Graph-RAG 中的推理瓶颈：面向多跳问答的结构化提示与上下文压缩

Yasaman Zarrinkia维多利亚大学
yasamanz@uvic.ca

Venkatesh Srinivasan圣克拉拉大学
vsrinivasan4@scu.edu

Alex Thomo维多利亚大学
thomo@uvic.ca

arXiv:2603.14045v2 [cs.IR] 18 Mar 2026

摘要

Graph-RAG 系统通过将文档索引为知识图谱，实现了强大的多跳问答能力，但强大的检索并不保证能够生成高质量的答案。我们在三个多跳问答基准（HotpotQA、MuSiQue、2WikiMultiHopQA）上评估了领先的 Graph-RAG 系统 KET-RAG，发现有 77% 至 91% 的问题在检索上下文中包含标准答案，但准确率仅为 35% 至 78%，且 73% 至 84% 的错误属于推理失败。我们提出了两项增强方法：（i）SPARQL思维链提示，将问题分解为与实体—关系上下文对齐的三元组模式查询；（ii）图遍历压缩，通过知识图谱遍历压缩上下文，且无需调用大语言模型。SPARQL 思维链可将准确率提升 +2 至 +14 个百分点；在较小模型上与结构化提示结合时，图遍历压缩平均可额外提升 +6 个百分点。出人意料的是，我们表明，通过问题类型路由，一个经过完全增强、预算充足的开放权重 Llama-8B 模型，在三个基准测试上均以~12× 更低的成本达到或超过未经增强的 Llama-70B 基线。我们在 LightRAG 上进行的复现实验证实，这些增强方法能够迁移到不同的 Graph-RAG 系统中。

1 引言

多跳问答（QA）需要对散布在多个文档中的多条证据进行推理。基于图的检索增强生成（图检索增强生成）通过从文档语料库构建知识图谱，并利用图结构检索相关上下文来解决这一问题（Edge 等人，2024年；Huang 等人，2025年）。

检索与推理差距。强大的检索并不保证能得到出色的答案。我们采用先进的图检索增强生成系统 KET-RAG（Huang 等人，2025年），结合预算有限的开放权重模型（§4），在三项基准测试中实现了 77% 至 91% 的上下文覆盖率（表2）。然而，Llama-3.1-8B 的整体准确率仅为 23% 至 67%，Llama-3.3-70B 则为 35% 至 78%。问题出在上下文长度：KET-RAG 检索到的上下文通常长达 ~10,000 个词元，其中包含实体描述、关系描述、社区报告摘要和原始文本块。模型必须在如此庞大的上下文中找到相关事实并将它们串联起来，这无异于大海捞针；所需推理跳数越多，挑战就越严峻。将错误分解为检索失败（答案不在上下文中）和推理失败（答案存在，但模型回答错误）后发现，在各数据集中，推理失败占所有错误的 73% 至 84%。由此自然引出一个问题：无需重新训练或重新索引，仅通过对推理阶段进行推理时增强，能否弥合这一差距?

推理增强。为了解决这一瓶颈，我们提出了两种协同策略，无需额外的模型训练。这些增强适用于任何检索到的上下文保留实体关系结构的图检索增强生成系统；我们重点研究 KET-RAG，并验证这些策略也适用于 LightRAG（Guo 等人，2024年），后者是另一种重要的图检索增强生成系统。

1. **SPARQL 思维链提示**（§3.1）：一种结构化的思维链提示，要求模型将问题分解为针对检索到的事实的 SPARQL 查询，从而利用知识图谱上下文固有的实体关系结构。这只需调用一次大语言模型并修改提示；唯一的额外成本是输出稍长。

2.**图遍历上下文压缩**(§3.2)：一种检索后过滤步骤，从问题中的实体出发，在知识图谱中进行广度优先遍历，仅保留结构上相连的上下文。这能减少输入词元 ~60%，且无需调用大语言模型，直接

减轻大海捞针式挑战 .

在三个标准多跳问答基准测试上（Yang 等人，2018年；Trivedi 等人，2022年；Ho 等人，2020年），我们在多种模型规模和配置下进行了评估（§4），主要贡献如下：

1. 定量错误分解表明，瓶颈主要在推理而非检索：当检索上下文中存在标准答案时，仍有 73% 至 84% 的错误发生。

2. SPARQL 思维链始终能提升准确率（+2至 +14 个百分点），且在较难的数据集上提升幅度最大；消融实验证实，分解是大部分收益的来源。

3. 系统分析揭示了图遍历压缩与结构化提示之间的协同效应：模型遵循推理路线图时，压缩带来的帮助最大；小型模型的平均准确率提升了 +6 个百分点。

4. 研究发现，结合结构化提示、图遍历压缩和问题类型路由后，预算有限的 80 亿参数模型能够以 $\sim$12$\times$更低的成本，在全部三个基准测试上达到或超过未经增强的 700 亿参数基线。

5. 普适性检验表明，SPARQL 思维链可迁移至 LightRAG（在 2WikiMHQA 上提升 +7.6 个百分点），而图遍历压缩的效果取决于检索管线结构，而非仅取决于上下文大小。

2 背景：KET-RAG

KET-RAG（Huang 等人，2025年）是一个多粒度索引框架，构建于微软的 GraphRAG（Edge 等人，2024年）之上。它通过结合三种检索策略（**K-E-T**），降低全语料实体抽取的成本：**(K)** 基于所有文本块构建文本关键词二部图，无需调用大语言模型；**(E)** 在通过 PageRank 选出的核心块子集上进行基于大语言模型的实体/关系抽取（β=0.8），生成可通过 GraphRAG 的本地搜索查询的知识图谱；**(T)** 则通过嵌入相似度进行标准文本块检索。参数 $\theta \in [0,1]$ 在总计 λ 词元的预算内平衡实体与关键词通道；我们采用θ=0.5，λ=12，000 (已发布代码的默认值)。

KET-RAG 在多跳问答上取得了最先进的结果，在其评估中优于所有竞争系统（Lewis 等人，2020年；Wang 等人，2024年；Edge 等人，2024年；Sarmah 等人，2024年；Gao 等人，2023年；Gutiérrez 等人，2024年；Guo 等人，2024年）。更广泛地说，KET-RAG 是日益壮大的图检索增强生成系统家族中的一员，其中包括

