる可能性がある．そこで本稿では，転移学習によりデータ量を拡充する．

具体的には，まず目的タスクと同形式の別データセットを用いて GNN 検索器を事前学習し，その後，目的データセットで追加学習を行う二段階学習を採用する．これにより，関連データ上で獲得した質問表現とグラフ構造の対応づけや関係パターンの知

識を初期値として活用でき，データ量増加によるスケール則の検証を行う．

4 実験

4.1 実験設定

本研究では，関連研究で示したベース検索器（Algorithm 1）を用い，GNN が出力する検索結果のみを評価対象とする．すべての実験で，部分グラフ抽出およびサブグラフサイズ等の前処理は固定し，検索器の学習条件のみを比較する．

4.1.1 データセット

目的データセットを CWQ とし，質問 q ごとに部分グラフ G_q と正解集合 A_q が与えられるとする．転移学習では，目的データと同形式の関連データセット WebQSP と FreebaseQA を用いる．

4.1.2 モデル設定

層数 L（および反復回数 T，instruction 数 K を用いる場合はそれら）を固定し，隠れ次元（幅）d のみを $d \in \{50, 150, 200, 250, 300\}$ の範囲で変更する．

4.1.3 学習設定

学習条件は原則として統一した．ただし，モデルの幅 d を増加させると最適化が不安定になりうるため，関連研究に基づき，学習率のみをモデル規模に応じて調整した．具体的には，基準モデル（$d = 50$）の学習率を 5.0×10^{-4} とし，幅の増加に伴い学習率を段階的に低下させた．学習率を表 1 に示す．

表 1 幅スケーリングに伴う学習率

dim d	Parameters	Ratio vs base	Learning Rate
50	887,970	1×	5.0×10^{-4}
150	4,223,270	4.75×	3.4×10^{-4}
200	6,560,770	7.39×	3.0×10^{-4}
250	9,363,270	10.55×	2.8×10^{-4}
300	12,630,770	14.22×	2.6×10^{-4}

4.1.4 評価指標

検索精度は Hits@1（Hit1）および F1 で評価する．Hit1 は，最上位候補が正解集合に含まれる割合として定義する．F1 は，recall と precision から算出する．

4.2 実験 1：幅スケーリング

目的データ CWQ のみを用いて検索器を学習し，幅 d の増加が検索性能に与える影響を検証する．具体的には，$d \in \{50, 150, 200, 250, 300\}$ の各設定について同一の学習手順でモデルを学習し，test 上で Hit1 および F1 を比較する．この実験により，モデルサイズの増加のみで性能が改善するか，あるいは性能が頭打ちになるかを確認する．

4.3 実験 2：転移学習によるデータ増強

実験 1 で観測される頭打ちがデータ不足に起因する可能性を検討するため，WebQSP で事前学習した後に FreebaseQA,CWQ で微調整する二段階学習（転移学習）を行う．事前学習・微調整のいずれも，同一のベース検索器（Algorithm 1）を用いる．幅 d は実験 1 と同様に $d \in \{50, 150, 200, 250, 300\}$ を比較し，転移学習が幅スケーリング時の性能低下／頭打ちを緩和するかを評価する．

4.4 実験 3：質問変換によるデータ増強

目的データ CWQ の学習データに対して，質問 q をパラフレーズ q' に変換し，(q', A_q) を追加することで学習データの分布を保ったまま学習データを増強する．パラフレーズ生成には LLM 等を用い，正解集合 A_q は元質問と共有する．増強後のデータに対して検索器を学習し，幅 d の増加に対する性能変化を実験 1 と同一条件で比較する．これにより，表層表現の多様化による汎化性能の改善と，スケールアップ時のデータ量不足の緩和を検証する．

