# 多版本系统问答：Ciena案例研究

Parham Khamsepour*†、Mark Cole†、Ish Ashraf†、Sandeep Puri†、Mehrdad Sabetzadeh*、Shiva Nejati*
*渥太华大学，800 King Edward Avenue，渥太华，ON K1N 6N5，加拿大
†Ciena公司，7035 Ridge Road，汉诺威，MD 21076，美国 电子邮箱：{parham.khamsepour、m.sabetzadeh、snejati}@uottawa.ca；{mcole、iashraf、spuri}@ciena.com

arXiv:2601.02345v1 [cs.SE] 5 Jan 2026

摘要—**公司经常需要应对多版本系统，即同一软件的多个版本同时运行。针对多版本系统中的文档进行问答颇具挑战性，因为不同发布版本拥有各自独特但又相互重叠的文档。鉴于当前最先进的问答技术在多版本系统文档上的准确性不佳，我们提出了 QAMR，这是一种旨在跨多版本系统文档回答问题的聊天机器人。QAMR增强了传统的检索增强生成（RAG），以确保在不同发布版本的文档高度相似但又存在差异的情况下仍能保持准确性。它通过预处理、查询改写和上下文选择的创新组合实现这一目标。此外，QAMR采用双重分块策略，使检索和答案生成能够分别使用经过调优的分块大小，从而提升整体问答准确性。**

**我们使用公开的软件工程基准测试，以及来自行业合作伙伴 Ciena 的真实多版本系统文档集合，对 QAMR 进行了评估。评估得出五项主要发现：(1) QAMR 的表现优于基于基线检索增强生成的聊天机器人，答案正确性平均达到 88.5%，检索准确率平均达到 90%，分别提升了 16.5% 和 12%。(2) 消融研究表明，QAMR 用于处理多版本文档的机制能够直接提高答案准确率。(3) 与移除相应组件的变体相比，QAMR 相对于表现最佳的消融版本，答案正确性平均提升 19.6%，检索准确率平均提升 14.0%。(4) 与基线相比，QAMR 的响应时间平均缩短了 8%。(5) 评估中使用的自动计算准确率指标与专家人工评估高度相关，验证了我们方法的可靠性。**

索引词——**问答、大语言模型、检索增强生成、多版本系统。**

## I. 引言

系统和软件工程流程依赖技术文档，例如需求规格说明、设计文档和安装指南。这些文档帮助最终用户、产品经理、软件工程师、系统架构师和客户支持团队等不同利益相关者理解系统功能和依赖关系，并履行各自职责。

公司通常会维护系统的多个版本，每个版本都在前代基础上不断演进，以满足客户需求、支持旧版用户、遵守法规或实现分阶段推出 [1]。这类共存版本的集合称为多版本系统。与软件产品线 [2]——一个有计划的相关产品系列这些产品共存，但并不构成连续演进阶段——多版本系统则逐个版本推进。每个版本通常都有自己的文档，记录该发布版本特有的更新、改进和配置变更，同时保留大部分早期内容。

如今，大语言模型（LLM）已广泛用于开发软件和系统工程领域的问答聊天机器人。这些聊天机器人帮助利益相关者搜索大量文档集，并实时获取问题的答案。Abdellatif 等人 [3], [4], Daniel 和 Cabot [5], Chaudhary 等人 [6], 以及 Huang 等人 [7] 等研究考察了聊天机器人在软件工程中的应用，包括开发和运行阶段，并展示了它们在改善沟通以及支持缺陷跟踪、文档编写、代码生成和运行时辅助等任务方面的潜力。

现代聊天机器人广泛采用的一种架构是检索增强生成（RAG）。RAG 将文档检索与大语言模型相结合，以生成更准确且与上下文更相关的回答 [8], [9], [10], [11]。将 RAG 应用于同一系统多个发布版本的相关文档时，会面临两个挑战：（**C1**）由于众多版本并存，且各版本内容既有差异又有重叠，RAG 的检索准确率会下降。例如，当查询涉及随发布版本而变化的系统属性时，RAG 的检索组件可能会混淆，提取出彼此不一致的信息。（**C2**）标准 RAG 无法有效地为多版本系统的上下文中的最相关信息赋予优先级。在这种情况下，并非所有文档或其部分内容都必然与特定发布版本相关联。为生成准确的答案，需要将查询重新解释为更具体的查询——其中一些与特定发布版本相关，另一些则与发布版本无关。重新解释会使 RAG 检索出多组文本，但这些文本的相关性并不相同。因此，需要一种策略，从根据不同查询解释检索到的文本中仅保留相关文本。

本文提出了 QAMR，这是一种新颖的、基于检索增强生成的**问**答聊天机器人，专为**多**次**发布**系统设计，旨在应对上述挑战。我们的方法包括一个预处理步骤：将文档按发布版本分别编入独立语料库，从而防止检索增强生成混合不同发布版本的信息。借助一种提示技术，QAMR 会将给定的用户查询转换为多个独立版本，使其充分包含对话历史。与传统的检索增强生成不同，后者检索

单组文本，QAMR 会检索多组文本，每组均使用改写后的独立查询。为了仅使用与原始用户查询最相关的文本来生成答案，QAMR 采用了最先进的、基于大语言模型的排序策略 RankRAG [12]。RankRAG 不会使用所有检索到的文档来生成答案，而是先利用大语言模型对文本进行排序，并筛选出与输入查询在上下文上最相关的文本，再生成答案。

除了支持多版本系统外，QAMR 还缓解了检索增强生成在软件工程应用中的一个已知局限 [11], [6]：对分块大小——即检索到的文本大小——的敏感性。较小的文本块通常能提高检索精确率，但为答案生成提供的上下文有限；较大的文本块则能提供更多上下文，但往往会降低检索准确率。QAMR 不再依赖单一的折中大小，而是采用双重分块 策略，在检索时使用较小的文本块，在答案生成时使用较大的文本块。

**贡献**。**(1)** 我们介绍了 QAMR，这是一款基于检索增强生成的聊天机器人，用于针对多版本系统进行问答。QAMR 能处理表格和示意图等结构化内容，并生成对用户查询的多个解读，以提高检索准确率。它采用 RankRAG 来识别与上下文相关、适合用于生成回答的信息。此外，QAMR 还实现了双重分块策略：检索时使用较小的文本块，答案生成时使用较大的文本块。这种设计避免了传统检索增强生成系统因依赖单一分块大小而固有的权衡。

**(2)** 我们在来自 Ciena 的行业数据集和开源需求集 REQuestA [13]上评估 QAMR。我们将 QAMR 与最先进的基线进行比较，并开展消融研究，以分离 QAMR 各个组件的影响。为此，我们报告了六项大语言模型评审指标，重点衡量答案正确性和检索准确率 [14],[15]，以及响应时间。我们采用大语言模型评审指标，是因为人工评估和传统词语重叠指标（如 BLEU [16],ROUGE [17], 和余弦相似度）都存在局限：人工评估虽然能够进行细致判断，但难以扩展，因为它需要检查由拟议的聊天机器人、其变体和基线生成的大量答案，还要检查多次运行的结果以考虑随机性。BLEU、ROUGE 和余弦相似度等自动化指标只能进行浅层的词语匹配，无法捕捉有效评估大语言模型生成文本所需的类人特质；已有研究表明，这些指标在这方面并不充分 [6], [18], [19], [20]。

我们采取两项措施，确保大语言模型计算得出的指标能够可靠地代表人类专家的判断。首先，我们与 Ciena 的主题专家合作创建了大语言模型评审框架，其中使用的提示词能够充分反映人工评估者在评估多版本系统中的响应质量时的推理方式。其次，我们将大语言模型生成的指标与 Ciena 领域专家的评估结果进行验证；这些专家会在完整的工业数据集上，手动评估 QAMR 单次运行所得响应的准确性。我们的分析显示，大语言模型计算得出的指标与专家评估之间具有很强的相关性，这证实了我们指标的可靠性。