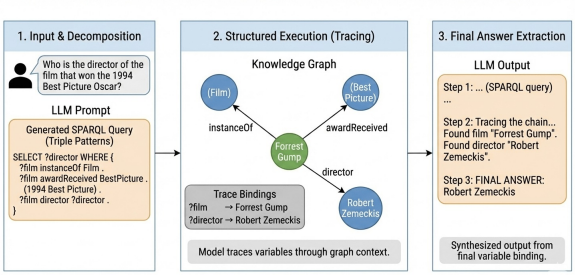


图 1：SPARQL 思维链提示。模型生成一个 SPARQL 查询，其三元组模式与检索上下文相对应，随后追踪变量绑定，并通过一次大语言模型调用生成答案。

LightRAG（Guo 等人，2024年）和 GraphRAG（Edge 等人，2024年），它们的检索上下文保留了显式的实体关系结构（实体描述、关系三元组、社区摘要）。这种结构带来了一个机会：我们不必将上下文视为平铺文本，而可以通过推理增强方法利用其实体关系布局。我们在附录 A.1 中验证了这种方法在 LightRAG 上的普适性。我们将 KET-RAG 的商业组件（GPT-4o-mini、OpenAI 嵌入）替换为经济实惠的开放权重替代方案，且覆盖率没有损失（§5.1），并重点弥合推理差距。

3 方法

3.1 SPARQL 思维链提示

多跳问题要求模型识别并串联分散在大量检索上下文中的多个事实。标准提示让模型隐式地完成这一串联过程，因此常常导致模型遗漏中间跳，或在推理路径不明确时放弃作答。

我们观察到，KET-RAG 的检索上下文天然具有结构化特征：它由实体描述、关系描述（主语-谓语-宾语三元组）、社区报告摘要和文本块组成。这种结构与可使用形式语言进行查询的知识图谱十分相似。我们利用这种对应关系，引入一种思维链（CoT）提示策略，将 SPARQL（Harris 和 Seaborne，2013）——万维网联盟针对 RDF知识图谱制定的标准查询语言——用作推理形式体系。SPARQL 的三元组模式语法（`? subject predicate ?object`）能够自然地表达多跳实体链，因此非常适合将涉及实体-关系上下文的问题分解开来。我们不再要求模型用自然语言“逐步思考”，而是让它

将问题表述为结构化图查询，其中的三元组模式可直接对应上下文中的实体-关系对（图1）。

提示模板如下：

> 步骤 1：编写一个简单的 SPARQL 查询（最多包含 4 个三元组模式，谓词使用通俗英语，不使用 URI、FILTER 或子查询）。示例：SELECT ?answer WHERE { ?x name "Paradise Creek" . ?x tributaryOf ?y . ?y tributaryOf ?answer . } 步骤 2：根据上下文追踪每个三元组模式的变量绑定。步骤 3：写出最终答案： <你的答案>如果答案不在上下文中，写出最终答案：我不知道

这是一次单次大语言模型调用：模型生成 SPARQL 查询，根据上下文追踪查询，并在一次响应中给出答案。与通用的子问题分解相比，其主要优势在于结构对齐：三元组模式（例如 ?x tributaryOf ?y ）与检索上下文中的关系描述直接对应，将对 ∼10,000 个词元进行开放式搜索转化为结构化的模板匹配任务，其中每个变量绑定都会缩小下一跳的搜索空间。唯一额外的成本是输出更长（SPARQL 查询和逐步追踪通常会多出 100–200 个词元）。

通过对模型响应中的 FINAL ANSWER: 进行正则表达式匹配来提取最终答案。如果模型判断答案不在上下文中，可以回答“我不知道”，我们将其计为弃答。

3.2 图遍历上下文压缩

尽管 KET-RAG 实现了较高的覆盖率，但其 ∼10,000 个词元的上下文中，有很大一部分在拓扑结构上远离问题的推理链，从而增加了大海捞针的难度。

我们提出**图遍历上下文压缩（GW）**，这是一种检索后步骤，利用知识图谱自身的拓扑结构，分离出与查询相关的紧凑推理子图，且无需额外的大语言模型调用或嵌入成本。该算法分四个阶段提炼上下文：

阶段 1：种子实体识别。通过多启发式字符串匹配流程，将问题中的实体锚定到检索上下文中：(a) 精确子串匹配（不区分大小写）；(b) 多词匹配，即多词实体的所有实词都出现在问题中；以及 (c) 部分匹配，即问题中的重要词语（≥5个字符）出现在实体名称中；排除常见停用词。

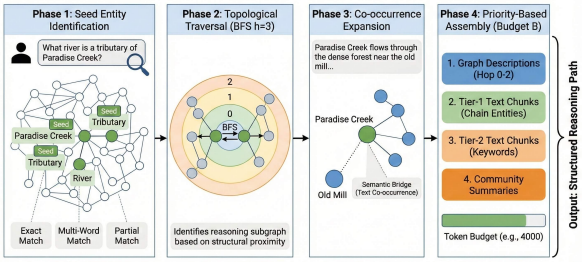


图2：图遍历上下文压缩。从以问题为锚点的种子实体执行广度优先搜索，找出结构上相连的实体，并按跳数层级将其整合为压缩上下文。无需调用大语言模型。

阶段 2：拓扑遍历。从种子实体出发，在知识图谱中执行广度优先搜索（BFS），搜索深度最多为 3 跳。为每个到达的实体标记其与最近种子实体之间的跳数距离，据此建立结构邻近度排序。

阶段 3：共现扩展。通过语义桥接扩展广度优先搜索到达的实体集：如果某个文本块同时提及一个广度优先搜索到达的实体和一个未被广度优先搜索到达的实体，则将后者加入上下文。这样可以找回那些在语义上相关、但在提取图中没有直接连接的实体。

阶段 4：基于优先级的组装。在令牌预算B（默认 B=4,000 个词元）内组装压缩上下文，并按照与问题的结构距离确定优先级：

1. 实体和关系描述，按与种子的跳数距离排序（距离更近 = 优先级更高） 2.一级文本块：提及广度优先搜索到达的实体的文本块，按提及次数排序 3. 二级文本块：包含问题关键词但不含广度优先搜索到达实体的文本块 4. 社区报告摘要（如有，优先级最低）

