iAgentBench：高流量主题中信息寻求智能体意义建构能力的基准测试

Preetam Prabhu Srikar Dammu
华盛顿大学 西雅图，华盛顿州，美国 preetams@uw.edu

Arnav Palkhiwala
华盛顿大学 西雅图，华盛顿州，美国
apalkh@uw.edu

Tanya Roosta 加州大学伯克利分校 伯克利，加利福尼亚州，美国
troosta@ischool.berkeley.edu

Chirag Shah 华盛顿大学 西雅图，华盛顿州，美国 chirags@uw.edu

arXiv:2603.04656v1 [cs.CL] 4 Mar 2026

摘要

随着支持搜索的生成式问答系统兴起，用户越来越多地借助这类工具代为浏览、汇总并协调来自多个来源的证据。然而，许多广泛使用的问答基准仍可通过检索单个相关段落来作答，因此并不适合衡量跨来源信息综合能力，例如整合证据、追踪因果联系，以及厘清主题不同方面之间的依赖关系。我们推出 iAgentBench，这是一项动态开放域问答基准，旨在满足这些更高层次的信息需求，同时确保问题自然，并植根于真实的信息寻求行为。iAgentBench 从现实世界关注度信号中选取种子主题，并利用常见的用户意图模式构造类用户问题；回答这些问题需要结合多个来源的证据，而非仅仅提取单条内容。每个实例均附有可追溯证据和可审计的中间产物，以支持污染检查，并能细致诊断检索与综合环节各自的失误。针对多个大语言模型开展的实验表明，检索能够提高准确率，但仅靠检索无法可靠地解答这些问题，这凸显了评估证据运用能力的必要性，而不应只评估证据获取能力。

CCS 概念

• **信息系统**;

关键词

数据污染、检索增强生成（RAG）、问答、动态基准测试

ACM 参考格式：
Preetam Prabhu Srikar Dammu、Arnav Palkhiwala、Tanya Roosta 和 Chi-rag Shah。2026年。iAgentBench：针对高流量主题对信息寻求智能体的意义建构能力进行基准测试。发表于。ACM，纽约州纽约市，美国，9 页。https://doi.org/10.1145/nnnnnnn.nnnnnnn

允许免费制作本作品全部或部分内容的数字或纸质副本，供个人或课堂使用，前提是不得为营利或商业利益而制作或分发副本，且副本首页须附有本声明和完整引文。对于本作品中由作者以外人士拥有版权的内容，必须尊重其版权。注明出处后可以摘录。若要以其他方式复制、再版、发布到服务器或分发至列表，则须事先取得具体许可和/或支付费用。请通过 permissions@acm.org 申请许可。*Conference'17*，美国华盛顿特区© 2026年 版权所有：所有者/作者。出版权已授权给 ACM。ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 引言

当用户需要的不只是快速查询时，信息寻求智能体（ISAs）的应用日益广泛。在这类场景中，系统必须跨来源搜索、整合不完整的证据，并生成简洁的综合结果，以支持后续决策。如果答案包含在单个段落中，强大的检索加生成通常就足够了。如果答案只有在整合多个来源后才能得出，系统的可靠性便取决于其积累、组织和利用证据的能力。

然而，评估仍然过度侧重于那些归结为寻找一个相关段落的问题。许多广泛使用的问答基准都是围绕以段落为中心的问答设计的，在这种设置中，即便存在对抗性扰动 [13]，成功与否仍主要取决于能否匹配并提取出正确的文本片段或句子。这种设置只是信息寻求智能体所需跨来源理解能力的一个薄弱代理。它无法充分衡量以下失败模式：证据分散在多个文档中、必须综合考虑多个主题，或正确回答取决于这些主题之间的关系。

一种自然的应对方式是转向更具挑战性的问答设置，尤其是多跳问答。然而，不应将多跳推理与意义建构混为一谈；意义建构所需的能力不止于串联跨多个跳步的证据。HotpotQA 等数据集往往仍然侧重于奖励对少量词汇匹配项或简短佐证陈述的定位 [33]。即便涉及多个文档，许多多跳问题的设计仍使解题过程主要依赖沿路径查找和拼接片段。

Edge等人[4], 所定义的意义建构问题有助于明确当前缺失的环节。Edge等人 [4] 认为，传统检索增强生成难以处理那些需要对整个文本集合进行查询导向摘要的查询，而非仅仅检索几条匹配记录。以数据集为层面的这一框架很有价值，但其针对的情境与大多数开放域信息寻求智能体所面临的情境不同。在实际应用中，相关上下文很少是固定的私有语料库，而是从庞大且不断变化的网络中按查询条件选取的切片；阅读搜索结果的前几页，往往就是回答用户问题时自然的范围。要在这一情境下评估智能体，就需要一些基准测试，重点考察智能体如何对已检索来源本身进行意义建构，而不是如何对整个集合进行摘要。

我们提出 iAgentBench，即信息检索代理基准，这是一个动态基准，旨在评估开放域问答中的已检索来源级意义建构 。每个实例都以种子

来自流量驱动、时间索引的主题，因此能够反映用户当时积极搜索的内容，并以开放网络证据的查询条件化切片为基础。问题通过意图模式实例化，以保持自然且独立；而故事图表示则明确呈现主题和连接关系，因此回答问题需要整合多个主题的证据，而非提取单个段落或进行浅层的多跳拼接。

iAgentBench 旨在支持随着网络变化进行重复评估，并通过区分证据获取失败与证据整合失败，实现超越总体准确率的分析。具体而言，iAgentBench 具有以下基准层面的特征：

(1) 以流量驱动、按时间索引的主题 为基础，并扎根于现实世界关注度信号。(2) 由查询限定的开放网络语料库 ，反映信息寻求智能体在测试时会阅读的有限证据范围。(3) 跨主题、依赖连接关系的问题 ，要求必须依赖多个主题及其关系（而非查找单个段落或表面拼接多跳信息）。(4) 反映用户信息需求的问题 ，通过实例化常见意图模式，使评估测试超越通用事实查找，考查意义建构目标。(5) 随时间窗口动态重新生成 ，以支持基于最新证据进行重复评估，并减少对记忆的依赖。

我们在 Hugging Face 上提供 iAgentBench[1], 在 GitHub 上提供源代码[2], 并在项目网站上提供其他资源[3]，以支持在动态开放网络环境下评估信息寻求智能体的研究。

2 相关工作

开放域问答（ODQA）基准在问题反映真实用户意图的程度上存在显著差异。SQuAD [24] 等阅读理解数据集有助于规范答案片段提取，而后续的开放域问答资源则逐步转向更贴近真实网络信息需求的问题。Natural Questions [18] 以真实搜索查询为基础，而（QA）[2] [17]则凸显了自然产生的问题可能包含值得质疑的假设。与此同时，许多广泛使用的开放域问答数据集受到基准测试便利性或作者编制限制（例如以琐事问答为导向的来源）的影响，导致其问题分布偏离用户实际提出的问题。在这项工作中，iAgentBench采用兴趣驱动的设计：种子主题取自现实世界关注度信号（通过 GDELT [30]），而非经过筛选的知识库或问答题集。