**研究发现**。 我们的评估结果表明，QAMR 优于不采用双重分块的基线检索增强生成聊天机器人，答案正确性平均达到 88.5%，检索准确率达到 90%。与基线相比，这两项指标分别平均提升了 16.5% 和 12%。

我们的消融研究得出了两个主要发现：首先，QAMR 的各个组成部分相辅相成。虽然每个组成部分单独使用都能提高准确性，但只有包含所有组成部分的完整 QAMR 才能持续实现最佳性能；与表现最佳的消融版本相比，答案正确性平均提升 19.6%，检索准确率平均提升 14.0%。其次，在多个系统版本之间重新解读用户查询的能力，对于在多版本文档上有效开展问答至关重要，因为具备这一能力的消融版本，其答案正确性和上下文召回率均高于不具备这一能力的消融版本。

在响应时间方面，针对 Ciena 和 REQuestA 两个数据集，QAMR 的平均响应时间分别缩短约 9% 和 8%，与基线相比，整体提升约 8%。

**创新性**。我们的工作创新之处在于，为多版本系统设计了一款基于检索增强生成的聊天机器人，以应对文档各不相同但又相互重叠所带来的挑战。此外，我们提出了一种分块策略，通过结合用于检索的小文本块和用于答案生成的大文本块，缓解检索准确率与上下文丰富度之间的权衡。

**重要性**。 多版本系统在行业中十分普遍，因为组织往往需要应对不断演进、拥有多个活跃版本的产品。实用且准确的聊天机器人能够帮助用户从与此类系统相关的复杂技术文档中高效获取所需信息。另一个重要之处是我们的双重分块方法，它有望在更广泛的场景中提高聊天机器人准确性，值得在多版本系统之外进一步研究。

**复现实验包**。我们公开数据集的所有代码、评估脚本和实验数据均可在线获取 [21]

## II. 行业背景与研究动机

Ciena 正在探索利用聊天机器人自动执行日常任务，重点是管理其复杂的多发布系统产品组合。Ciena 的每个发布版本都需要更新特定版本的文档，涵盖需求、架构、配置和安装。各个发布版本之间虽然存在大量重叠内容，但也有重要差异。这些文档服务于不同的利益相关者——工程师、开发人员和产品经理——他们都需要查询单个或多个发布版本的文档。例如，软件开发人员可能会询问有关某个产品特定发布版本的问题，例如：“如何将 *[产品]* 更新到发布版本 *17*？”。要准确回答这个问题，必须从与发布版本 17 对应的文档中检索信息。类似地，要回答“如何升级到 *[产品]* 发布版本 *12*？”，则需要查阅发布版本 12 的文档。

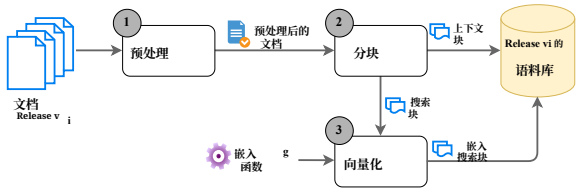



图 1：QAMR 为每个文档版本创建语料库

发布版本相关问题之间的相似性，以及不同发布版本文档之间的重叠内容，使查询恰当的文档对于生成正确答案至关重要。或者，开发人员也可能提出一个答案不受特定发布版本限制的通用问题，例如：“如何连接到 *[产品]*的仪表板？”。在这种情况下，Ciena 希望保留灵活性，提供不受版本限制且适用于多个发布版本的答案。所有查询和示例中的原始内容都经过了少量修改，以确保机密性；删节内容以方括号标示（[...]）。

本文旨在利用最先进的大语言模型技术开发一款聊天机器人，使 Ciena 的利益相关者能够查询结构复杂、语义丰富的系统文档，同时兼顾公司系统的多版本特性。尽管我们的工作源于 Ciena 的具体需求，但我们所应对的挑战并非该公司独有。各行业中多版本系统的普遍存在，要求我们开发能够处理不断演变的软件文档复杂性的方案。认识到这一工作的广泛适用性后，我们设计聊天机器人时着重考虑了其通用性。

## III. 多版本系统聊天机器人

QAMR 以多版本文档和查询为输入，并根据文档生成答复。III-A 节介绍语料库构建，III-B 节则介绍 QAMR。

### *A.* 语料库构建

给定一组多版本文档，QAMR 会为每个发布版本分别创建一个语料库，而不是将所有版本合并到单个语料库中。此方法可确保不同发布版本之间的差异不会影响大语言模型在答案生成过程中的准确性。QAMR 根据用户查询选择最相关的语料库进行文档检索（见 III-B 节）。为将特定发布版本的文档转换为语料库，我们按照图 1 所示的三个步骤进行，具体如下。

步骤***1***：预处理。为了准备供大语言模型处理的文档，我们会从所有组件中提取文本，包括 HTML 或 SVG 格式的表格、示意图和流程图。我们会移除页眉、页脚和版权信息等冗余元素，以防止 QAMR 因匹配到相关术语而检索到无关页面。文档名称
会添加到每个页面的元数据中。虽然生成答案并不需要这些元数据，

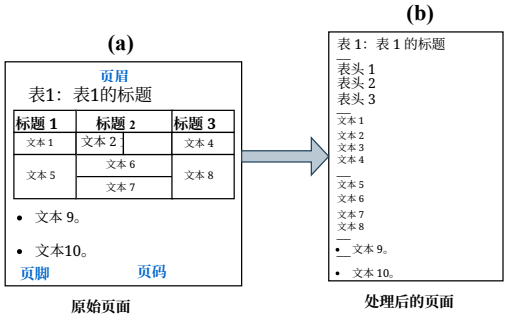



图 2：QAMR 对单个页面的预处理

**算法1** QAMR 的双重分块算法

**输入** 文档：来自图 1 步骤1 的一组已处理文档
**参数** $k$：每个页面的搜索块数量
**参数** $ps$：填充大小
**输出***S-Chunks*：一组搜索块
**输出***C-Chunks*：一组上下文块

*C-Chunks* =*S-Chunks* ← ∅
**对于**每个 文档$_i$ ∈ 文档**执行**
  **对** 每个 $pg^i{}_j$ ∈ $doc_i$ **执行**//构建搜索块
    将$pg^i{}_j$ 划分为 $k$ 个大小相等的搜索块$sc^i{}_j{}_1$、……、$sc^i{}_j{}_k$
    *S-Chunks* ← *S-Chunks* ∪{$sc^i{}_j{}_1$、……、$sc^i{}_j{}_k$}
  **结束循环**
  **对** 每个 $pg^i{}_j$ ∈ $doc_i$ **执行**//构建上下文块
    $cc_i{}_j$ ←$pg^i{}_j$ //初始化上下文块$cc^i{}_j$
    添加 $ps$ 从$pg^i$的底部添加字符，$j-1$ 到$cc^i{}_j$的开头
    添加 $ps$ 从$pg^i$的顶部添加字符，$j+1$ 到$cc^i{}_j$的末尾
    元数据（$cc_i{}_j$） ← doc$_i$的标题信息
    *C-Chunks* ← *C-Chunks* ∪{$cc^i{}_j$}
  **结束于**//每个搜索块$sc_{ijl}$ 与上下文块$cc_{ij}$相关.
**结束循环**
**返回***S-Chunks*，*C-Chunks* //返回搜索块和上下文块

但它们能确保 QAMR 的回答与原始文档之间具有可追溯性。表格、图表和示意图中的文本按从左上到右下的顺序提取，并重构为行，以保留水平和垂直顺序。不同部分之间使用分隔符。图2展示了我们的预处理流程：移除页眉、页脚和页码；按从左上到右下的顺序提取表格文本，将其转换为行，并用“—”分隔。

步骤***2***：分块。步骤1处理后的文本被划分为更小的片段，称为文本块 [11], [8], [6]。分块的作用不仅仅是让文本符合大语言模型的词元上限：对于搜索，较小的文本块往往能提高检索准确率。而对于答案生成，选择尽可能大的文本块（即大语言模型的词元上限）并不一定是最佳选择，因为存在“大海捞针”问题 [22], [23],[24], [11]。为考虑这些因素，我们将搜索和答案生成的分块大小解耦，使这些大小能够明确且独立地配置。我们生成两组不同的文本块：一组用于搜索，另一组用于答案生成。我们将这种方法称为双重分块，详见算法1。