至关重要的是，输出会按跳数层级重新组织，从而与 SPARQL 思维链 (§3.1) 形成自然衔接：各跳数层级部分与生成的查询中的三元组模式链相对应，使模型能够在局部上下文区域内解析每个变量绑定，而不必扫描整个输入。如果没有匹配到种子实体，该算法将回退到原始的未压缩上下文。

整个过程纯粹是图遍历和

字符串匹配，无需调用大语言模型或进行嵌入计算。实际上，广度优先搜索平均可到达 14–21 个实体，而基于优先级的组装实现了 ∼60 % 的压缩（∼10,000→ ∼4,000 个词元）。图 2 展示了这四个阶段。

4 实验设置

4.1 数据集

我们使用三个标准的多跳问答基准：

• **HotpotQA** （Yang 等人，2018年）：围绕维基百科段落设置的桥接问题和比较问题。每个问题包含 10 个段落（2 个金标准支撑段落 + 8 以及 8 个经过人工整理的干扰项）。桥接问题要求沿着实体链进行推理（例如：“电影 X 的导演出生于哪座城市？”），而比较问题则要求比较属性（例如：“电影 X 和 Y 中哪部更早上映？”）。

• **MuSiQue** （Trivedi 等人，2022年）：要求进行 2–4 个推理步骤的多跳问题，通过将单跳问题组合成推理链构建而成。更深层的推理链（最多 4 跳）是其主要难点来源：每增加一跳，发生推理失败的概率就会累积。每个问题包含 20 个段落（2–4 个金标准支撑段落 + 16 以及 16–18 个从相关问题的金标准段落中抽取的干扰项，因此比随机干扰项更难区分）。通常被认为是三个基准中最难的一个。

• **2WikiMultiHopQA**（Ho 等人，2020年）：基于维基数据构建的问题，要求在两篇维基百科文章之间进行推理；在三个基准测试中，其推理类型范围最广（比较、推理、组合式推理和桥接）。每个问题均配有 10 个段落（2 或 4 个黄金支撑段落 +，以及 6–8 个干扰项），遵循 HotpotQA 的格式。KET-RAG 自身的评估不包含此数据集；我们将其纳入，是因为它涵盖的类型多样。

对于每个数据集，我们使用固定随机种子（42）抽取 $N=500$ 个问题，并汇集所有问题对应的段落，以构建检索语料库。

4.2 模型栈

表 1 展示了我们的模型配置。所有大语言模型调用均使用 Groq 推理平台，该平台以较低的每词元成本为开放权重模型提供快速推理。

索引与模型无关：知识图谱和关键词索引不依赖于问答模型，因此我们可以一次构建，并在无需重新索引的情况下更换问答模型。与 KET-RAG 的默认技术栈（GPT-4o-mini + OpenAI 嵌入）相比，我们使用了 3× 更便宜的大语言模型和免费的嵌入，且维度 4× 更低，每个数据集的总索引成本仅为 ∼1 美元。

表 1：模型栈。索引使用 80 亿参数模型构建一次，并在所有问答配置中重复使用。

组件	模型	Cost
嵌入	MiniLM-L6-v2 (384d)	Free
索引大语言模型	Llama-3.1-8B（Groq）	$0.05/M
问答（预算）	Llama-3.1-8B（Groq）	$0.05/M
问答（标准）	Llama-3.3-70B（Groq）	$0.59/M

在推理时，8B 问答模型的单词元成本低于 70B 模型（每百万词元分别为 $0.05 和 $0.59），这引出了一个问题：增强方法能否缩小准确率差距。

4.3 配置

我们评估四种问答配置，将两种提示策略与可选的图遍历压缩相结合：

1. **基线：** 问答模型接收完整的 KET-RAG 检索上下文（∼10,000 个词元），并直接作答。2. **基线 + 图遍历压缩：**与基线相同，但使用经过图遍历压缩的上下文。3. **SPARQL 思维链：**问答模型接收完整上下文，并使用 SPARQL 思维链提示（§3.1）。4. **SPARQL 思维链 +图遍历压缩：** 将 SPARQL 思维链提示应用于经过图遍历压缩的上下文。

每种配置均分别使用 8B 和 70B 问答模型运行，因此每个数据集运行 8 次，三个基准测试共运行 24 次，总计进行 12,000 次问答评估。所有问答调用的温度均为 0.3。

4.4 评估

我们报告准确率、词元级 F1 和精确匹配（EM），并采用 SQuAD 标准化。准确率通过启发式优先流程确定：先进行规范化字符串匹配（转换为小写、去除冠词和标点，并比较子串及最后一个词元）；对于剩余的不匹配项，则使用大语言模型 Llama-3.1-8B 判断语义等价性作为后备方法（例如，“NYC”与“New York City”）。使用 Claude Opus 4.6 进行的 102 次判断审计显示，其与我们的流程有 96.1% 的一致率。我们还报告

表2：在 GPT-4o-mini Huang 等人，2025年）中，与 KET-RAG 相比的各预算组件上下文覆盖率（%）（+ OpenAI 嵌入。

数据集	Ours	KET 低	KET 高
HotpotQA	90.8 (454/500)	60.2	82.6
MuSiQue	77.2 (386/500)	77.0	79.6
2WikiMHQA	81.0 (405/500)	—	—

弃答率和上下文覆盖率（即标准答案是否出现在检索上下文中，以区分检索失败与推理失败）。所有比较均基于每个数据集中相同的 500 个问题及其检索上下文。

5 结果

5.1 检索质量：覆盖率

采用我们的预算配置（免费的 384 维嵌入，索引大语言模型每百万 token 仅需 0.05 美元），KET-RAG 在三个基准测试中的覆盖率均达到或超过已发表的结果（表 2），尽管 3×索引大语言模型成本更低，且 4× 嵌入维度更低。

5.2 问答性能

表3 展示了全部 24 种配置下的准确率、F1 和精确匹配。

主要结果是：经过 SPARQL 思维链和图遍历压缩增强的低成本 80 亿参数模型，在 2WikiMHQA 上（55.8% 对 48.8%）超越了未增强的 70B 基础模型，并缩小了在 MuSiQue 和 HotpotQA 上的差距，同时每词元成本更低。SPARQL 思维链单独使用时，能提升每一种配置的表现；在 2WikiMHQA 上，80 亿参数模型的准确率最高提升+14.2 个百分点，70B 模型最高提升 +12.2 个百分点。两种增强同时应用时，70B + SPARQL 的准确率总体最高（2WikiMHQA 上为 61.0%，HotpotQA 上为 80.2%）。难度与推理深度的关系符合预期：HotpotQA（两跳，67%–80%）、2WikiMHQA（混合，30%–61%）、MuSiQue（2–4 跳，19%–44%）。