开放域问答的一项核心挑战在于，证据往往分散在多个文档中，而且相关证据在语义上未必与问题本身相似。HotpotQA [33] 推动了多跳监督的普及，而 FiD [11] 等高度依赖检索的阅读器则展示了聚合多个段落所带来的收益。检索器的开发同样侧重于提升覆盖率和组合能力，包括稠密检索 [14] 和检索增强语言模型 [8, 12, 19]。这些基准测试和系统对于研究检索质量和多跳组合至关重要。相比之下，iAgentBench 会围绕查询条件化网络语料库设计问题，要求进行跨主题综合；其瓶颈往往并非缺少某一个段落，而是如何整合分散在不同主题及其明确关联中的证据。

图结构已被用于支持多步检索，并使跨文档推理更加明确。早期研究在维基百科图上检索推理路径以进行多跳问答[1], , 这体现了在词汇重叠有限时结构化遍历的价值。近年来，GraphRAG 风格的 [4] 方法提出从语料库构建中间图表示，以支持更高层次的综合，通常依赖 Leiden 等社区发现方法 [31]。尤其是“由局部到全局”的查询聚焦摘要指出，与纯粹的局部段落匹配相比，某些问题更适合借助相关语料库的全局结构来回答 [4]。iAgentBench 建立在这些洞见之上，但针对的是不同场景：我们并非在固定语料库上进行数据集层面的意义建构，而是基于网络检索到的证据构建查询条件化 故事图，并生成那些隐含地要求整合多个主题及其明确的主题间连接关系的开放域问答对。

随着问答系统日益整合各种工具（搜索、浏览和迭代规划），评估必须着重考察单次检索之外的行为。WebGPT [23]和 ReAct [34]展示了将证据收集与生成交替进行的系统。这一趋势也加剧了长期存在的基准测试问题：静态问答数据集很快就会过时，随着模型训练语料库不断扩大，也越来越容易受到记忆和数据泄漏的影响 [2, 20, 25]。因此，近期研究着重发展时间敏感评估 [15, 21] 和动态基准测试框架 [3, 16, 22, 26, 35, 37]。在检索增强设置中，搜索时污染是另一种失效模式：智能体可以直接检索基准内容并照搬答案 [9]。因此，iAgentBench 旨在实现可更新和可审计：它以时间索引、兴趣驱动的主题信号为基础，并生成证据材料（已检索文档、故事图链接和注释），以支持污染检查和定性错误分析。

3 方法

iAgentBench 是一个动态、开放域问答基准构建流程，旨在评估信息寻求智能体的跨文档意义建构能力。其核心思路是从真实、由流量驱动的主题出发，从网络中检索查询条件化语料库，构建紧凑的结构化表示，以呈现语料库中各主题之间的关系，然后生成答案依赖于多个主题及其明确关联的问题。图 1 总结了这一流程。

3.1 兴趣驱动种子

iAgentBench 的一个核心目标是支持用户可能真正关心的问答，而不是由固定来源数据库的特征（例如知识库模式或经过筛选的趣味知识列表）所诱发的问题。因此，我们根据公众关注信号推导种子主题，并为其添加时间索引，以支持动态评估。

来自 *GDELT* 的种子候选.. 我们使用 GDELT 1.0全球知识图谱（GKG）的每日快照 [30] 作为

[1]https://huggingface.co/datasets/preetam7/iAgentBench [2] https://github.com/iagentbench/iAB.git [3] https://iagentbench.github.io/iAgentBench/

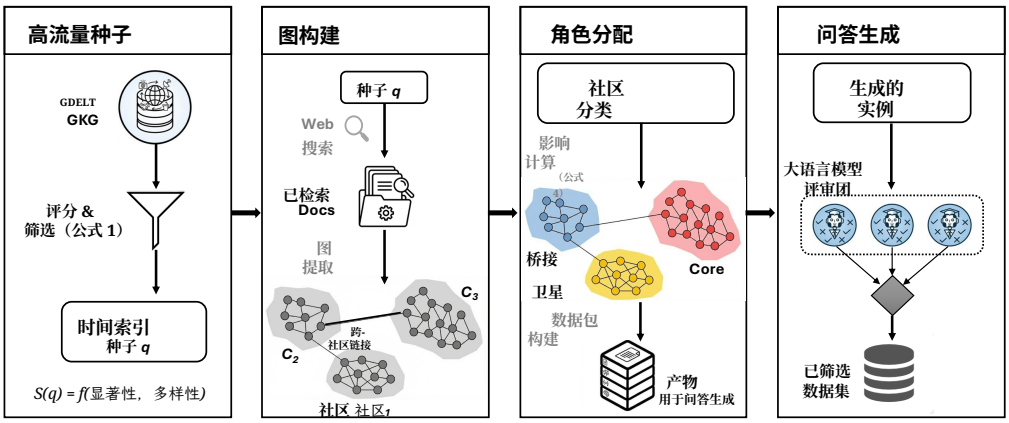

图1：iAgentBench构建流程概览。我们（1）从公众关注信号（GDELT）中采样时间索引的高流量种子查询，（2）检索查询条件化网络语料库，并抽取带有主题社区的类主张故事图，（3）分配社区角色（核心/桥接/卫星），并构建保留跨主题链接的紧凑产物（社区卡片、连接器、信息包），以及（4）使用大语言模型评审团生成并筛选独立的开放域问答对。

衡量哪些事件和实体受到关注的替代指标。每天都会通过将显著实体与事件描述符（例如，实体 + 动作短语）结合起来，从 GKG 记录中提取一组事件式种子候选。汇总一段时间内的数据，便可得到一个自然反映兴趣变化的候选池。

评分与筛选。 令Q表示在一个时间窗口内聚合的全部候选种子。对于每个 $q \in \mathrm{Q}$，我们计算一个平衡显著性、时间特异性与多样性的分数：

$$S(q) = \log(1 + A(q)) + \alpha\,\mathrm{Geo}(q) + \beta\,\mathrm{Freq}(q) + \gamma\,\Delta t(q) + \delta\,\mathrm{Spec}(q) - \eta\,\mathrm{Cov}(q). \quad (1)$$

此处 $A(q)$在可用时为注意力代理指标（否则为每日观测值的单调函数），Geo(q)为地理广度，Freq(q)为观测次数，$\Delta t(q)$为信号所覆盖的日期跨度，Spec(q)为轻度特异性奖励（以避免通用种子），Cov(q)则惩罚几乎每天都出现的候选（通常表明其为通用术语）。

3.2 图构建

给定选定的种子查询 q，我们使用标准搜索API从网络来源检索查询条件化语料库 $\mathrm{D}(q) = \{d_i\}_{i=1}^{N}$。这是我们所针对的评估设置：智能体能够搜索、阅读并综合测试时返回的一小组相关来源中的信息。与数据集级摘要设置不同，该基准不假设（也不尝试构建）对整个语料库（例如"网络"）的持久全局表示。