算法1以步骤1中的一组已处理文档（记为文档）作为输入。对于文档中的每份文档，该算法会生成两类输出：一组搜索块（*S-Chunks*）和一组上下文块（*C-Chunks*）。较小的搜索块用于语义相似度

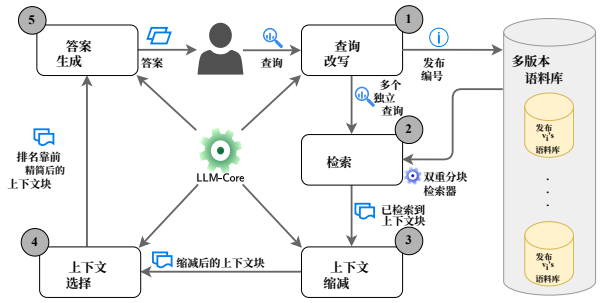



图3：QAMR架构。

搜索，以准确定位相关文本。确定最相关的搜索块后，会为每个搜索块检索对应的（较大）上下文块，以生成答案。重复的上下文块会被移除。

算法1有两个可配置参数：$k$，即每个页面要创建的搜索块数量；以及$ps$，即用于构建上下文块的填充大小。为创建搜索块，每个文档页面被划分为 $k$ 个块，随后将这些块添加到*S-Chunks*中（第3–6行）。每个上下文块都通过扩展页面来创建：在页面开头添加其前一页面底部的 $ps$ 个字符，并在页面末尾添加其后一页面顶部的$ps$ 个字符（第8–10行）。因此，上下文块与页面之间存在一一对应关系。为保持可追溯性，每个上下文块的元数据被设置为其对应文档的标题信息（第11行）。最后，将创建的上下文块添加到*C-Chunks* 中（第12行）。处理完*Docs*中的所有文档后，返回*S-Chunks* 和*C-Chunks*（第15行）。

步骤***3***：向量化。我们使用嵌入函数为每个搜索块生成嵌入向量。所得向量在高维空间中以数值形式表示这些块，并捕捉其语义 [25]。这些向量存储在向量数据库中。需要指出的是，在我们的方法中，上下文块无需进行向量化。如算法1所示，搜索块$sc_{ij}$映射到其对应的页面$pg_{ij}$，而该页面又映射到相关的上下文块$cc_{ij}$。因此，当某个搜索块被判定为与查询相关（通过语义相似度）时，可以直接确定其对应的上下文块。

### *B.* 聊天机器人架构

图 3 概述了 QAMR，它包含五个步骤，用于根据第 III-A 节创建的语料库生成对用户查询的回复。图 3 中的步骤 1 、 3 、 4 和 5 需要使用大语言模型。尽管每个步骤都可以使用不同的大语言模型，但考虑到这些步骤按顺序执行，为提高效率，我们使用单个大语言模型实例。图中及本文后续部分均将此实例称为 LLM-Core。工作示例 [26] 和 QAMR 的提示词 [27] （因篇幅所限，此处省略）可在线获取。

步骤***1***：查询改写。QAMR 首先从用户查询中提取发布编号（如果存在）。提取出的
发布编号既用于查询改写过程，也用于

**用户查询 #1：**
产品[的可接受传输范围是多少]?
**查询改写输出 #1：**
***Base1***： [产品]?
***Filtered1***： [产品]
发布编号： 无

**用户查询 #2：（后续问题）**
它在[发布]中的配置规则是什么?
**查询改写输出 #2：**
***Base2***： [产品] 在[发布]有哪些配置规则?
***Filtered2***： [产品] 在[发布]配置规则
***Versionless2***： [产品] 配置规则
发布编号： [发布]

图4：QAMR 中查询改写的示例。

在步骤2中确定用于文档检索的合适语料库。为提取发布编号，QAMR 会向LLM-Core提供提示词。这样，我们就能将用户查询中的各种表达形式正确映射到预期的发布版本。例如，“R17.2”、“Rel 17.20”或“Release 17.2”等不同表达都会被映射为“Release 17.20”，因为文档中采用了这一写法。与受固定模式限制的正则表达式相比，使用大语言模型进行这种映射能够提供更高的准确性。

提取发布编号（如果存在）后，QAMR 会使用用户查询、对话历史和提取出的发布编号来改写查询。如果识别出发布编号，QAMR 会生成三个独立查询；否则会生成两个。独立查询是指内容完整、无需额外上下文即可理解的查询。查询改写通过三个在线提供的提示词完成 [27]。如下文所述，我们将生成的三个独立查询称为*Base*、*Filtered*和*Versionless*。

**此*Base* 查询**会结合用户对话历史的上下文来解析用户查询。在交互式设置中，用户经常会提出不够明确的问题，尤其是依赖先前上下文的后续问题，例如使用“它”或“他们”等代词，或“前一个”之类的短语 [28]，[6], [11]。图 4展示了两个*Base*查询的示例。第一个示例，$Base_1$，与用户查询完全相同，因为该查询本身已包含完整信息。相比之下，第二个示例，$Base_2$,将代词“它”替换为 [产品]，该信息来自对话历史。

**此*Filtered* 查询**会从*Base* 查询（上文）中移除停用词和其他常见词，聚焦于核心要素。生成的查询通常与文档表格中的内容和用语更加接近，从而提高准确性，尤其是在处理表格密集型文档时。例如，在图4中，$Filtered_1$ 和 $Filtered_2$ 分别表示相应用户查询的*Filtered* 查询。

**该*Versionless* 查询**仅当用户查询中包含发布编号时才会创建：从上文的*Filtered* 查询中移除发布编号。*Versionless* 查询针对可能导致检索不准确的两种情况：（1）相关文档中明确提及了发布编号，但用户查询中没有；或（2）用户查询中明确包含发布编号，但相关文档中没有。

发布编号在用户查询中明确包含，但相关文档中没有。例如，在图4中，$_2$, 是通过从$_2$,中省略发布编号而形成的*Versionless*查询，是针对第二个用户查询生成的第三个独立查询。相比之下，对于第一个不包含发布编号的用户查询，只会生成两个独立查询。

步骤***2***：检索。 在此步骤中，我们首先使用两种广泛采用的评分方法——余弦相似度和最大边际相关性搜索 [29], [30]——通过合并与给定独立查询匹配的得分最高的（搜索）块，检索最相关的搜索块。用于检索的语料库由用户查询中提及的发布编号（如步骤1中所提取）决定。如果未提供发布编号，则使用最新语料库生成响应。由于存在多个独立查询，因此将检索到多组搜索块。我们对这些搜索块取去重后的并集。接下来，利用算法1中保留的搜索块与上下文块之间的可追溯性信息，将搜索块映射到对应的上下文块。最后，我们使用一组去重后的最相关上下文块，继续执行下文所述的步骤3，并丢弃所有冗余内容。

步骤***3***：上下文缩减。 如前文所述，大型上下文块即使处于大语言模型的词元上限以内，也可能造成“大海捞针”问题：无关细节会掩盖关键信息，妨碍大语言模型准确地进行答案生成 [23], [22], [24], [11]。为解决这一问题，QAMR 使用少样本思维链（CoT）提示词 [27]，从每个检索到的上下文块中仅提取与用户查询相关的信息。这确保每个独立块仅包含与查询相关的信息。每个精简后的块都会被保留，除非它为空，这表明原始块不包含相关信息。

步骤***4***：上下文选择。QAMR 根据缩减后的上下文块对回答相应独立查询的有用性，从中识别出最相关的缩减后的上下文块。与步骤3类似，这一过程通过少样本 CoT 提示 [27] 实现。我们在此使用的提示受 Yu等人 [12] 提出的提示启发。随后选择排名靠前的缩减后的上下文块。