表格右半部分以已覆盖问题（上下文中存在标准答案）为条件，将推理失败与检索缺口区分开来；我们将在第5.7节讨论这一点。下面我们逐一分析各个因素。

5.3 SPARQL 思维链提示的效果

表4 通过与相应的基线（不使用图遍历压缩）进行比较，单独考察 SPARQL 思维链的效果。

准确率的提升与弃答率的降低高度相关：在 MuSiQue 上，8B 模型的弃答率从 52.0% 降至 20.6%（-31.4 个百分点），这表明将关系表示为三元组模式会促使模型参与多跳链，而非选择弃答，从而将开放式推理转化为结构化模式匹配。

5.4 思维链消融实验：SPARQL 与通用分解

我们通过在 2WikiMHQA 上与采用自然语言子问题的通用思维链提示进行比较（表5），消融实验考察收益究竟源于 SPARQL 形式体系，还是源于一般性的分解。

两种思维链方法的表现都显著优于基线，证实分解是主要驱动因素。排名取决于模型规模：80 亿参数模型更倾向于通用思维链（+21.2 个百分点），而非 SPARQL（+14.2 个百分点）；70B 更倾向于 SPARQL（+12.2 个百分点），而非通用方法（+7.6 个百分点）。SPARQL 语法可能给 80 亿参数模型带来额外负担，而 70B 则能利用三元组模式与 KET-RAG 实体关系上下文之间的结构匹配。SPARQL 思维链也能更稳定地降低弃答率（80 亿参数模型为 21.6%，而通用方法为 25.2%；70B 为 30.6%，而通用方法为 36.8%）。值得注意的是，两种思维链方法都提升了 80 亿参数模型的准确率，这与思维链主要有助于更大模型的研究发现形成对照（Wei 等人，2022年）。

一种将桥接问题路由至 SPARQL、将比较／推理问题路由至通用思维链的问题类型路由策略，使 8B 的准确率提升至 58.4%，超过 700 亿参数基线（48.8%）。路由还将 70B 的准确率提升至 66.4%，证实问题类型感知的思维链选择对两种模型规模都有益。

5.5 问题类型路由

由于 8B 偏好的思维链变体取决于问题类型（§5.4），我们在全部三个数据集上采用图遍历压缩应用该路由策略（表6）。

通过路由和图遍历压缩，80 亿参数模型在三个基准测试上的表现均达到或超过未增强的 700 亿参数基线，且成本更低（ $\sim 12\times$ ）。在 2WikiMHQA 上的提升最大（+16.0 个百分点），因为问题类型多样（桥接、比较、推理、组合），路由在此能发挥最大作用。不使用路由时，仅靠 SPARQL+GW 只在 2WikiMHQA 上超过 700 亿参数模型（+7.0 个百分点），但在 HotpotQA（-1.4）和 MuSiQue（-4.6）上仍落后；路由弥补了这些差距。弃答率也降至所有配置中的最低水平：HotpotQA 为 1.0%，

表 3：每个数据集 500 个配对问题上的结果（%）。“已覆盖” = 表示上下文中存在标准答案（括号内为数量）。**粗体** = 表示每列最佳结果。GW = 表示图遍历压缩。F1/精确匹配使用 SQuAD标准化；8B 输出经过大语言模型标准化为简短答案（§4）。

		所有问题								已覆盖问题							
		Hotpot			MuSiQue			2Wiki		Hotpot (454)			MuSiQue (386)			2Wiki (405)	
方法	模型	准确率	F1	精确匹配	准确率	F1	精确匹配	准确率	F1	精确匹配	准确率	F1	精确匹配	准确率	F1	精确匹配	
Base	8B	67.0	59.5	49.4	23.6	18.0	12.4	31.4	29.1	24.2	69.6	62.2	51.5	27.5	21.2	15.8	32.1	30.1	24.7
+GW	8B	63.6	55.9	46.0	19.4	14.6	9.0	30.4	28.1	24.8	66.3	58.3	48.0	22.6	17.0	11.4	33.1	30.8	26.9
SPARQL	8B	70.6	60.8	49.4	28.8	22.6	15.4	45.6	36.9	31.0	74.4	64.2	52.6	34.8	**27.4**	19.7	49.9	40.6	33.6
+GW	8B	**76.6**	**62.2**	**50.6**	**30.6**	**22.9**	**15.8**	**55.8**	**41.1**	**34.8**	**79.5**	**64.9**	**52.9**	**35.1**	26.4	**20.0**	**59.5**	**44.5**	**37.5**
Base	70B	78.0	67.9	56.6	35.2	26.9	18.6	48.8	43.0	36.8	81.3	70.9	59.0	40.3	32.5	23.9	54.8	47.9	40.7
+GW	70B	79.2	68.6	56.0	39.8	30.8	20.6	53.6	49.0	44.0	83.3	71.9	58.6	46.2	37.3	26.2	61.0	56.1	50.4
SPARQL	70B	**80.2**	**70.6**	**57.8**	**43.8**	32.8	23.4	**61.0**	**54.9**	**48.6**	**83.9**	**74.3**	**61.2**	**53.2**	40.4	29.6	**66.9**	**60.0**	**52.6**
+GW	70B	79.6	70.5	**57.8**	42.2	**34.0**	**25.8**	59.8	51.5	44.0	83.5	74.0	60.8	51.4	**42.3**	**32.5**	66.7	57.6	49.6

表4：SPARQL 思维链的效果：准确率 Δ (pp) 和弃答率 Δ (pp)相对于基线的变化，不含图游走压缩。

模型	Hotpot	MuSiQue	2 Wiki
	准确率 Δ *(pp)*		
8B	+3.6	+5.2	+14.2
70B	+2.2	+8.6	+12.2
	弃答 Δ *(pp)*		
8B	−6.4	−31.4	−24.4
70B	+0.4	−8.2	−4.2

表 5：2WikiMHQA 上的思维链消融实验（不使用图游走压缩）。

方法	8B 准确率	8B 弃答率	70B 准确率	70B 弃答率
基线	31.4	46.0	48.8	34.8
通用思维链	52.6	25.2	56.4	36.8
SPARQL 思维链	45.6	21.6	61.0	30.6
路由	**58.4**	**14.2**	**66.4**	**26.8**