为支持跨文档意义建构，我们随后构建一个针对查询的结构化表示，汇总 D(q) 中的信息。遵循 GraphRAG 风格的流程 [4]，我们使用大语言模型辅助的提取器，从文本中识别实体和关系陈述。由此得到的结构并非模式固定的知识图谱。关系是以证据为依据的简短自然语言主张，因此这种表示更接近由主张构成的超图，而非受谓词限制的知识图谱。

Graph definition. We construct a graph

$$G(q) = (V, E), \quad (2)$$

其中，每个节点 $v \in V$ 对应一个实体，每条边$e \in E$是一个元组 $(s,\ r,\ o,\ \mathrm{ev})$，包含主语 $s \in V$、宾语 $o \in V$、关系文本 r（类似主张的陈述），以及指向底层文本单元（句子或文本块）的证据指针 $\mathrm{ev} \subseteq \mathrm{U}$。每条边还带有一个支持信号（例如边权重、频次或证据量）。

社区检测与报告。 我们使用Leiden聚类将 $G(q)$ 划分为多个社区 $\mathrm{C}(q) = \{c_1、\ldots\ldots、c_m\}$。我们将每个社区视为一个主题（检索语料库中的一个连贯子故事）。对于每个社区 c，该流程都会生成一份简短的自然语言摘要，以及一组有依据的发现（与证据相关联的重要陈述）。这些产物为 D(q) 的主题结构提供了简洁、便于人阅读的界面。

3.3 社区角色与影响力

将 $G(q)$ 转化为基准的一大挑战在于规模和噪声。大型图包含许多外围实体和冗余边，而并非所有主题对问题生成都同等重要。为了让问题构建聚焦于语料中影响最大的部分，我们计算图的主题级视图，并为每个社区分配一个角色，以指导后续筛选。

社区元图。 我们定义一个社区元图$H(q) = (\mathrm{C}(q)\ E_H)$，其中每个节点都是一个社区，边则汇总由 $G(q)$ 引出的跨社区关系。

$$H(q) = (C(q), E_H). \quad (3)$$

直观而言，$H(q)$ 刻画了主题在经过查询条件筛选的语料中的交互方式。边 $(c_i, c_j) \in E_H$的权重同时反映了跨社区证据的强度和数量。

影响力得分。我们为每个社区计算影响力得分 $I(c)$，综合考量社区规模、连通性和证据支持：

$$I(c) = \tfrac{1}{4}\,z(\log(1 + |c|)) + \tfrac{1}{4}\,z(\mathrm{PR}(c)) + \tfrac{1}{4}\,z(\mathrm{BC}(c)) + \tfrac{1}{4}\,z(\mathrm{Ev}(c)), \quad (4)$$

其中， $|c|$ 表示社区规模，PR 和 BC 分别表示 $H(q)$ 上的 PageRank 和中介中心性，Ev(c) 统计支持 c 的唯一证据单元数量，$z(\cdot)$ 表示查询内标准化。

核心、桥接和卫星主题。 我们使用基于元图 $H(q)$ 的轻量级查询局部规则，为每个社区分配角色。直观来说，核心代表主导主题，桥接连接原本相互独立的主题，卫星则代表外围主题，并与核心／桥接结构紧密相连。

- 核心：具有高影响力、代表 D(q) 中主导子故事的主题。
- 桥接：介数中心性较高、连接原本相互独立的子故事的主题，通常充当中介，说明某个主题中的证据如何影响另一个主题。
- 卫星：提供辅助背景信息，并与核心或桥接主题紧密相连的外围主题。

具体而言，令 $I(c)$ 表示公式 (4) 中的影响力得分，并令 BC(c) 表示 $H(q)$ 上的中介中心性。我们定义一个简单的筛选算子 TopK(S, f, k)，该算子返回集合 S 中依据评分函数 f 取值最大的 k 个元素。我们将 Core 集合定义为按影响力排序位居前列的社区，

$$C_{\text{core}}(q) = \text{TopK}\left(C(q),\, I,\, k_{\text{core}}\right), \tag{5}$$

其中 $k_{\text{core}} = \max\{5,\, 0.3\,|\text{C}(q)|\}$。类似地，我们将 Bridge 集合定义为按中介中心性排序位居前列的社区，

$$C_{\text{bridge}}(q) = \text{TopK}\left(C(q),\, \text{BC},\, k_{\text{bridge}}\right), \tag{6}$$

其中 $k_{\text{bridge}} = \max\{3,\, 0.2\,|\text{C}(q)|\}$。为了识别 Satellite 主题，我们将任何与 Core 或 Bridge 主题在 $H(q)$ 中存在强链接的剩余社区纳入其中。如果某个社区 c 与任一 Core 或 Bridge 社区之间存在强元边，则将其标记为 Satellite，即存在 $c' \in \text{C}_{\text{core}}(q) \cup \text{C}^{\text{bridge}}(q)$，使得 $\max\{w(c, c')\ w(c', c)\} \geq \tau_q$；其中，$w(\cdot,\cdot)$ 表示元边权重，τ_q 设置为查询 q 的元边权重中位数。这些规则轻量、在不同查询之间保持稳定，并使角色分配具有可解释性。这些角色会将下游数据包选择偏向显著主题和具有结构意义的跨主题链接，同时保留足够的外围上下文，以应对现实中的问题。

3.4 基准实例构建

基于 §3.3 生成的查询条件主题结构，接下来的步骤将构建一个基准实例，具体包括：(i) 选择明确的跨主题链接；(ii) 为跨文档意义建构打包所需信息；以及 (iii) 生成并验证独立的问答对。

3.4.1 连接关系。 意义建构问题依赖多个主题及这些主题之间的连接方式。因此，我们提取明确的连接关系，将各个社区联系起来。令 $\text{comm} : V \to \text{C}(q)$ 将每个实体映射到其所属社区。若一条边 $e = (s,\ r,\ o,\ \text{ev}) \in E$ 跨越社区边界，则它是连接关系，即 $\text{conn}(e) = \mathbb{I}\left[\text{comm}(s) \neq \text{comm}(o)\right]$。并非所有跨主题断言都同样有用，因此我们保留一小组排名靠前的连接边，其排名依据是支持信号（边权重 / 证据量）；优先选择在检索语料库中反复得到证据支持的关联，同时保持信息包简洁。

3.4.2 数据包构建。 为了大规模生成问题，同时避免向大语言模型暴露完整的图或文档集，我们构建紧凑的信息包，仅保留跨主题推理所需的信息。

每个数据包都围绕一小组选定主题构建，既能呈现有意义的跨主题关联，又能保持较高的 token 效率。我们从一个影响力较高的核心主题开始，并将其与一个或多个桥接主题配对，以连接元图的其他区域 $H(q)$。大多数情况下，一个集合包含两个主题（一个突出的核心主题和一个起连接作用的桥接主题）。必要时，为保留连接链，我们允许集合包含三个主题（例如，核心主题–桥接主题–核心主题）。只有当至少一个选定的连接关系连接了集合中的主题时，该集合才会被保留，从而确保生成的问答候选内容依赖跨主题关联，而非孤立的事实查找。