4.5 実験結果と考察

本節では，Hit1 および F1 に基づき，幅スケーリング，転移学習，質問パラフレーズによるデータ増強の 3 条件におけるスケーリング挙動を比較する．

4.5.1 実験 1：幅スケーリング

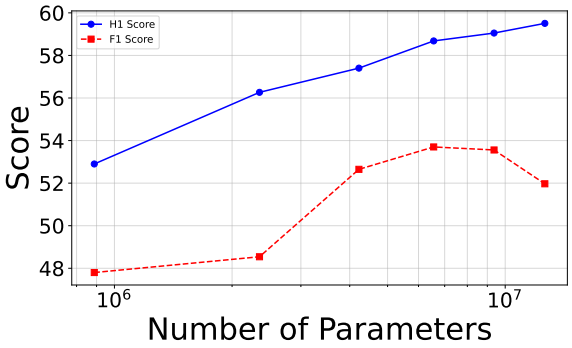

図 1 実験 1：幅 d のみを増加させた場合の性能．

図 1 に幅 d を増加させた際の Hit1 および F1 の推

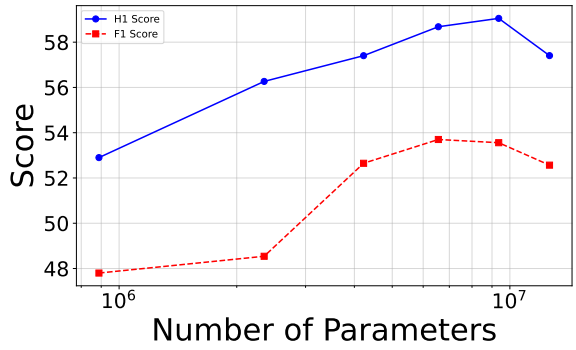

図 2　実験 2：転移学習における性能

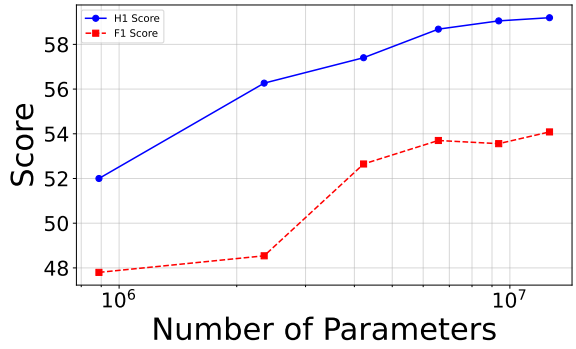

図 3　実験 3：質問パラフレーズにより学習データを増加させた性能.

移を示す．小さい幅では性能が改善する一方，最大幅では性能が低下し，単純なパラメータ数増加のみでは一貫した改善が得られないことが確認された．この結果は，本設定において大規模モデルが十分に学習されず，データ量がボトルネックとなっている可能性を示唆する．

4.5.2　実験 2：転移学習

図 2 に転移学習を導入した場合の結果を示す．転移学習によりデータ量の増加を狙ったが，最大幅において性能が一貫して改善するとは限らなかった．一つの要因として，事前学習データと目的データの間で，質問の複雑性や分布にギャップが存在する可能性がある．すなわち，事前学習で獲得した知識が目的データ上の多段推論に十分転移しない場合，データ量増加の利得が現れにくいと考えられる．

4.5.3　実験 3：質問パラフレーズによるデータ増強

図 3 に質問パラフレーズにより実質データ量を増強した結果を示す．この条件では，幅を増やした際の性能低下が緩和され，スケーリングがより安定する傾向が見られた．すなわち，データ量の増加によって，大きいモデルほど性能が改善した．しかし，より精度向上をさせるにはデータの分布を一致させたまま，より多様なデータ量が必要であると考える．

5　結論

本稿では，Graph RAG における GNN 検索器のスケーリング則を検証した．その結果，幅の拡大のみでは性能が一貫して向上せず，最大規模で頭打ちや劣化が生じうることを確認した．一方で，データ量を補う設定ではスケーリングが安定化する傾向が見られ，モデル規模の拡大と同時にデータ量を確保することの重要性が示唆された．

6　今後の展望

今後は，本稿で扱った線形層幅のスケーリングに加えて，新たなアーキテクチャの導入も含めたより効率的なスケール戦略を検討する．また，モデル規模の拡大に見合う学習データを確保するため，転移学習や質問パラフレーズに留まらない，質の高いデータ拡張・学習データのスケーリング手法を確立し，検索精度が向上する GNN 検索器の実現を目指す．

参考文献

[1] Costas Mavromatis and George Karypis. Gnn-rag: Graph neural retrieval for efficient large language model reasoning on knowledge graphs. In **Findings of the Association for Computational Linguistics: ACL 2025**, pp. 16682–16699, Vienna, Austria, 2025. Association for Computational Linguistics.

[2] Costas Mavromatis and George Karypis. Rearev: Adaptive reasoning for question answering over knowledge graphs. In **Findings of the Association for Computational Linguistics: EMNLP 2022**, pp. 2447–2458, Abu Dhabi, United Arab Emirates, 2022. Association for Computational Linguistics.

[3] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B. Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. **arXiv preprint arXiv:2001.08361**, 2020.

[4] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, et al. Training compute-optimal large language models. In **Advances in Neural Information Processing Systems (NeurIPS)**, 2022.

[5] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. **arXiv preprint arXiv:1706.02677**, 2017.