MuSiQue 为 6.2%，2WikiMHQA 为 7.2%。路由策略只需额外进行一次成本低廉的分类器调用（几个词元），且每个问题最多重试一次，因此总推理成本远低于单次 700 亿参数模型调用。

实现。 路由器通过一次 8B 调用（最多输出 5 个词元）将每个问题分类为 桥接、比较或推理（确切提示见附录 A.2）。桥接问题使用 SPARQL 思维链；比较和推理问题则使用通用思维链。如果所选方法弃答，则使用另一种方法重试该问题。因此，整个流程每个问题最多需要三次 8B 调用：分类、作答，以及可选的重试。

5.6 图遍历压缩的效果

图遍历压缩与模型能力及提示策略之间呈现出更复杂的交互关系（表7）。

呈现出一致的规律：GW 对 SPARQL + 8B 的帮助最大（平均提升 +6.0 个百分点，在 2WikiMHQA 上提升 +10.2 个百分点）

表6：KET-RAG 上采用路由的 8B 模型 +GW 与未增强的 700 亿参数基线的比较。路由会对每个问题进行分类，采用首选的思维链，并在弃答时使用备用方法。

数据集	准确率（%） 8B	70B	Δ	弃答率（%） 8B	70B	Δ
HotpotQA	**79.8**	78.0	+1.8	**1.0**	**5.6**	−4.6
MuSiQue	**35.6**	35.2	+0.4	**6.2**	16.2	−10.0
2WikiMHQA	**64.8**	48.8	**+16.0**	**7.2**	34.8	**−27.6**

表7：图遍历压缩的效果：相较于非图遍历配置的准确率 Δ（百分点）。

方法	模型	Hot.	Mus.	2Wi.	Avg
Base	8B	−3.4	−4.2	−1.0	−2.9
Base	70B	+1.2	+4.6	+4.8	+3.5
SPARQL	8B	+6.0	+1.8	+10.2	+6.0
SPARQL	70B	−0.6	−1.6	−1.2	−1.1

；对 Base + 70B 也有帮助（平均提升 +3.5 个百分点），但会降低 Base + 8B 的表现（-2.9 个百分点）。对于 SPARQL + 70B，GW 平均仅造成 -1.1 个百分点的损失，同时将输入词元减少 ∼60%（∼10,000 → ∼4,000），考虑到 70B 模型较高的单词元价格，这种准确率与成本之间的权衡颇为有利。这揭示了二者的互补性：当模型获得结构化指导（SPARQL + 8B），或具备足够能力在降低噪声后进行推理（700 亿参数基线）时，GW 能发挥作用；若两者皆无，小模型便依赖穷尽式上下文进行模式匹配，而压缩会移除有用信号。

5.7 错误分解

为区分检索失败与推理失败，我们根据检索上下文中是否包含标准答案来分解错误。表8 展示了推理错误占总错误的比例。表 3中的“已覆盖”列报告了所有配置下以覆盖率为条件的准确率、F1 和精确匹配。

推理失败占主导：例如，在 MuSiQue 上，77.2% 的问题的答案都存在于上下文中

表 8：推理错误占总错误的比例（%）：标准答案已存在于上下文中但答案错误的情况。

配置	Hotpot	MuSiQue	2Wiki
8B 基线	84	73	80
70B + SPARQL 思维链	74	64	69

（表 2）但 8B 基线的整体正确率仅为 23.6%，相差逾 50 个百分点，这归因于推理失败。采用 70B 的 SPARQL 思维链将这一差距缩小了约一半（已覆盖问题的正确率为 53.2%），这证实了图检索增强生成问答中的主要瓶颈在于推理，而非检索。

在 HotpotQA 上，70B + SPARQL 思维链在已覆盖问题上的得分达到 83.9%，接近 90.8% 的覆盖率上限，这表明对于两跳问题，剩余错误大多源于检索缺口。

5.8 按问题类型分析

每个基准数据集都会标注问题的推理类型。表 9 按类型细分了 SPARQL 思维链带来的提升；表 10 展示了 GW 压缩对不同类型问题的影响。

提升最大的是沿链追踪类型的问题：在 2WikiMHQA 上，桥接_比较问题的得分分别提升了 +18.2 个百分点（8B）和 +28.1 个百分点（70B），而需要隐式推理的推理问题提升最小（+3.7 个百分点）。这与 SPARQL 思维链的设计相符：三元组模式分解能自然地形式化实体链，因此当问题结构符合这种形式化方式时，效果最佳。在 MuSiQue 上，70B 模型在所有跳数深度上都从 SPARQL 思维链中显著受益（两跳时提升+9.4，三跳时提升 +5.5，四跳时提升 +11.1），其中四跳问题的提升最大。

GW 压缩同样有利于更深的跳数：在 MuSiQue 上，GW 在两跳时略有负面影响，但在三跳时（基础模型+70B 为 +11.7）和四跳时（+10.0）效果显著为正。在 2WikiMHQA 上，GW 对推理（SPARQL+8B 提升 +16.7 个百分点）以及桥接比较问题（基础模型+70B 提升 +17.4 个百分点）的帮助最大，因为跨多个实体整合信息最能受益于聚焦的上下文。

6 讨论

瓶颈在于推理，而非检索。有趣的是，即使使用预算有限的组件（§5.1），KET-RAG 的覆盖率仍达到 77% 至 91%，但准确率却远远落后：覆盖率与准确率之间的差距在 MuSiQue 上超过 50 个百分点（§5.7）。由于图索引与模型无关，我们可以先用低成本模型构建索引，再逐步升级问答组件，而无需重新索引。

结构化分解是主要的杠杆。我们的关键发现表明，SPARQLCoT 能提升我们测试过的每种配置（§3.1），包括 80 亿参数模型；这与思维链主要使参数量超过 ∼100B 的模型受益这一发现形成对照（Wei 等人，2022年）。消融实验（§5.4）表明，收益来自分解本身；SPARQL 形式化体系还能带来额外价值，70B 模型可以利用与图上下文的结构对齐。