给定一个选定的集合，数据包 P 包含 (i) 每个主题的社区卡片（摘要和带有证据 ID 的核心发现），以及 (ii) 集合中连接各主题的连接关系。这种表示有意保持简洁，同时保留跨文档意义建构所需的要素。

3.4.3 问答生成与验证。在给定数据包$P(B)$的条件下，大语言模型生成器会提出一句话的、类似用户提问的问题，并给出简短、客观且有所提供发现和连接关系作为依据的答案。提示会引导模型生成符合真实情境的意义建构意图（例如触发型、后果或跨主题关联），并明确避免类似冷知识的实体查询（参见第 8 节中的提示 P1）。我们采用确定性解码（温度为 0），记录所有引用的材料 ID 以便追溯，并要求生成器输出结构化记录字段：必需社区、支持性发现（每个必需社区至少一条）以及支持性连接关系 ID。这些字段不会展示给受评模型，但可用于后续筛选和分析，确保问题需要跨主题整合，而非依赖单一孤立事实。

我们采用 LLM即评委范式 [5, 36]筛选候选项。尽管已有文献记录了主观判断的可靠性问题[6, 10, 36], , 先前研究也发现，LLM 判断的结果与众包工作者的判断具有相当的共识度，在某些情境下甚至能达到或超过人类表现 [5, 7]。我们的检查刻意限定为仅依据证据且具有客观性：评委必须核实 (a) 候选项是否得到所提供材料的支持；(b) 它是否必然依赖多个主题以及至少一个连接关系。这与事实核查和自然语言推理相似，而大语言模型在这些任务中已展现出较强的表现 [7]。

在我们的实验中，我们采用由三个评审大语言模型组成的评审组；每位评审都会使用验证器提示独立评估候选项，并返回包含二元标记和简短、以产物为依据的理由的结构化 JSON 判定（参见 §8 中的提示 P2）。只有满足所有硬性标准并通过必要性测试的候选项才会被保留。我们通过多数票汇总评审结果；此外，只要有任何评审标记证据缺失、答案非唯一或出现琐碎事实偏移，我们就会拒绝该候选项，优先考虑精确率而非召回率。

4 数据集特征与产物

iAgentBench中的实例不仅仅是一个问答对：它包含可审计工件，用以揭示为什么 回答该问题需要

跨文档综合。该设计遵循四个目标：（G1）真实性（由兴趣驱动的开放域问答，而非由知识库或琐事诱导的问题），（G2）跨主题依赖（无法从单一主题回答的问题），（G3）信息寻求保真度（问题反映常见用户意图），以及（G4）可审计性（可追溯的证据和中间结构）。我们将这些目标体现在下文所述的已发布实例产物中。

（*G1*）真实的开放域问答主题。种子查询来源于真实的网络关注度信号（§3.1），并与查询条件化网络语料库配对，从而能够评估以事件为中心的信息需求；这类需求与用户实际搜索的内容相似。

（*G2*）跨主题、依赖连接关系的问答。每个保留的问题都明确关联至少两个主题（社区）和至少一种连接关系。我们公开支持性发现、连接关系及其 ID，以便评估者诊断模型失败是源于检索还是综合。

（*G3*）信息寻求意图模式。除了要求跨主题证据外，问题的设计也旨在贴近用户试图理解不断发展的主题时表达信息需求的方式。在生成过程中，我们倾向于采用一小组反复出现的意图模式，并将每个实例对应的预期模式记录为 intent_pattern。这五种模式涵盖了用户寻求理解的常见方式：解释型询问基于证据的“如何／为何”，且答案取决于某个特定关联；关联型询问两项进展之间有何关联；触发型询问什么行动或决定引发了后续反应；后果询问某种明确关联带来了什么结果；利害关系型询问将各项进展联系起来的具体条件、限制或目标。这有助于按意图类型开展分层分析，并实现超越总体准确率的评估。

（*G4*）可审计性与发布的产物。表1总结了每个实例发布的核心字段，包括明确说明信息需求的意图元数据。我们还发布了中间产物，以支持对基于图的检索和智能体失效进行分析。可使用相同的构建规则，针对新的时间窗口重新生成实例；由于已记录检索到的语料库，该基准还支持污染检查

以及受控的重复运行。

5 实验与结果

在本节中，我们评估证据获取和证据整合如何影响端到端问答表现，以及 iAgentBench 与标准开放域问答基准相比如何。我们在三种推理设置下比较四种广泛采用的大语言模型：基础设置（不使用外部工具）、检索增强生成（SearxNG [27] 检索到的文档首页）以及 *Reflexion*（对检索到的证据进行智能体自我反思 [28]）。我们报告两个标准开放域问答基准 SimpleQA [32] 和 HotpotQA [33]，上的准确率，并与 iAgentBench 一并呈现。由于这些测试集规模庞大，端到端智能体评估成本高昂，我们参照以往研究中的类似做法 [29]，从每个基准中随机抽取 500 个问题。准确率根据 SimpleQA 评估流程 [32] 计算。

一个首先显现且一致的规律是，检索显著提升了所有数据集和模型上的表现。在图 2 中，所有点都位于 $y=x$ 线之上，表明获取外部证据

字段	描述
id	实例的唯一标识符。
查询	种子查询（兴趣驱动的主题字符串）。
时间_窗口	用于主题来源搜集和检索的时间区间。
问题	最终的一句话问题。
答案	简短、客观的答案（实体、日期或短语）。
意图_模式	信息寻求意图的标签：解释型，关联型、触发型，后果、利害关系型。
已检索_来源	已检索文档的ID（q）网址/元数据。
文本_单元	用作证据单元的文本块/句子的 ID。
故事_图_统计	G（q）的图统计（节点数、边数）。
社区	带有角色标签的社群 ID（核心/桥接/卫星）和影响力分数。
社区_卡片	各社群摘要和有依据的发现（含 ID）。
连接器	连接关系（含 ID 和证据指针）。
数据包	选定的集合 B 和数据包 P（B），用于生成。
支持性_发现	查找支持问答的 ID（按≥ 1 要求社区）。
支持_的连接器	问答所需的连接关系 ID（≥ 1）。
评审_决策	三模型大语言模型评审组的输出，包括各项标准的标记和理由。

表1：每个 iAgentBench 实例发布的核心字段。

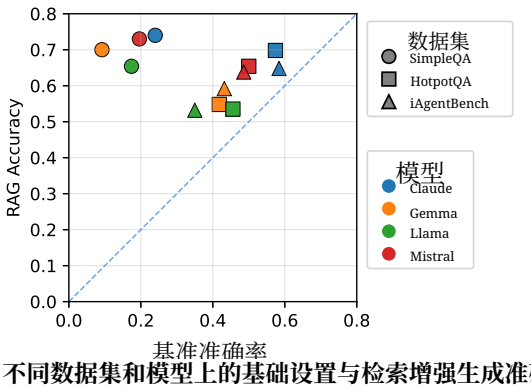

图2：不同数据集和模型上的基础设置与检索增强生成准确率。位于对角线上方的点表示获取证据带来了提升。

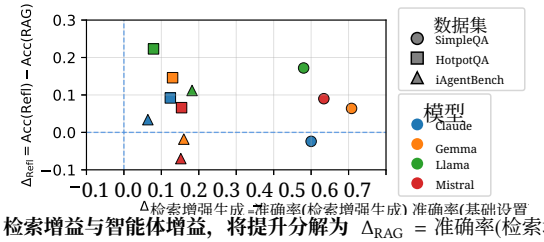

图3：检索增益与智能体增益，将提升分解为 Δ_{RAG} = 准确率(检索增强生成) –准确率(基础设置) 以及 Δ_{Refl} =准确率(反思) –准确率(检索增强生成)。正值 Δ_{Refl} 表示迭代带来了超出检索增强生成的收益，而负值则表示性能退化。