步骤***5***：答案生成。QAMR 使用步骤4中排名靠前的缩减后的上下文块来回答查询。提示 [27] 指示 LLM-Core 仅依赖这些文本块，而不是其内部知识；如果这些文本块缺乏回答查询所需的足够信息，则回复“我不知道”，从而降低幻觉风险。

## IV. 实证评估

我们的评估旨在回答三个研究问题：

**研究问题1（准确性）**。*QAMR* 的准确性如何？研究问题1使用来自 Ciena 的专有数据集和开源数据集 REQuestA [13]评估 QAMR 的准确性。我们使用六项大语言模型评审指标 [14], [15], 将 QAMR 与第四节 C 部分所述的基线进行比较，这些指标用于衡量答案准确率和检索准确率。为确保这些大语言模型生成的指标的可靠性，我们进行相关性分析，以通过专家人工判断对这些指标进行验证。

表 I：数据集的汇总统计。

| | Ciena | REQuestA |
|---|---|---|
| 文档数量 | 59 | 6 |
| 每份文档的平均字符数 | 887,559 | 119,741 |
| 每份文档的平均页数 | 531 | 65 |
| 问题数量 | 88 | 159 |
| 每条用户查询的平均字符数 | 74 | 65 |
| 每条真实答案的平均字符数 | 240 | 528 |
| 非文本元素数量 | 5694 | 20 |

**RQ2（消融研究）**。*QAMR* 的每个步骤如何影响其整体准确性？在 QAMR 的五个步骤中，最后一步——答案生成——对于任何问答聊天机器人都至关重要。然而，其他四个步骤旨在提升准确性，尤其是在多版本系统中。RQ2 探究这四个步骤分别如何促进 QAMR 的整体准确性。

**RQ3（响应时间）**。*QAMR* 的执行时间是多少？RQ3 比较了评估中两个数据集上 QAMR 和基线的响应时间。

### *A.* 数据集与真实答案

我们的评估使用两个数据集：Ciena 的光网络和电信系统规格说明文档，以及 REQuestA——一个涵盖航空航天、国防和安全软件需求的公开数据集 [13]。

Ciena 数据集是两个数据集中规模较大的一个，具有 REQuestA 所不具备的多版本特征。在评估中纳入 REQuestA 有三个目的：(1) 确认 QAMR 能够推广到 Ciena 数据集之外，并在不同类型的文档中保持准确性；(2) 系统地评估双重分块的有效性，而不受是否存在多版本特征的影响；(3) 提供公开的复现包，因为 Ciena 数据集具有专有性质，无法公开发布。

表 I 汇总了两个数据集的统计信息。Ciena 数据集包含 59 份多版本系统规格说明，平均每份有 531 页和 887,559 个字符，并含有 5,694 个非文本元素。相比之下，REQuestA 数据集由 6 份软件需求规格说明组成，平均每份有 65 页和 119,741 个字符，且仅含 20 个非文本元素。

Ciena 文档附有真实值，其中包含 88 个问答对，问题平均为 74 个字符，答案平均为 240 个字符。该数据集是公司为评估聊天机器人项目而开发的内部基准的一部分。所有问题和答案均由主题专家精心编写，他们均非本文的共同作者。REQuestA 的真实值由 159 个定义明确且经人工验证的问答对组成 [13]，问题平均为 65 个字符，答案平均为 528 个字符。

### *B.* 评估指标

我们使用以下六项大语言模型评审指标来评估 QAMR：答案正确性、答案相关性、答案忠实性、上下文精确率、上下文召回率和上下文

相关性。答案正确性使用 G-Eval [15],进行衡量，而其他五项使用 DeepEval [14]。所有指标均依赖大语言模型作为评估器。下面介绍这六项指标：

令 $Q$ 为我们评估中的用户查询集合， $R$ 为生成的答案集合， $G$ 为 $Q$的真实答案集合。对于每个查询 $q \in Q$，令 $r_q \in R$ 表示生成的答案， $g_q \in G$ 表示真实答案。此外，令$C_q = (c_1, \ldots\ldots, c_l)$表示用于生成查询 $q$答案的有序文本块列表。我们将用于计算准确率指标的底层大语言模型称为LLM-Eval。在整个评估过程中， $q$ 始终指查询改写前的原始用户查询。

*1*）答案正确性：该指标使用LLM-Eval ，并依赖领域专用提示词，模拟人工评估者在聊天机器人的应用场景 [15]中评估生成回答质量时所采用的推理方式。我们与 Ciena 合作开发了八个提示词，详见在线材料 [27], ，旨在复现专家的推理过程。这些提示词分为四组：

**事实准确性（两个提示词）**确保对于每个问题 $q \in Q$， $r_q \in R$ 和$g_q \in G$ 在事实层面保持一致。

**单位和值匹配（两个提示词）**确保$r_q$ 和$g_q$之间的值或单位不存在不匹配，例如将秒误用作接收器损坏阈值的分贝毫瓦。

**核心含义等价性（两个提示词）**确保忽略$r_q$ 和$g_q$ 在详略程度或信息呈现顺序上的差异，只要 $r_q$ 和$g_q$ 传达相同的核心含义。例如，考虑一个问题 $q$ ，询问特定产品可以连接到哪些引脚。假设$g_q$ 明确列出了引脚编号，而 $r_q$ 提供了确定这些引脚的指导方针，使工程师能够根据这些指导方针识别它们。在此示例中，我们的提示词确保$r_q$ 被视为正确答案。

**范围识别（两个提示词）**确保：(1) 对于超出范围的查询（其中 $g_q$ 表明不存在答案），$r_q$明确承认不存在答案；以及(2) 对于范围内的查询，$r_q$ 不声称不存在答案。

给定一组领域专用提示词，这些提示词会生成“正确”或“错误”的判定，G-Eval 会为每个查询 $q$ 评分（介于 0 和 1 之间），评估 $r_q$ 与 $g_q$ [15]。

*2*）答案相关性：对于查询 $q$，答案相关性使用 LLM-Eval 将 $q$ 生成的回答 $r_q$ 中的陈述分类为与 $q$ 中的陈述相关或不相关。设 $A$ 为 $r_q$ 中陈述的总数， $B$ 为 $r_q$ 中与 $q$ 相关的陈述数。答案相关性定义为 $\frac{B}{A}$，用于衡量答案陈述中与 $q$ 相关的比例。

*3*）答案忠实度： 对于查询 $q$，答案忠实度使用 LLM-Eval 将 $q$ 的生成回答 $r_q$ 中的陈述分类为相关或不相关，判断其是否与 $C_q = (c_1、\ldots\ldots、c_l)$ 中任一文本块的至少一个陈述相关；这些文本块用于回答 $q$。设 $A$ 为 $r_q$中的陈述总数， $B$ 为$r_q$ 中能找到相关陈述的陈述数。答案忠实度定义为 $\frac{B}{A}$，用于衡量答案中的陈述与用于回答 $q$ 的文本块相关的比例。

*4*）上下文精确率： 对于查询 $q$，上下文精确率用于衡量回答该查询所用文本块 $C_q = (c_1、\ldots\ldots、c_l)$ 的相关性。此指标使用LLM-Eval 将每个文本块 $c_i$ 分类为与 $q$ 相关或不相关，方法是将$c_i$ 与 $q$ 的真实答案（即 $g_q$）进行比较。对于每个 $c_i$，设 $b_i$ 为一个二元变量：当 $c_i$ 与$g_q$ 相关时取值为 1，否则取值为 0。 $q$ 的上下文精确率定义如下：

$$\text{Contextual precision} = \frac{\sum_{i=1}^{l}\left(\frac{\#\text{ of relevant chunks in sublist}(c_1,\ldots,c_i)}{i} \times b_i\right)}{\#\text{ of relevant chunks in } C_q}$$

上下文精确率更重视排名较高（接近 1）的上下文，而非排名较低（接近 $l$）的上下文。

*5*）上下文召回率：对于查询 $q$，上下文召回率使用 LLM-Eval将 $q$ 的真实答案 $g_q$ 中的陈述分类为与用于回答 $q$ 的文本块 $C_q = (c_1, \ldots\ldots, c_l)$ 相关或不相关。设 $A$ 为 $g_q$ 中陈述的总数， $B$ 为 $g_q$ 中与某个文本块 $c_i$ 中的某条陈述相关的陈述数。上下文召回率定义为 $\frac{B}{A}$ ，用于衡量真实答案中的陈述被 $C_q$ 覆盖的比例。