小模型能做到什么程度？ 8B + SPARQL 思维链 +在 2WikiMHQA 上超过未经增强的 70B 基础模型（55.8% 对 48.8%），且 ∼12×每词元成本更低。然而，在 MuSiQue 上，70B 仍有 +4.6 个百分点的优势，这表明更深层的推理链（3–4 跳）需要更大的模型，而两跳问题即使使用小模型，也能通过有效搭建推理支架来处理。问题类型路由（§5.5）进一步提升了效果：将每个问题引导至合适的思维链变体，并在弃答时重试后，全面增强的 8B 在全部三个基准测试上均超过未经增强的 70B。

在 KET-RAG 之外的普适性。为检验这些结果是否仅适用于特定系统，我们在三个基准测试上对 LightRAG（Guo 等人，2024年）复现了实验（附录 A.1）。SPARQL 思维链始终能提升 LightRAG 的表现：在 2WikiMHQA 上提高 +7.6 个百分点，+5在 MuSiQue 上提高 .4 个百分点，在 +2 HotpotQA 上提高 .2 个百分点，证实只要上下文包含实体—关系结构，结构化推理就能迁移。不过，图遍历压缩在 LightRAG 的三个数据集上均会损害表现（-8.6/-6.6/-11.0 个百分点），尽管上下文大小相近（中位数分别为 ∼13,000 对比 ∼10,600 个词元）。差异源于结构：KET-RAG 的 β分层检索形成了相关性梯度，图遍历能够加以利用；而 LightRAG 的混合检索返回的材料相关性均匀，剪枝会不加区分地将其削弱。这表明 SPARQL 思维链是一种与系统无关的提示策略，而 GW 压缩要发挥作用，则需要检索管线具备相关性梯度。

部署建议。结果表明，可以采用一条简单的决策规则。对于预算有限的部署（8B），始终启用带图遍历

表 9：按问题类型划分的 SPARQL 思维链准确率相较于基线的提升（百分点）。HotpotQA：桥接（398）和比较（102）。2WikiMHQA：组合式（205）、桥接_比较（121）、比较（120）、推理（54）。MuSiQue：按跳数深度划分（264/146/90）。

模型	HotpotQA		2WikiMHQA				MuSiQue		
	Bri.	比较	比较。 基础模型比较。		Cmp.	Inf.	2h	3h	4h
8B	+3.0	+5.9	+16.6	+18.2	+10.9	+3.7	+8.7	−0.6	+4.5
70B	+2.3	+2.0	+4.9	**+28.1**	+12.5	+3.7	+9.4	+5.5	+11.1

表10：按问题类型／跳数深度划分的图游走压缩准确率 Δ (pp)。括号内为计数。

数据集	类别	SPARQL+8B 基础模型+70B	
Hotpot	桥接（398）	+4.3	+1.2
	比较（102）	+12.8	+1.0
2 Wiki	组合（205）	+5.8	−1.0
	桥接-比较（121）	+11.6	+17.4
	比较 (120)	+13.3	+2.5
	推理 (54)	+16.7	+3.7
MuSiQue	两跳 (264)	−1.9	−1.2
	三跳 (146)	+6.8	+11.7
	四跳 (90)	+4.4	+10.0

压缩：它以 ∼12×更低的成本达到或超过 70B 的表现，且额外开销可以忽略不计（一次分类器调用，外加最多一次回退）。对于追求最高质量的部署（70B），使用不带压缩的 SPARQL 思维链；在这一规模下，GW 的 ∼1 个百分点准确率代价通常不足以抵消节省的令牌。只有对于基础模型 + 8B 才应避免使用 GW，因为小模型缺乏引导其在压缩上下文中进行导航的推理能力。

7 相关工作

图检索增强生成系统。 图检索增强生成系统会从语料库中构建知识图谱，并利用图结构指导检索。一个关键区别在于检索上下文是否保留实体—关系（E-R）结构：GraphRAG（Edge 等人，2024年）、KET-RAG（Huang 等人，2025年）和 LightRAG（Guo 等人，2024年）会返回实体描述、关系三元组和文本块，因此其上下文适用于结构化提示和基于图的压缩。HybridRAG（Sarmah 等人，2024年）也会将知识图谱子图与向量检索段落相结合，不过目前仅在金融业绩电话会议文字稿上进行过评估。

其他系统利用图结构进行检索，但返回的是扁平段落：HippoRAG（Gutiérrez 等人，2024年，2025年）、KGP（Wang 等人，2024年）、Clue-RAG（Su 等人，2025年）和 GraphRAG-V（Yu 等人，2025年）。Xiang 等人（2025年）指出，上下文膨胀是一个关键失效模式。

以上所有方法都旨在提升检索质量；我们研究使用预算有限的模型对检索到的 E-R 上下文进行推理。

思维链提示。思维链（Wei 等人，2022年）及其结构化变体（思维程序（Chen 等人）、分解式提示（Khot 等人，2023年）通过显式呈现中间步骤来提升推理能力。

IRCoT（Trivedi 等人，2023年）和Self-Ask（Press 等人，2023年）等多轮调用方法将推理与检索交替进行；这些方法与我们的单次调用设定相互独立，在该设定中，图检索增强生成已提供完整的实体关系上下文。

SPARQL 已用于知识库问答（Gu 等人，2023年），但从未被用作图检索增强生成的思维链脚手架。

上下文压缩。LLMLingua（Jiang 等人，2023年）和 RECOMP（Xu 等人，2023年）通过学习模型压缩段落，但会产生推理成本。图检索增强生成的上下文具有可加以利用的拓扑结构，因此无需大语言模型即可通过广度优先搜索进行压缩。此前尚未研究压缩与结构化提示之间的交互。

8 结论

当检索达到较高覆盖率后，图检索增强生成问答的主要瓶颈便是推理。我们提出三种针对这一差距的推理时增强：SPARQL 思维链（+2至 +14个百分点）、图遍历压缩和问题类型路由；这些方法均无需重新训练或重新索引。三者结合后，经过全面增强的 80 亿参数模型在所有三项基准测试上的表现均可媲美或超越未经增强的 700 亿参数基线，成本则降低 ∼12× 。由于图索引与模型无关，因此可直接换用更强的未来模型，无需重新索引。对于预算受限部署，采用图遍历压缩的路由方案只需极低成本，即可达到或超越 700 亿参数模型的质量；对于追求质量最大化的部署，仅使用 SPARQL 思维链即可。