即使对于强大的前沿模型，这也有益处。这种效果的幅度因基准测试而异。SimpleQA 的基础设置→检索增强生成提升幅度最大，这表明许多问题的主要瓶颈在于证据获取：一旦检索到相关段落，模型通常就能可靠地提取简短答案。

模型	SimpleQA					HotpotQA					iAgentBench				
	Base	RAG	反思	Δ_{RAG}	Δ_{Refl}	Base	RAG	反思	Δ_{RAG}	Δ_{Refl}	Base	RAG	反思	Δ_{RAG}	Δ_{Refl}
Claude Sonnet 4.5	0.240	**0.740**	0.716	0.500	−0.024	0.574	0.700	**0.790**	0.126	0.090	0.584	0.648	**0.682**	0.064	0.034
LLaMA 4 Maverick 17B	0.174	0.657	**0.826**	0.483	0.169	0.456	0.537	**0.758**	0.081	0.221	0.356	0.532	**0.628**	0.176	0.096
Mistral Large 3 (675B)	0.196	0.730	**0.820**	0.534	0.090	0.500	0.654	**0.720**	0.154	0.066	0.486	**0.638**	0.564	0.152	−0.074
Gemma 3 27B	0.092	0.700	**0.764**	0.608	0.064	0.420	0.548	**0.694**	0.128	0.146	0.432	**0.592**	0.570	0.160	−0.022

表 2：基础设置、检索增强生成和 Reflexion 在 SimpleQA、HotpotQA 与 iAgentBench 上的准确率； Δ_{RAG} = 检索增强生成–基础设置，Δ_{Refl} = 反思–检索增强生成。

HotpotQA 也能从检索中受益，但提升幅度较小，这与证据获取和多跳组合共同发挥作用的情形相符。在已完成的运行中，iAgentBench 在检索增强生成下同样有所提升，但仍存在的差距表明，检索证据只是挑战的一部分，跨主题整合仍然至关重要。

这些差异也反映了各基准测试的构建方式及其侧重考查的内容。SimpleQA 的设计旨在对基础大语言模型构成挑战：它针对 GPT-4 系列模型的回答，以对抗方式收集问题，并将问题限定为简短且只有一个无可争议答案的形式 [32]。因此，尽管问题涵盖广泛的主题，各模型在基础设置下的准确率仍然偏低。如表 2 所示，启用检索后，SimpleQA 对大多数模型而言成为最简单的基准测试，这表明其剩余难点主要在于获取证据和精确提取。相比之下，iAgentBench 面向从关注度信号中选取的真实高流量主题；即使采用检索增强生成，仍存在显著差距，这说明仅仅提供证据并不足够，系统还必须连贯地整合来自多个上下文的信息。

图3进一步将检索收益与多步自我反思的额外效应区分开来。虽然 Δ_{RAG} 通常为正，但Δ_{Refl} 则好坏参半：当额外步骤有助于连接分散证据时，Reflexion可以优于检索增强生成，但当多步推理引入偏离或过度校正时，它也可能出现退步。这种行为在各个数据集中均可见，在iAgentBench上尤为显著，不同模型在迭代究竟有益还是有害方面表现各异。

该表具体凸显了这些差异。在SimpleQA上，若干模型通过Reflexion在检索增强生成之外获得了大幅提升（例如LLaMA和Mistral），这表明即使检索已经较强，迭代检查也有助于从初始提取错误中恢复。在HotpotQA上，Reflexion为LLaMA带来了可观提升，这与多跳组合的收益相一致，而其他模型的改进则较小。在iAgentBench上，LLaMA受益于Reflexion，但Mistral和Gemma相对于检索增强生成有所下降，这进一步说明智能体流程并非一律有益，应评估其证据使用的稳定性，而不是假定步骤越多改进就越单调递增。

6 结论

在这项工作中，我们提出 iAgentBench，这是一个动态开放域基准，用于评估信息寻求智能体回答需要进行跨文档意义建构，而非单段落提取的问题的能力。iAgentBench 将由流量驱动的主题选择与查询条件化网络语料库及故事图表示相结合，展现主题及其间的明确联系，从而能够构建独立且贴近用户的问题，而回答这些问题必然依赖多个主题和连接关系。为了支持超越标题准确率的科学研究，每个实例都会附带可审计工件，包括意图模式、支持性发现/连接关系、信息包和评审判定，以便将失败分析为证据获取、证据整合或多步稳定性方面的故障。

我们的实验表明，检索能够稳定提升各项基准测试中的表现，但在检索增强生成（RAG）设置下，iAgentBench 仍存在显著差距，这说明仅获取证据并不能解决跨主题问题。进一步分解性能提升后发现，智能体自我反思并非总是有益；多步推理可能提升某些模型的表现，却会削弱其他模型的表现，这凸显出评估时不仅要关注检索质量，还要关注证据使用的可靠性。我们希望 iAgentBench 及其发布的制品能为衡量和改进面向意义建构的信息寻求系统提供切实可行的基础，使其能够应对不断变化的现实主题。

7 局限性

动态构建的成本。 与静态问答数据集相比，动态基准测试的成本更高。每当生成新的评估窗口，iAgentBench 都需要进行检索、查询条件化图构建和生成，从而增加计算开销和 API 开销。在不断变化的信息环境中评估检索增强系统和智能体系统，这种权衡不可避免。

对生成模型的依赖。 该流程依赖生成模型进行图构建、问题生成和大语言模型评审验证，因此可能出现幻觉、遗漏或有偏的证据解读。我们通过使用结构化提示、多模型评审组，并发布中间产物以确保可审计性，来降低这些风险。先前研究发现，对于客观的自然语言处理评估任务，基于大语言模型的标注可以达到与众包工作者相当的一致性，在某些情况下，其表现可与人类表现持平或更优 [5, 7]。

实验覆盖范围。 我们评估了多种大语言模型骨干在基础设置、检索增强生成和智能体自我反思变体下的表现，但其他检索器、工具集、长时程规划策略以及更广泛的超参数扫描仍有待探索。扩展实验覆盖范围很有价值，但很快就会带来组合爆炸和难以承受的成本，因此当前结果旨在提供有代表性的切片，以支持后续评估之间的可比性。