*6*）上下文相关性：对于查询 $q$，上下文相关性使用 LLM-Eval将用于回答 $q$ 的文本块$C_q = (c_1, \ldots\ldots, c_l)$中的陈述分类为与 $q$ 相关或不相关。设 $C_q$中的陈述总数为 $A$ ，设 $C_q$ 中与 $q$ 相关的陈述数量为 $B$ 。那么，上下文相关性定义为 $\frac{B}{A}$，用于衡量文本块中的陈述与$q$ 相关的比例。

### *C.* 消融实验与基线

我们为 QAMR 设计了五种消融实验，此外，还将 QAMR 与最先进的基线进行比较。这五种消融实验分别对应 QAMR 的四个步骤——查询改写、双重分块、上下文缩减和上下文选择——以及一种排除所有这些步骤的消融实验。具体而言，我们构建了：（i）一种不包含任何中间步骤、但包含答案生成的消融实验；（ii）四种各自仅包含一个中间步骤以及答案生成的消融实验。这些消融实验使我们能够单独评估 QAMR 各个组件的影响。尤其是，包含查询改写的消融实验使我们能够评估该步骤在提高多版本文档答案正确性方面的效果。

为了与现有方法进行比较，我们采用了软件工程领域最先进的基于 RAG 的问答聊天机器人的架构 [11], [6], [28], [12], ，并对其进行调整，加入 QAMR 中的查询改写和上下文选择步骤。通过这一调整，基线能够处理多版本文档；值得注意的是，现有聊天机器人以当前形式并不适合用于此类文档。我们的基线实现已在线提供 [21]。

### *D.* 实现

对于 QAMR 和基线，我们均使用 Meta 的指令微调 Llama 3.0 模型，该模型具有 70B 参数（Llama-3-70B-Instruct） [31]。Llama 3-70B 在效率与强大的指令遵循能力之间实现了平衡，同时仍保持

表 II：QAMR 和基线的关键参数。

| | 分块 | | 检索 | |
|---|---|---|---|---|
| QAMR | 搜索块数量<br>每页 *(k)* | 填充大小 *(ps)* | 检索块数量<br>每个独立查询 | 用于答案生成的块数量<br>答案生成 |
| | 2 | 500个字符 | 4 | 3 |
| 基线 | 分块大小<br>每个不超过 3000 个字符的页面作为一个文本块 | | | |

适用于本地部署 [31] ——这是我们的行业背景所要求的。我们使用 Python 3.10 和 Transformers 库（v4.45.1）[32]实现 QAMR 和基线，用于加载模型和进行语言分词。为尽量降低计算开销，LLM-Core 和 LLM-Eval 均使用 BitsAndBytes（v0.43.1） [33]以 4 位量化格式加载。文档索引方面，我们使用 BAAI/bge-base-en-v1.5 模型，该模型可生成高质量且高效的语义嵌入，因此非常适合大型语料库 [34], [35]。双重分块方法通过 LangChain 的多向量检索器 [36], 实现，向量存储在 Chroma 向量存储中 [37]。我们使用 LangChain（v0.2.8） [30] 作为底层主要框架，并利用 LangChain 表达式语言（LCEL） [38] 构建聊天机器人管道。我们的实现已在线发布 [21]。

### *E.* 实验过程

我们的实验在一台配备两颗 Intel Xeon Gold 6338 处理器、512 GB 随机存取存储器和一块 NVIDIA A40 GPU（46 GB 显存）的机器上进行。为减小随机性，每项实验均重复十次。我们总共在两个数据集上提出了 4,940 个查询，并使用第 IV-B 节所述的六项指标对响应进行了评估。

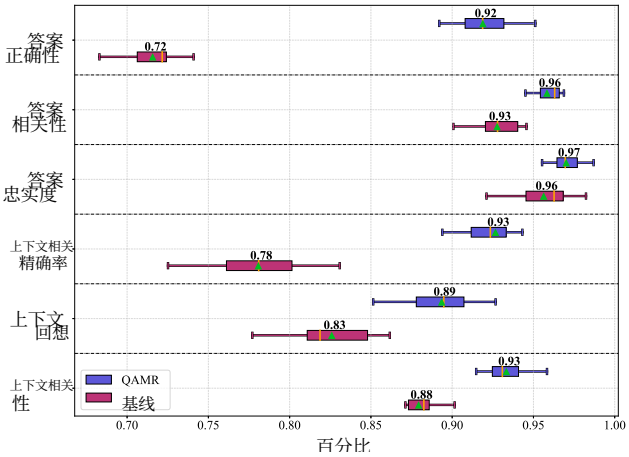
表 II 展示了配置 QAMR 和基线所需的参数。QAMR 的双重分块器（算法1）需要两个参数：$k$ 和$ps$。我们设置$k = 2$，将每个页面划分为两个搜索块，通过避免过大或过小的块，在搜索准确性和有效性之间取得平衡。对于填充大小$ps$，我们使用 500个字符，以提供足够的前置和后续上下文（约为页面的 1/3 至 1/4）。对于基线，其对搜索和答案生成使用相同的块，因此我们将每个页面视为单个文本块；对于超过 3,000 个字符的页面，则将其拆分为更小的块。这些块与前一个和后一个块均保持 25% 的重叠，遵循先前关于传统（单块）检索增强生成系统的研究建议 [6], [13]。

如表 II 所示，QAMR 和基线对每个独立查询均检索四个文本块；其中两个通过余弦相似度检索，
另外两个则通过最大边际相关性进行选择。随后，答案生成步骤使用排名前三的文本块生成回复，这与先前的研究一致 [6], [13]。

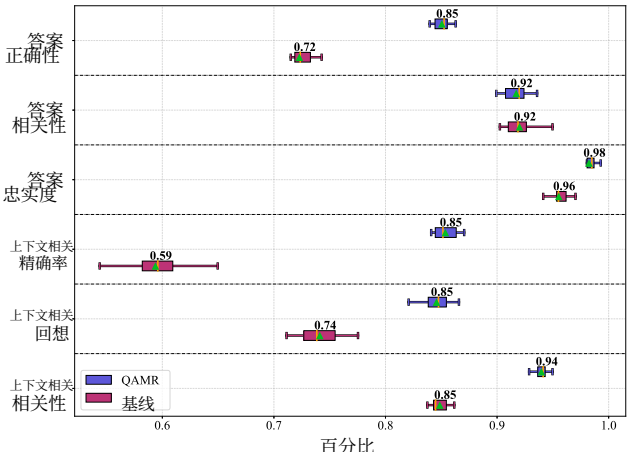
我们使用表 II 中针对 Ciena 数据集所述的 QAMR 相同设置，对第 IV-C 节讨论的五项消融实验进行评估，并将每项实验重复十次。

### *F.* 结果

在本节中，我们展示研究发现并回答 RQ1-RQ3。为了为分析提供统计支持，我们使用



（a）Ciena 数据集



（b）REQuestA 数据集

图 5：使用第 IV-B 节定义的指标对 QAMR 和基线进行比较。

Wilcoxon 秩和检验 [39]和 Vargha–Delaney 效应量 ˆ大小 ($A_{12}$) [40]，显著性水平设为 1%。当 $p$值低于此阈值时，差异被认为具有统计显著性。当 $A_{12}$ 与 0.5 的偏差至少为 0.06、0.14 和 0.21 时，效应量分别归类为小、中或大。

**研究问题1（准确性）**。我们使用第 IV-B 节定义的六项指标来衡量准确性。图 5 比较了在 Ciena 和 REQuestA 数据集上由 QAMR 和基线得到的六项指标的分布。对于 Ciena 数据集（图 5(a)），在全部六项指标上，QAMR 的平均值均高于基线。对于 REQuestA 数据集（图 5(b)），QAMR 在其中五项指标上的平均值更高，而在答案相关性方面，QAMR 和基线产生了相同的平均值。表 III 中的统计检验表明，在 Ciena 数据集上，除答案忠实度外，QAMR 在所有指标上均以较大的效应量显著优于基线；在 REQuestA 数据集上，除答案相关性外，QAMR 在所有指标上均以较大的效应量显著优于基线。