代码和数据可在 https://github.com/thomouvic/graph-rag-qa-pub获取。

局限性

我们的研究存在一些局限性。首先，我们仅使用单一嵌入模型和两种大语言模型规模进行评估；在模型规模和嵌入维度上进行更细粒度的遍历，将有助于更好地刻画退化曲线。其次，我们的上下文覆盖率指标采用子串匹配，这可能导致过度计数（偶然匹配）或计数不足（释义后的答案）。第三，对于 KET-RAG 的实体-关键词平衡，我们使用默认$\theta = 0.5$，而针对每个数据集调节 θ 可能会改善结果。最后，尽管大语言模型评判的评估比精确匹配更加灵活，但可能会引入评估噪声；我们通过较大的样本量（N=500）来减轻这一影响。我们使用人工智能编程助手（Claude Code）协助进行实验脚本编写和论文编辑；所有科学贡献均由作者完成。

参考文献

Wenhu Chen、Xueguang Ma、Xinyi Wang 和 William W Cohen。思维程序提示：为数值推理任务解耦计算与推理。《机器学习研究汇刊》。

Darren Edge、Ha Trinh、Newman Cheng、Joshua Bradley、Alex Chao、Apurva Mody、Steven Truitt、Dasha Metropolitansky、Robert Osazuwa Ness 和 Jonathan Larson。2024年。从局部到全局：一种面向查询导向摘要的图检索增强生成方法。*arXiv* 预印本 *arXiv:2404.16130*。

Luyu Gao、Xueguang Ma、Jimmy Lin 和 Jamie Callan。2023年。无需相关性标签的精准零样本稠密检索。载于计算语言学协会第*61*届年会论文集（第*1*卷：长篇论文），第1762–1777页。

Yu Gu、Xiang Deng 和 Yu Su。2023年。不要生成，而要判别：一项旨在将语言模型落地到真实世界环境的提案。载于计算语言学协会第*61*届年会论文集（第*1*卷：长篇论文），第4928–4949页。

Zirui Guo、Lianghao Xia、Yanhua Yu、Tu Ao 和 Chao Huang。2024 年。LightRAG：简洁快速的检索增强生成。*arXiv* 预印本 *arXiv:2410.05779*。

Bernal J Gutiérrez、Yiheng Shu、Yu Gu、Michihiro Yasunaga 和 Yu Su。2024年。HippoRAG：受神经生物学启发的大语言模型长期记忆。神经信息处理系统进展，37：59532–59569。

Bernal Jiménez Gutiérrez、Yiheng Shu、Weijian Qi、Sizhe Zhou 和 Yu Su。2025年。从检索增强生成到记忆：面向大语言模型的非参数持续学习。发表于国际机器学习大会，第 21497–21515 页。PMLR。

Steve Harris 和 Andy Seaborne。2013。SPARQL 1.1 查询语言。万维网联盟推荐标准。

Xanh Ho、Anh-Khoa Duong Nguyen、Saku Sugawara 和 Akiko Aizawa。2020年。构建用于全面评估推理步骤的多跳问答数据集。发表于第 *28* 届计算语言学国际会议论文集，第 6609–6625 页。

Yiqian Huang、Shiqi Zhang 和 Xiaokui Xiao。2025年。KET-RAG：面向图检索增强生成的高性价比多粒度索引框架。发表于 第 *31* 届 *ACM SIGKDD*知识发现与数据挖掘大会论文集第 *2* 卷，第 1003–1012 页。

Huiqiang Jiang、Qianhui Wu、Chin-Yew Lin、Yuqing Yang 和 Lili Qiu。2023年。LLMLingua：压缩提示词以加速大语言模型的推理。载于*2023*年自然语言处理实证方法会议论文集，第13358–13376页。

Tushar Khot、Harsh Trivedi、Matthew Finlayson、Yao Fu、Kyle Richardson、Peter Clark 和 Ashish Sabharwal。2023年。分解式提示：一种解决复杂任务的模块化方法。载于第十一届国际学习表征会议。

Patrick Lewis、Ethan Perez、Aleksandra Piktus、Fabio Petroni、Vladimir Karpukhin、Naman Goyal、Hein- rich Küttler、Mike Lewis、Wen-tau Yih、Tim Rock- täschel等人。2020年。面向知识密集型自然语言处理任务的检索增强生成。神经信息处理系统进展，33：9459–9474。

Ofir Press、Muru Zhang、Sewon Min、Ludwig Schmidt、Noah A Smith和Mike Lewis。2023年。衡量并缩小语言模型中的组合性差距。载于计算语言学协会论文集：*EMNLP 2023*年，第5687–5711页。

Bhaskarjit Sarmah、Dhagash Mehta、Benika Hall、Ro- han Rao、Sunil Patel和Stefano Pasquali。2024年。HybridRAG：融合知识图谱与向量检索增强生成，实现高效信息抽取。载于第五届*ACM*人工智能金融国际会议论文集，第608–616页。

Yaodong Su、Yixiang Fang、Yingli Zhou、Quanqing Xu 和 Chuanhui Yang。2025年。Clue-RAG：通过多部图和查询驱动的迭代检索，实现准确且经济高效的基于图的检索增强生成。*arXiv* 预印本 *arXiv:2507.08445*。Harsh Trivedi、Niranjan Balasubramanian、Tushar Khot 和 Ashish Sabharwal。2022年。MuSiQue：通过单跳问题组合构造多跳问题。计算语言学协会汇刊，10：539–554。

Harsh Trivedi、Niranjan Balasubramanian、Tushar Khot 和 Ashish Sabharwal。2023年。将检索与思维链推理交替进行，以应对知识密集型多步问题。载于第*61*届计算语言学协会年会论文集（第一卷：长篇论文），第10014–10037页。

Yu Wang、Nedim Lipka、Ryan A Rossi、AlexaSiu、Ruiyi Zhang 和 Tyler Derr。2024年。用于多文档问答的知识图谱提示。载于*AAAI*人工智能会议论文集，第38卷，第19206–19214页。

Jason Wei、Xuezhi Wang、Dale Schuurmans、Maarten Bosma、Fei Xia、Ed Chi、Quoc V Le、Denny Zhou 和另外1位作者。2022年。思维链提示能够激发大语言模型的推理能力。神经信息处理系统进展，35：24824–24837。