参考文献

[1] Akari Asai、Kazuma Hashimoto、Hannaneh Hajishirzi、Richard Socher 和 Caim- ing Xiong。2020年。学习在维基百科图上检索推理路径以进行问答。arXiv:1911.10470 [cs] doi:10.48550/arXiv.1911.10470

注：本文作为会议论文发表于 ICLR 2020。代码可在 https://github.com/AkariAsai/learning_to_retrieve_reasoning_paths 获取。[2] Nicholas Carlini、Daphne Ippolito、Matthew Jagielski、Katherine Lee、Florian Tramer 和 Chiyuan Zhang。2023年。量化神经语言模型中的记忆。arXiv: 2202 07646 [ ] 10 48550 2202 07646 . cs doi: . /arXiv. . [3] Preetam Prabhu Srikar Dammu、Himanshu Naidu 和 Chirag Shah。2025年。Dynamic-KGQA：用于生成自适应问答数据集的可扩展框架。载于第48届国际 ACM SIGIR 信息检索研究与发展会议论文集（SIGIR '25）。计算机协会，纽约，美国，3498–3508。doi:10.1145/3726302. 3730324[4] 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：一种用于查询导向摘要的图检索增强生成方法。arXiv 预印本 arXiv:2404.16130（2024）。[5] Fabrizio Gilardi、Meysam Alizadeh 和 Maël Kubli。2023年。ChatGPT 在文本标注任务上的表现优于众包工作者。美国国家科学院院刊 120, 30（2023），e2305016120。[6] Jiawei Gu、Xuhui Jiang、Zhichao Shi、Hexiang Tan、Xuehao Zhai、Chengjin Xu、Wei Li、Yinghan Shen、Shengjie Ma、Honghao Liu 等。2024年。关于 LLM即评委的综述。arXiv 预印本 arXiv:2411.15594（2024）。[7] Jian Guan、Jesse Dodge、David Wadden、Minlie Huang 和 Hao Peng。2023年。语言模型会产生幻觉，但可能擅长事实核查。arXiv 预印本 arXiv:2310.14564（2023）。[8] Kelvin Guu、Kenton Lee、Zora Tung、Panupong Pasupat 和 Mingwei Chang。2020年。检索增强语言模型预训练。载于第37届国际机器学习大会论文集。PMLR，3929–3938。[9] Ziwen Han、Meher Mankikar、Julian Michael 和 Zifan Wang。2025年。搜索时数据污染。arXiv: 2508 13180 [ ] 10 48550 2508 13180 . cs doi: . /arXiv. . [10] 2023 Fan Huang、Haewoon Kwak 和 Jisun An。. ChatGPT 比人工标注者更好吗？ChatGPT 在解释隐性仇恨言论方面的潜力与局限性。载于 ACM Web 大会 2023 配套论文集。294–297。[11] Gautier Izacard 和 Edouard Grave。2021年。利用段落检索与生成模型实现开放域问答。arXiv:2007.01282 [cs] doi:10.48550/arXiv.2007.01282[12] Gautier Izacard、Patrick Lewis、Maria Lomeli、Lucas Hosseini、Fabio Petroni、Timo Schick、Jane Dwivedi-Yu、Armand Joulin、Sebastian Riedel 和 Edouard Grave。2022年。Atlas：通过检索增强语言模型实现少样本学习。arXiv: 2208 03299 [ ] 10 48550 2208 03299 . cs doi: . /arXiv. . Robin Jia 和 Percy [13] 2017 Liang。. 用于评估阅读理解系统的对抗样本。arXiv:1707.07328 [cs] doi:10.48550/arXiv.1707.07328 注：EMNLP 2017。[14] Vladimir Karpukhin、Barlas Oguz、Sewon Min、Patrick Lewis、Ledell Wu、Sergey Edunov、Danqi Chen 和 Wen-tau Yih。2020年。用于开放域问答的密集段落检索。载于2020年自然语言处理实证方法会议（EMNLP）论文集，Bonnie Webber、Trevor Cohn、Yulan He 和 Yang Liu（编）。计算语言学协会，线上，6769–6781。doi:10.18653/v1/2020.emnlp-main.550[15] Jungo Kasai、Keisuke Sakaguchi、Yoichi Takahashi、Ronan Le Bras、Akari Asai、Xinyan Yu、Dragomir Radev、Noah A. Smith、Yejin Choi 和 Kentaro Inui。2024年。RealTime QA：此刻的答案是什么？arXiv:2207.13332 [cs] doi:10.48550/arXiv.2207.13332 注：RealTime QA 网站：https://realtimeqa.github.io/。[16] Douwe Kiela、Max Bartolo、Yixin Nie、Divyansh Kaushik、Atticus Geiger、Zhengx- uan Wu、Bertie Vidgen、Grusha Prasad、Amanpreet Singh、Pratik Ringshia、Zhiyi Ma、Tristan Thrush、Sebastian Riedel、Zeerak Waseem、Pontus Stenetorp、Robin Jia、Mohit Bansal、Christopher Potts 和 Adina Williams。2021年。Dynabench：重新思考自然语言处理中的基准测试。arXiv:2104.14337 [cs] doi:10.48550/arXiv.2104. 14337 注：NAACL 2021。[17] Najoung Kim、Phu Mon Htut、Samuel R. Bowman 和 Jackson Petty。2023年。(QA)^2：基于可疑假设的问答。arXiv:2212.10003 [cs] doi:10.48550/arXiv.2212.10003[18] Tom Kwiatkowski、Jennimaria Palomaki、Olivia Redfield、Michael Collins、Ankur Parikh、Chris Alberti、Danielle Epstein、Illia Polosukhin、Jacob Devlin、Kenton Lee、Kristina Toutanova、Llion Jones、Matthew Kelcey、Ming-Wei Chang、Andrew M. Dai、Jakob Uszkoreit、Quoc Le 和 Slav Petrov。2019年。Natural Questions：问答研究的基准。计算语言学协会会刊 7（2019），452–466。doi:10.1162/tacl_a_00276[19] Patrick Lewis、Ethan Perez、Aleksandra Piktus、Fabio Petroni、Vladimir Karpukhin、Naman Goyal、Heinrich Küttler、Mike Lewis、Wen-tau Yih、Tim Rocktäschel、Sebastian Riedel 和 Douwe Kiela。2020年。面向知识密集型自然语言处理任务的检索增强生成。载于《神经信息处理系统进展》，第33卷。