在两个数据集上，QAMR 的答案正确性平均得分为 88.5%，上下文精确率为 89%，上下文召回率为 87%，上下文相关性为 93.5%。

表 III：QAMR 和基线的统计检验结果。除黄色高亮的单元格（未观察到统计显著性）外，QAMR 的表现显著优于基线，且效应量较大。

| | Ciena | | REQuestA | |
|---|---|---|---|---|
| | p 值 | $\hat{A}_{12}$ | p 值 | $\hat{A}_{12}$ |
| 答案正确性 | 0.002 | 1 (L) | 0.002 | 1 (L) |
| 答案相关性 | 0.006 | 0.83 (L) | 0.84 | – |
| 答案忠实度 | 0.08 | – | 0.002 | 0.98 (L) |
| 上下文精确率 | 0.002 | 1 (L) | 0.002 | 1 (L) |
| 上下文召回率 | 0.004 | 0.98 (L) | 0.002 | 1 (L) |
| 上下文相关性 | 0.002 | 1 (L) | 0.002 | 1 (L) |

表 IV：第 IV-B 节中各指标与领域专家提供的充分性标签之间的皮尔逊相关系数。绿色单元格表示 p 值 $\ll$ **0。01**。

| 数据集 | 指标 | 相关性 |
|---|---|---|
| Ciena | 答案正确性 | 0.88 |
| | 上下文精确率 | -0.02 |
| | 上下文召回率 | -0.01 |
| REQuestA | 答案正确性 | 0.75 |
| | 上下文精确率 | 0.63 |
| | 上下文召回率 | 0.54 |

这些结果表明，与基线相比，答案正确性平均提高了 16.5%。对上下文精确率、召回率和相关性的提升幅度取平均后，整体较基线提高了 12%。

如第 I 节所述，鉴于我们的评估规模以及进行全面人工评估的不可行性，我们使用大语言模型评审指标，而不是专家判断。由此产生的一个重要问题是，我们的大语言模型评审指标与专家判断的吻合程度如何。为此，我们邀请 Ciena 专家（并非本研究作者）评审 QAMR 在完整 Ciena 数据集（全部 88 个问题）上生成的回答。对于该数据集中的每个问题所进行的 10 次运行，我们随机选取其中一次运行，专家根据其回答与真实答案的匹配程度，将答案标记为“合格”或“不合格”。同样，我们随机选取 QAMR 在 REQuestA 数据集上的一次运行，并邀请一名独立标注员（同样不是本研究作者）对该数据集 159 个问题所生成的回答进行标注，以确定生成的答案与真实答案的匹配程度。随后，我们计算了人类提供的标签与大语言模型评审指标值之间的皮尔逊相关系数，结果如表 IV 所示。在此分析中，在第 IV-B 节讨论的六项指标中，我们仅考虑将生成的答案与真实答案进行比较的指标，即答案正确性、上下文精确率和上下文召回率。如表中所示，答案正确性与人类判断高度相关（Ciena 为 0.88，REQuestA 为 0.75，p 值 $\ll$ 0.01）。对于 REQuestA，上下文精确率和上下文召回率也与人类判断呈正相关。

**发现1.** QAMR 在所有答案准确率和检索准确率指标上均显著优于基线，但 Ciena 数据集上的答案忠实度和答案相关性（REQuestA 数据集）除外。值得注意的是，与基线相比，QAMR 的答案正确性平均提高了 16.5%，上下文精确率、召回率和相关性平均提高了 12%。

**发现2.** 答案正确性指标与两个数据集的人工评估结果均呈现强正相关。**要点1.** QAMR 的双重分块比传统分块更常识别出相关文本块。结合上下文缩减，可确保答案生成仅使用相关信息。

**要点2.** 在两个数据集中，答案正确性指标与人工判断之间均存在强相关性，这表明在我们的应用场景中，该指标是可靠的替代指标，可作为人工评估的一种可扩展替代方案，用于验证大语言模型生成的文本是否符合真实值。

表 V：消融研究的平均评估结果（%）。*Base*消融仅使用 QAMR 的答案生成步骤。高亮单元格标示了各项指标中表现最佳的消融设置。

| | Base | + 查询改写 | + 双块文本块 | + 上下文减少 | + 上下文选择 |
|---|---|---|---|---|---|
| 答案正确性（%） | 68.29 | 72.28 | 72.13 | 59.30 | 64.39 |
| 答案相关性（%） | 93.56 | 92.64 | 93.67 | 78.07 | 90.69 |
| 答案忠实度（%） | 95.04 | 94.95 | 96.24 | 97.91 | 95.09 |
| 上下文精确率（%） | 67.46 | 71.25 | 57.77 | 58.88 | 72.39 |
| 上下文召回率（%） | 73.26 | 74.76 | 63.93 | 61.45 | 73.25 |
| 上下文相关性（%） | 82.10 | 78.82 | 75.97 | 74.68 | 87.63 |

**RQ2（消融研究）**。表 V 展示了使用 Ciena 数据集进行消融研究所得的结果。我们将仅使用答案生成步骤的消融称为*Base*消融。其余四种消融分别在 Base 的基础上，仅结合 QAMR 以下步骤中的一个：查询改写、双重分块、上下文缩减或上下文选择。复现实验包中提供了箱线图，展示表 V 中各项平均值所依据的数据分布 [41]。

图 5(a) 和表 V 中的结果表明，与五种消融实验相比，QAMR 显著提高了答案正确性、答案相关性、上下文精确率、上下文召回率和上下文相关性的平均值，最低提升幅度分别为 19.60%、2.16%、20.27%、14.59% 和 5.70%。此外，QAMR 在答案忠实度指标上的表现优于或不逊于消融实验。

表VI中的统计检验显示，除答案相关性和答案忠实度指标外，QAMR在所有指标上均显著优于所有消融实验，且效应量较大。对于答案相关性，QAMR显著优于两种消融实验；对于答案忠实度，QAMR显著优于一种消融实验。在这两种情况下，差异的效应量均较大。对于答案忠实度和答案相关性的其他所有比较，QAMR与其消融实验之间均无统计学显著差异。因此，根据我们的统计检验，在任何指标上，QAMR都不会被其任何消融实验超越。

表VI：表VI中 QAMR 与各项消融实验的统计检验结果。除黄色高亮的单元格外，QAMR 的表现均显著优于所有消融实验，且效应量较大；黄色高亮的单元格中未观察到统计显著性。

| | Base | | +查询改写 | | + 双块 | | + 上下文缩减 | | + 上下文选择 | |
|---|---|---|---|---|---|---|---|---|---|---|
| | P-值 | $\hat{A}_{12}$ | P-值 | $\hat{A}_{12}$ | P-值 | $\hat{A}_{12}$ | P-值 | $\hat{A}_{12}$ | P-值 | $\hat{A}_{12}$ |
| 回答正确 | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) |
| 回答相关 | 0.064 | - | 0.080 | - | 0.002 | 0.75（L） | 0.002 | 1(L) | 0.020 | - |
| 答案事实性。 | 0.064 | - | 0.002 | 0.84（L） | 0.080 | - | 0.200 | - | 0.014 | - |
| 上下文精确率 | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) |
| 上下文召回率 | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) |
| 上下文相关性 | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) | 0.002 | 1(L) |

将基础消融与四种非基础消融（每种都纳入了 QAMR 的一个单独步骤）进行比较时，我们发现，每种非基础消融至少在一项指标上优于基础消融。具体而言，与基础消融相比，查询改写消融提升了答案正确性、答案忠实度、上下文精确率和上下文召回率；双重分块消融提升了答案正确性、答案忠实度和答案相关性；上下文选择消融提升了上下文精确率、上下文召回率和上下文相关性；上下文缩减消融提升了答案忠实度。值得注意的是，基础消融在任何指标上的平均值都不是最高的。