Zhishang Xiang、Chuanjie Wu、Qinggang Zhang、Shengyuan Chen、Zijin Hong、Xiao Huang 和 Jin-song Su。2025年。何时在检索增强生成中使用图：对图检索增强生成的全面分析。*arXiv*预印本 *arXiv:2506.05690*。

Fangyuan Xu、Weijia Shi 和 Eunsol Choi。2023年。Re-comp：通过压缩和选择性增强改进检索增强语言模型。*arXiv* 预印本 *arXiv:2310.04408*。

Zhilin Yang、Peng Qi、Saizheng Zhang、Yoshua Bengio、William Cohen、Ruslan Salakhutdinov 和 Christopher D Manning。2018年。HotpotQA：一个用于多样化、可解释的多跳问答的数据集。载于*2018*年自然语言处理实证方法会议论文集，第 2369–2380 页。

Tengkai Yu、Venkatesh Srinivasan 和 Alex Thomo。2025年。GraphRAG-V：通过文本块社区实现快速多跳检索。载于社交网络分析与挖掘国际会议，第 3–14 页。Springer。

A 附录

本附录提供了正文中提及的补充分析和实验细节。§A.1 介绍了完整的 LightRAG 通用性实验，该实验已在 §6中概述。§A.2 给出了§5.5中使用的路由分类器提示。

A.1 通用性：LightRAG实验

为了测试 SPARQL 思维链能否推广到 KET-RAG 之外，我们在 LightRAG（Guo 等人，2024年）上复现了完整实验。LightRAG 是一种颇具影响力的开源图检索增强生成系统，实验覆盖全部三个基准数据集。我们采用与 KET-RAG 设置相同的模型（使用 Llama-3.1-8B 建立索引，问答使用 8B 和 70B 两种模型），并为 500 个问题分别检索混合（关键词 + 向量）上下文，然后运行全部八种问答配置（4 种方法 $\times$ 2模型）。LightRAG 在 HotpotQA、MuSiQue 和 2WikiMHQA 上的上下文覆盖率分别为 450/500、374/500 和 417/500。

表 11与表 3 直接比较。SPARQL 思维链持续提升所有配置的表现，对 8B 模型的提升更大，这证实了结构化分解对较小模型的益处尤为显著。整体提升小于在 KET-RAG 上的提升，这可能是因为 LightRAG 在抽取过程中会将实体聚合成高层描述，因此三元组模式分解与检索上下文的匹配度较低。

相比之下，图遍历压缩会降低LightRAG 在所有配置下的表现，与 KET-RAG 上的结果相反。由于这两个系统检索到的上下文规模相近（中位数分别为 $\sim$13，000 和 $\sim$10，600 个词元），差异并非源于上下文长度，而是上下文结构不同：KET-RAG 的检索会将核心块（β=0.8）与外围块区分开来，形成相关性梯度，供图遍历利用。相比之下，LightRAG 的混合检索返回的材料没有这种分层，因此剪枝时相关内容和噪声同样容易被移除。

这些结果支持了我们的核心主张：SPARQL 思维链是一种与系统无关的提示策略；只要上下文包含实体关系结构，无论内容由哪个图检索增强生成系统生成，它都能提升多跳问答效果。图游走压缩的发现补充了一项实践层面的注意事项：只有当检索流程形成了清晰的相关性梯度，供图遍历加以利用时，压缩才最为有益，而非仅由上下文大小决定。

表11：LightRAG 在每个数据集的 500 个配对问题上的通用性实验（%）。“已覆盖” = 表示 LightRAG上下文中存在标准答案（括号内为数量）。**加粗** = 表示每个模型组中的最佳结果。GW = 表示图遍历压缩。F1/精确匹配采用SQuAD标准化；8B 的输出经过大语言模型标准化为简短答案（§4）。

		所有问题									已覆盖问题								
		Hotpot			MuSiQue			2Wiki			Hotpot（450）			MuSiQue（374）			2Wiki（417）		
方法	**模型**	准确率	F1	精确匹配	准确率	F1	精确匹配	准确率	F1	精确匹配	准确率	F1	精确匹配	准确率	F1	精确匹配	准确率	F1	精确匹配
Base	8B	62.2	55.1	45.2	26.6	19.3	12.8	32.2	29.0	24.6	65.8	57.8	47.3	34.2	23.7	16.6	34.5	31.2	26.6
+GW	8B	54.4	47.8	38.4	25.4	18.1	11.8	35.2	31.9	27.6	56.0	49.1	39.1	30.7	22.0	15.2	35.7	32.2	27.6
SPARQL	8B	**66.8**	**57.5**	**46.6**	**37.4**	**26.9**	**18.6**	**57.4**	44.3	36.2	**69.6**	**59.3**	**47.8**	**46.3**	**33.0**	**24.1**	**61.9**	**47.1**	38.8
+GW	8B	66.2	55.2	44.4	33.6	23.5	15.8	56.4	**44.6**	**37.8**	69.3	57.0	45.6	40.1	28.1	20.3	60.2	46.6	**39.8**
Base	70B	79.8	68.1	55.6	41.6	32.0	24.2	59.8	49.1	41.8	83.3	70.7	57.8	52.1	40.2	31.8	64.5	52.1	44.8
+GW	70B	68.8	59.9	49.4	35.0	25.7	17.2	51.2	45.8	39.2	71.8	62.2	51.1	44.4	32.5	23.0	54.9	49.0	42.0
SPARQL	70B	**82.0**	**71.6**	**57.8**	**47.0**	**36.5**	**28.0**	**67.4**	**58.1**	**49.0**	**85.8**	**74.2**	**59.8**	**58.8**	**46.6**	**36.9**	**72.4**	**62.7**	**52.8**
+GW	70B	64.8	56.5	45.4	36.6	29.7	23.6	50.2	45.0	37.6	67.3	58.2	46.4	47.1	38.4	31.3	53.7	47.8	40.0

A.2 路由分类器提示

问题类型分类器（§5.5）使用以下提示词，并采用相同的 8B 模型（最多生成 5 个词元）：

> 将此问题准确分类为以下类别之一：“桥接”“比较”或“推理”。定义：- 桥接：答案需要沿着事实中的实体链进行追踪 - 比较：答案需要比较两个实体或值 - 推理：答案需要进行隐式推理，而不是沿着清晰的实体链推理 请仅回复一个词：bridge、comparison 或 inference 问题：{question}

如果模型返回的标签不属于 {bridge, comparison, inference}，系统将默认选择桥接。