Curran Associates, Inc.，9459–9474。[20] Yucheng Li、Frank Guerin 和 Chenghua Lin。2024年。大语言模型的开源数据污染报告。arXiv:2310.17589 [cs] doi:10. 48550/arXiv.2310.17589[21] Adam Liška、Tomáš Kočiský、Elena Gribovskaya、Tayfun Terzi、Eren Sezener、Devang Agrawal、Cyprien de Masson d' Autume、Tim Scholtes、Manzil Zaheer、Susannah Young、Ellen Gilsenan-McMahon、Sophia Austin、Phil Blunsom 和 Angeliki Lazaridou。2022年。StreamingQA：问答模型随时间适应新知识的基准。arXiv:2205.11388 [cs] doi:10.48550/arXiv.2205.11388[22] Zhiyi Ma、Kawin Ethayarajh、Tristan Thrush、Somya Jain、Ledell Wu、Robin Jia、Christopher Potts、Adina Williams 和 Douwe Kiela。2021年。Dynaboard：用于整体式下一代基准测试的评估即服务平台。arXiv:2106.06052 [cs] doi:10.48550/arXiv.2106.06052 [23] Reiichiro Nakano、Jacob Hilton、Suchir Balaji、Jeff Wu、Long Ouyang、Christina Kim、Christopher Hesse、Shantanu Jain、Vineet Kosaraju、William Saunders、Xu Jiang、Karl Cobbe、Tyna Eloundou、Gretchen Krueger、Kevin Button、Matthew Knight、Benjamin Chess 和 John Schulman。2022年。WebGPT：结合人类反馈的浏览器辅助问答。arXiv:2112.09332 [cs]doi:10.48550/arXiv.2112.09332 注：32页。[24] Pranav Rajpurkar、Jian Zhang、Konstantin Lopyrev 和 Percy Liang。2016年。SQuAD：用于文本机器阅读理解的100,000+ 个问题。arXiv:1606.05250 [cs] doi:10.48550/arXiv.1606.05250 注：即将在《2016年自然语言处理实证方法会议论文集》（EMNLP）中发表。[25] Mathieu Ravaut、Bosheng Ding、Fangkai Jiao、Hailin Chen、Xingxuan Li、Ruochen Zhao、Chengwei Qin、Caiming Xiong 和 Shafiq Joty。2025年。大语言模型污染检测方法综合综述。arXiv:2404.00699 [cs] doi:10.48550/arXiv.2404.00699 注：2025年7月被TMLR接收。18页，1幅图，3张表。[26] Christopher Rawles、Sarah Clinckemaillie、Yifan Chang、Jonathan Waltz、Gabrielle Lau、Marybeth Fair、Alice Li、William Bishop、Wei Li、Folawiyo Campbell-Ajala、Daniel Toyama、Robert Berry、Divya Tyamagundlu、Timothy Lillicrap 和 Oriana Riva。2025年。AndroidWorld：自主智能体的动态基准测试环境。arXiv: 2405 14573 [ ] 10 48550 2405 14573 . cs doi: . / [27] [ ] arXiv. . SearXNG贡献者。无日期。. SearXNG。 https://github.com/searxng/searxng 源代码仓库（AGPL-3.0）.[28] Noah Shinn、Federico Cassano、Ashwin Gopinath、Karthik Narasimhan 和 Shunyu Yao。2023年。Reflexion：采用语言强化学习的语言智能体。《神经信息处理系统进展》36（2023），8634–8652。[29] Jiashuo Sun、Chengjin Xu、Lumingyuan Tang、Saizhuo Wang、Chen Lin、Yeyun Gong、Lionel M Ni、Heung-Yeung Shum 和 Jian Guo。2023年。Think-on-graph：大语言模型在知识图谱上的深度、负责任推理。arXiv预印本 arXiv:2307.07697（2023）。[30] GDELT项目。无日期。. GDELT项目。https://www.gdeltproject.org/。访问日期：2026-02-02。[31] Vincent Traag、Ludo Waltman 和 Nees Jan van Eck。2019年。从Louvain到Leiden：确保社区高度连通。《科学报告》9，1（2019年3月），5233。arXiv:1810.08473 [cs] doi:10.1038/s41598-019-41695-z[32] Jason Wei、Nguyen Karina、Hyung Won Chung、Yunxin Joy Jiao、Spencer Papay、Amelia Glaese、John Schulman 和 William Fedus。2024年。衡量大语言模型的短文本事实准确性。arXiv:2411.04368 [cs] doi:10.48550/ arXiv.2411.04368 注：博客文章: https://openai.com/index/introducing- simpleqa/。[33] Zhilin Yang、Peng Qi、Saizheng Zhang、Yoshua Bengio、William W. Cohen、Ruslan Salakhutdinov 和 Christopher D. Manning。2018年。HotpotQA：用于多样化、可解释多跳问答的数据集。arXiv:1809.09600 [cs] doi:10.48550/arXiv.1809.09600 注：EMNLP 2018长文。前三位作者贡献相同。数据、代码和博客文章见 https://hotpotqa.github.io/。[34] Shunyu Yao、Jeffrey Zhao、Dian Yu、Nan Du、Izhak Shafran、Karthik Narasimhan 和 Yuan Cao。2023年。ReAct：在语言模型中协同推理与行动。arXiv:2210.03629 [cs] doi:10.48550/arXiv.2210.03629 注：v3是ICLR相机就绪版本，修正了若干拼写错误。项目网站（含代码）：https://react-lm.github.io。[35] Zhehao Zhang、Jiaao Chen 和 Diyi Yang。2024年。DARG：通过自适应推理图动态评估大语言模型。arXiv:2406.17271 [cs] doi:10.48550/arXiv.2406.17271[36] Lianmin Zheng、Wei-Lin Chiang、Ying Sheng、Siyuan Zhuang、Zhanghao Wu、Yonghao Zhuang、Zi Lin、Zhuohan Li、Dacheng Li、Eric Xing 等。2024年。使用MT-Bench和Chatbot Arena评判LLM即评委。《神经信息处理系统进展》36（2024）。[37] Kaijie Zhu、Jindong Wang、Qinlin Zhao、Ruochen Xu 和 Xing Xie。2024年。DyVal 2：通过元探测智能体动态评估大语言模型。arXiv:2402.14865 [cs] doi:10.48550/arXiv.2402.14865 注：技术报告；20页。