**发现1**。 在任何指标上，消融实验的表现都没有显著优于 QAMR；而在六项指标中的四项或五项上，QAMR 的表现显著优于所有消融实验。此外，每个非基础消融版本至少在一项指标上优于基础版本，表明每个步骤——查询改写、双重分块、上下文缩减和上下文选择——都对整体方法有所贡献。所有步骤相辅相成，使 QAMR 的表现优于每一种消融版本。

**发现2**。 查询改写消融版本的表现优于其他缺少跨多版本文档解读查询机制的消融版本。在五种消融版本中（表 V），查询改写消融版本的答案正确性和上下文召回率最高，上下文精确率则位居第二。这凸显了查询改写对于在多版本文档场景中实现准确问答的重要性。

**RQ3（响应时间）**。 表VII比较了 QAMR 和基线的平均响应时间。在 Ciena 数据集上的平均响应时间均高于 REQuestA，这是因为 Ciena 的文档规模更大、复杂度更高，且通常包含多个表格和示意图。从表VII可以观察到两点：（1）对于两个数据集，基线的平均响应时间比 QAMR 高约 8%（Ciena：45/41，REQuestA：40/37）。（2）在 Ciena 数据集上，QAMR 和基线的平均响应时间均比在 REQuestA 上高 12%（QAMR：41/37，基线：45/40），这表明随着文档复杂度增加，QAMR 的运行时间与基线成比例增长。

如表VII所示，对于QAMR，上下文缩减是最耗时的步骤，占总

表VII：QAMR 与基线的平均响应时间。

| | 平均响应时间 每个查询（以秒计） | | 响应时间分解 按步骤（%） | | | | |
|---|---|---|---|---|---|---|---|
| | Ciena | REQuestA | 查询改写 | 上下文缩减 | 上下文选择 | 答案生成 | 其他步骤 |
| **QAMR** | 41 | 37 | 27% | 36% | 23% | 13% | 1% |
| **基线** | 45 | 40 | 20% | N/A | 33% | 46% | 1% |

响应时间的36%，其次是查询改写和上下文选择，分别占27%和23%。对于不包含上下文缩减步骤的基线，答案生成是最耗时的步骤，其次是上下文选择和查询改写。在基线中，答案生成和上下文选择的开销高于查询改写，而QAMR则相反。这是因为在没有上下文缩减的情况下，基线必须处理较大的文本块；而上下文缩减能够为答案生成和选择提供简洁的上下文，从而提高效率。

**发现**。平均而言，QAMR 在两个数据集上都比基线快约 8%，这主要是因为上下文缩减步骤为答案生成和上下文选择步骤提供了更精简的上下文。

**结论**。 根据在 Ciena 基础设施上部署 QAMR 的内部情况，专家认为 QAMR 的响应时间切实可行。

### *G.* 有效性考量

**内部有效性**。 为确保比较公平，QAMR 和基线使用了相同的底层大语言模型（LLM-Core）、共享组件的相同提示词，以及相同的检索参数（表 II）。基线的单块处理方法按照先前研究中的最佳实践进行了配置 [13], [6]。QAMR 和基线的指标均使用相同的评估用大语言模型（LLM-Eval）计算。为降低随机性，我们将温度设为 0.01、top-k 设为 1，并将每项实验重复十次 [42], [43]。关于数据泄漏，我们确信 Ciena 专有数据集并未用于 LLM-Core的预训练。尽管 REQuestA 数据集是开源的，但 QAMR 和基线依赖同一个 LLM-Core，因此任何影响 LLM-Core 的数据泄漏都会同等影响两者。因此，我们的比较仍然公平且可靠。

**外部有效性**。我们的评估使用了来自不同来源的两个数据集。Ciena 数据集包含大量复杂的多版本系统文档，以及由专家编写和验证的问题与真实答案。REQuestA 数据集包含自动生成和人工编写的问题，所有自动生成的题目都经过严格审核，以确保其相关性和质量。在大语言模型的选择方面，我们对 LLM-Core 和 LLM-Eval均使用了 Llama 3-70B [31] 。开展实验时，行业合作伙伴的政策禁止在处理专有数据时使用托管于外部的大语言模型。此外，由于计算每项指标需要大量大语言模型调用，若使用托管于外部的大语言模型进行大语言模型评审评估，成本会非常高昂，达数千美元。尽管对其他大语言模型进行基准测试很有价值，但仅使用

Llama 3-70B 并不会带来显著风险，因为它仍是准确度最高的指令遵循模型之一，可免费用于研究和行业用途，也适合本地部署；其文本生成和问答性能可与更新的大语言模型相媲美 [44]。

## V. 相关工作

在大语言模型出现之前，基于 BERT 的抽取式模型被用于自动化问答，可通过集成检索增强生成 [13]或不集成检索增强生成 [45]来实现。较新的方法依赖 ChatGPT [42] 和 Llama [31], 等生成式大语言模型，它们不仅能识别相关文本，还能生成贴近人类表达、且针对用户查询和对话历史量身定制的回答。例如，Chaudhary 等人 [6]开发了一款基于大语言模型的聊天机器人，帮助系统工程师处理与持续集成/持续交付相关的查询。同样，Barnett 等人 [11] 介绍了将基于检索增强生成的大语言模型聊天机器人应用于不同领域案例研究的经验，并提出了在软件工程领域改进此类聊天机器人的建议。这些现有的检索增强生成架构采用传统的单分块策略，与我们在第 IV 节中的评估基线类似。

正如近期文献 [11], [6], [13], [28], [46],[12], [47], 所指出的，检索增强生成方案面临一些重要局限，包括难以确定最佳分块大小、检索到文本块中的无关或噪声数据，以及易受低质量用户查询影响。我们的方法旨在缓解这些众所周知的局限。具体而言，我们的双重分块策略将用于检索和生成的文本块解耦，从而无需为这两项任务选择统一的分块大小。此外，我们的方法还采用上下文缩减和选择机制，以减少文本块中的噪声数据。最后，我们方法中的查询改写组件会生成多个独立查询，从不同角度改进并扩展用户查询；与将用户查询改写为单个独立查询的技术相比，该组件有助于提高系统应对低质量用户查询的能力。

关于检索增强生成中的检索步骤，Liu 等人 [48]提出了一种分层方法：给定一组文档，先进行文档检索，再从检索到的文档中检索相关段落。当文档之间有足够的差异时，这种方法效果良好，因为文档检索阶段能够有效缩小搜索范围。然而，这种分层方法无法有效区分高度重叠的多版本发布文档；与上文讨论的其他方法类似，它也未能解决第 I 节所述的多版本发布文档问答挑战。

## VI. 经验教训

第 ***1*** 课：在第 IV-B 节介绍的六项指标中，答案正确性、上下文精确率和上下文召回率会根据真实答案评估生成的答案，而上下文相关性、答案相关性和答案忠实度则不会。相反，后面这些指标评估响应与用户查询或检索块的重合程度，而不是其是否与真实值匹配。即使答案本身不正确，只要与查询或检索块相匹配，也能在这些指标上取得高分。在我们的实验中，这种效应使基线在这三项指标上看起来与 QAMR 相当，尽管在另外三项依赖真实值的指标上，基线的表现明显更差。

经验教训 ***1*** 的启示。应将上下文相关性、答案相关性和忠实度视为合理性的指标，而非正确性的指标。评估准确性时，应结合基于真实值的指标进行解读，并优先考虑后者。

经验教训 ***2***：在特定领域和技术文档中，即使措辞只有细微变化，也可能影响聊天机器人准确性。例如，提问“性能监控有哪些注意事项？”会得到准确答案，而将“性能监控”替换为“*PM*”或“运营”——这些术语在文档中并未使用——则会导致结果不佳。这是因为检索在很大程度上是字面匹配：如果查询中的词汇没有出现在语料库中，就无法检索到相关文本块。