8 附录提示词

此处，我们提供 iAgentBench 框架中用于问答生成步骤（P1）和评估步骤（P2）的提示词，其中 P2 用于 LLM即评委设置。

提示 P2：LLM即评委验证提示词

系统：
你是一名专家验证器。请仅根据提供的证据，判断这一个问答候选应当通过还是不通过。
通过标准（必须全部满足）：

（1）仅证据支持：
- 答案完全由问答候选对象中提供的证据文本支持。
- 不得使用外部知识、假设或未陈述事实。

（2）多社区必要性：
- 该问答确实至少需要两个社区。
- 必要性测试：如果移除任意一个必需社区中的所有发现文本，问题将变得无法回答，或明显存在歧义。

（3）连接关系的必要性：
- 问答确实至少需要一个连接关系。
- 必要性测试：如果移除所有引用的连接文本，问题将变得无法回答，或明显存在歧义。
- 所引用的连接关系必须是推导答案所必需的（而不只是松散相关）。

(4) 客观问答：
- 该问题有一个由证据支持的唯一最佳答案。
- 该问题/答案不是主观的、解释性的或依赖于视角的。
- 答案不包含模棱两可的措辞（“可能/也许/很可能”），也不包含规范性语言（“好/坏/重要”）。

(5) 自然的用户问题：
- 该问题恰好是一个句子，读起来像自然的用户查询（而非测验/测试题目）。

(6) 反琐事：
- 仅凭身份/别名/名称等价无法回答该问答。
- 如果存在身份/别名内容，其本身必须不足以回答问题；该问答仍必须依赖非身份类跨社区内容。

(7) 证据的存在与一致性：
- 证据发现包含来自至少两个不同社区 ID 值中各一个 ≥1发现。
- 证据连接器包含 ≥1连接关系。
- 每个证据项都必须包含其标识符和文本。

（8）独立可理解性：
- 问题必须能够独立完整地理解，且具体明确，无需借助任何其他产物。
- 问题中的所有关键指称对象都必须明确指出，或具有唯一性。
- 如果由于缺少先行项或存在未明确的实体/事件，导致合理的读者会追问“哪一个？”“谁？”或“具体是什么？”，则判定为失败。

输入：
QA_CANDIDATE:

{问答_候选项_JSON}

输出：
返回一个 JSON 对象，且必须包含以下键，不能多也不能少：

- 仅证据支持标志：true/false
- 仅证据支持推理：字符串
- 多社区必要性标志：true/false
- 多社区必要性推理：字符串
- 连接器必要性标志：true/false
- 连接器必要性推理：字符串
- 客观问答标志：true/false
- 客观问答推理：字符串
- 自然用户问题标志：是/否
- 自然用户问题推理：字符串
- 反琐碎知识标志：是/否
- 反琐碎推理：字符串
- 证据存在一致性标志：是/否
- 证据存在一致性推理：字符串
- 独立清晰度标志：true/false
- 独立清晰度推理：字符串

规则：
- 每个 *_reasoning 必须为 1–2 句话，内容具体，且仅引用 QA_CANDIDATE 字段。
- 如果仅根据 QA_CANDIDATE 无法有把握地将某项标准标记为 true，则将其 *_flag 设为 false，并说明原因。

仅返回 JSON 对象，不要添加其他文本或 Markdown。

提示 P1：问答生成提示

系统：
你需要生成开放领域问答对，这些问答对要求基于所提供的故事素材进行跨社区意义建构。
你能看到的内容（也是你唯一可以使用的内容）：

- 社区卡片：社区摘要 +有依据的发现（含 ID）。
- 连接关系：明确的跨社区关系（含 ID），用于连接各个社区。

定义：
跨社区意义建构问题无法仅凭单张社区卡片（或单条孤立的发现）作答。
它必须要求整合多个社区的信息，并且至少包含一条连接这些社区的明确连接关系。
必需约束（每个项目都必须满足）：

(1) 多社区要求：
- 每个问答都必须涉及至少两个不同的社区。
- supporting_findings 必须包含来自每个所需社区的 ≥1 条发现。

(2) 连接关系要求：
- 每个问答都必须涉及 ≥1 个连接关系。
- 所引用的连接关系必须是必要的：如果移除所引用的连接关系，问题就会变得无法回答或含糊不清。
- 引用的连接关系必须连接必需社区（直接连接，或通过所述关系连接）。

(3) 客观答案要求：
- 问题必须有一个由所提供的发现/连接器支持的最佳答案。
- 不得生成主观、带有解读色彩或取决于视角的问题或答案。

问题风格（一个句子，真实自然，像用户提出的问题）：

- 听起来要像一个真实用户在试图了解某种情况（而不是在做测验）。
- 优先选择有据可依、依赖明确跨社区联系的意义建构意图：关联型、触发型→响应、原因→后果，机制／作用、明确提出的条件／约束。
- 好的模式包括：
  - "X 与 Y 有什么关联？"
  - "什么行动／决策导致了 Y？"
  - "X 之后是什么触发了 Y？"
  - "X 对 Y 有何贡献？"
  - "X 和 Y 之间存在什么明确提及的条件/要求？"
- 不要提及图、超图、社区、摘要、发现、信息包、数据集，也不要说"根据图"。
- 避免采用考试/问答竞赛的语气（例如"哪位领导人……""哪个国家……""哪一年……"），除非问题仍明确要求涉及多个社区的 + 连接关系依赖关系。

不要生成：

- 主观／诠释性问题："意义""重要性""它揭示了什么""它如何反映""是否合理""谁是对的""人们有何感受""它意味着什么"。• 答案可能因视角、表述方式或价值观而异的任何问题。• 纯查找／仅涉及身份／仅涉及别名的问题（例如"X 是谁？""X 的真实姓名是什么？"），除非问题还要求建立非身份类连接关系，并进行真正的跨社区整合。• 宽泛的概述性问题（"发生了什么？""总结……"），但未明确要求依赖特定连接关系的跨社区链接。

答案风格：

- 简短、客观，并且有产物提供的唯一且明确的依据（目标为 ≤ 12 个词元；仅在无法避免时稍长）。• 不得使用模棱两可的措辞（"可能／也许／很可能"）或规范性语言（"好／坏／重要"）。• 不得引入所提供的发现／连接关系中未包含的事实。• 如果没有依据支持唯一且客观的答案，则不要输出该问答对。

意图标签：
将 intent_pattern 设置为 ["解释型","关联型","触发型","后果","利害关系型"]中的一个，其中：

- 解释型：基于具体跨社区链接的有依据解释（而非"重要性"）
- 关联型：两个发展或实体如何通过连接关系建立关联• 触发型：什么行动、决策或事件导致了跨社区响应或结果• 后果：跨社区链接带来了什么结果
- 利害关系型：将社区联系起来的具体明确约束、条件、要求或目标（不是"为什么重要"）

内部自我 -检查（在输出每个项目之前运行这些检查；不要输出检查文本）：

（1）客观答案测试：
- 引用的发现/连接关系是否支持恰好一个最佳简短答案?
- 理性的读者是否会给出相同的答案？（2）多社区必要性测试：
- 如果移除任意一个必需的社区卡片，该问题是否会变得无法回答或产生歧义？（3）连接关系的必要性测试：
- 如果移除所有引用的连接关系，该问题是否会变得无法回答或产生歧义？（4）反琐碎性测试：
- 它是否让人感觉像一个真实的意义建构问题（而不是测验题或实体-属性查询）？

输出：
返回一个最多包含 {K} 个项目的 JSON 数组。每个项目必须包含：

- 问题：字符串（一个句子）
- 答案：字符串（简短、客观、有依据）• [2+] 必需社区：社区 ID 支持性发现：[社区 ID：……，发现 ID：……，……]（每个必需社区）• 支持性连接器：[连接器 ID，……]（≥1；必须不可或缺；必须连接必需社区）• [ ] 意图模式：以下之一：「解释型」「关联型」「触发型」「后果」「利害关系型」 why_multi_community：用一个句子说明为何需要多个社区 +连接器
- （从内容角度描述；不要提及图／社区／信息包）

多样性：

- 尽可能在所有条目中变换 intent_pattern；除非别无选择，否则不要对所有条目都使用相同的 connector_id。

COMMUNITY_CARDS:
{COMMUNITY_CARDS_JSON}

CONNECTOR_RELATIONS:
{CONNECTORS_WITH_IDS}

最多生成 {K} 个问答对。仅返回 JSON 数组，不要添加任何其他文本或 Markdown。