经验教训 ***2*** 的启示：基于特定领域术语表进行查询改写，通过消除歧义、扩展首字母缩略词，并将术语映射到术语表中的同义词，使用户查询与文档语言保持一致。

经验教训 ***3***：基线、QAMR 及其消融实验中最主要的错误来源是文档检索不正确，因为大多数错误都源于检索到的上下文不相关或不完整。其次，主要错误来源是幻觉；与 QAMR 相比，基线中出现幻觉的频率更高。这是因为查询改写能够在多版本文档中实现特定版本检索，从而提高检索准确率；而双重分块和上下文缩减则通过提高检索准确率并剔除无关信息，进一步减少幻觉。

经验教训 ***3*** 的启示。在多版本系统中采用查询改写实现特定版本检索，使用双重分块提高整体检索准确率，并通过上下文缩减从检索到的上下文中剔除无关信息。

## 七、结论

我们介绍了 QAMR，这是一款基于检索增强生成的聊天机器人，用于多版本系统问答。我们使用两个数据集对 QAMR 进行了评估，将其有效性和效率与最先进的基线进行比较，并开展消融研究，以评估各组件的影响，尤其是特定版本查询改写模块的影响。QAMR 在答案正确性和检索准确率方面均显著优于基线及其消融版本。此外，用于评估答案准确率的自动化「大语言模型担任裁判」指标与专家评估高度一致，验证了这些自动化指标作为人工判断替代指标的可靠性。

## VIII. 致谢

我们衷心感谢 Mitacs Accelerate、Ciena、安大略省研究生奖学金（OGS）计划，以及加拿大自然科学与工程研究委员会通过 Discovery 和 Discovery Accelerator 计划提供的资助。

# 参考文献


[1] M. M. Lehman，“重新审视软件演化定律”，载于《软件过程技术》，第五届欧洲研讨会 EWSPT ’96，法国南锡，1996年10月9日至11日，论文集，丛书《计算机科学讲义》，C. Montangero 编，第1149卷。Springer，1996年，第 108 124 [ ] 10 1007 0017737 –页。在线。获取地址：https://doi.org/。/BFb K. Pohl、G. Bockle 和 F. van der Linden，《软件产品线工程——基础、原则与技术》。Springer，2005 [ ] 10 1007 3 540 28901 1 。在线。获取地址：https://doi.org/。/ - - - A. Abdellatif、K. Badran、D. E. Costa 和 E. Shihab，“软件工程聊天机器人的自然语言理解平台比较”，《IEEE软件工程汇刊》，第 48， 8， 3087 3102, 卷，第页。在线。2022 [ ] 10 1109 2021 3078384 获取地址：https://doi.org/。/TSE。.

[4] A. Abdellatif、D. Costa、K. Badran、R. Abdalkareem 和 E. Shihab，“聊天机器人开发中的挑战：对 Stack Overflow 帖子的研究”，载于 MSR ’20：第17届国际软件仓库挖掘会议，韩国首尔，2020年6月29日至30日，S. Kim、G. Gousios、S. Nadi 和 J. Hejderup 编。ACM，2020年，第174–185页。 [ ] 10 1145 3379597 3387472 在线。获取地址：https://doi.org/。/。

[5] G. Daniel 和 J. Cabot，“将模型驱动工程应用于聊天机器人领域：xatkit 实践”，《科学计算与编程》，第232卷，第103032页，2024年。 [在线]。获取地址：https: //doi.org/ 10 1016 2023 103032 。/j.scico.。

[6] D. Chaudhary、S. L. Vadlamani、D. Thomas、S. Nejati 和 M. Sabetzadeh，“开发基于 Llama 的持续集成/持续交付问答聊天机器人：爱立信案例研究”，载于 IEEE国际软件维护与演化会议，ICSME 2024，美国亚利桑那州弗拉格斯塔夫，2024年10月6日至11日。IEEE，2024年，第707–718页。 [在线]。获取地址：https://doi.org/ 10 1109 58944 2024 00075 。/ICSME。。

[7] J. Huang、Y. Zhong、G. Y. adn Zhihan Jiang、M. Yan、W. Luan、T. Y. adn Rui Ren 和 M. Lyu，“iKnow：一种用于云运维的意图引导型聊天机器人，采用检索增强生成”，载于 IEEE/ACM 自动化软件工程（ASE 2025）会议，韩国首尔，2025年11月16日至20日，即将发表。[8] P. S. H. Lewis、E. Perez、A. Piktus、F. Petroni、V. Karpukhin、N. Goyal、H. K¨uttler、M. Lewis、W. Yih、T. Rockt¨aschel、S. Riedel 和 D. Kiela，“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：神经信息处理系统年度会议 2020，NeurIPS 2020，2020年12月6日至12日，线上，H. Larochelle、M. Ranzato、R. Hadsell、M. Balcan 和 H. Lin 编，2020年。 [在线]。获取地址：https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html[9] K. Guu、K. Lee、Z. Tung、P. Pasupat 和 M. Chang，“检索增强语言模型预训练”，载于第37届国际机器学习会议论文集，ICML 2020，2020年7月13日至18日，线上活动，丛书《机器学习研究论文集》，第119卷。PMLR，2020年，第3929–3938页。 [在线]。获取地址：http://proceedings.mlr.press/v119/guu20a.html [10] G. Izacard 和 E. Grave，“利用段落检索与生成模型实现开放域问答”，载于计算语言学协会欧洲分会第16届会议论文集：主会场，EACL 2021，线上，2021年4月19日至23日，P. Merlo、J. Tiedemann 和 R. Tsarfaty 编。计算语言学协会，2021年，第874–880页。 [在线]。获取地址：https://doi.org/ 10 18653 1 2021 74 。/v / .eacl-main。

[11] S. Barnett、S. Kurniawan、S. Thudumu、Z. Brannelly 和 M. Abdelrazek，“构建检索增强生成系统时的七个失效点”，载于 IEEE/ACM 第三届人工智能工程国际会议——人工智能软件工程，CAIN 2024，葡萄牙里斯本，2024年4月14日至15日，J. Cleland-Huang、J. Bosch、H. Muccini 和 G. A. Lewis 编。ACM，2024年，第194–199页。 [在线]。获取地址：https://doi.org/ 10 1145 3644815 3644945 。/。

[12] Y. Yu、W. Ping、Z. Liu、B. Wang、J. You、C. Zhang、M. Shoeybi 和 B. Catanzaro，“Rankrag：在大语言模型中统一上下文排序与检索增强生成”，载于《神经信息处理系统进展》第38卷：神经信息处理系统年度会议 2024，NeurIPS 2024，加拿大不列颠哥伦比亚省温哥华，2024年12月10日至15日，A. Globersons、L. Mackey、D. Belgrave、A. Fan、U. Paquet、J. M. Tomczak 和 C. Zhang 编，2024年。[在线]。获取地址：http://papers.nips.cc/paper files/paper/2024/hash/db93ccb6cf392f352570dd5af0a223d3-Abstract-Conference.html [13] S. Ezzini、S. Abualhaija、C. Arora 和 M. Sabetzadeh，“基于人工智能的问答辅助工具，用于分析自然语言需求”，载于《第45届IEEE/ACM国际软件工程会议》，ICSE 2023，澳大利亚墨尔本，2023年5月14日至20日。IEEE，2023年，第1277–1289页。 [在线]。可访问：https: //doi.org/ 10 1109 48619 2023 00113 。/ICSE . .

[14] Confident AI，“Deepeval”，https://github.com/confident-ai/deepeval，2024, [在线；访问日期为1月 6, 2026]。[15] Y. Liu、D. Iter、Y. Xu、S. Wang、R. Xu和C. Zhu，“G-eval：使用GPT-4并实现更佳人类对齐的自然语言生成评估”，载于《2023年自然语言处理经验方法会议论文集》，EMNLP 2023，新加坡，2023年12月10日，H. Bouamor、J. Pino和K. Bali编。计算语言学协会，2023年，第2511–2522页。 [在线]。可访问：https://doi.org/ 10 18653 1 2023 153 。/v / .emnlp-main。

[16] K. Papineni、S. Roukos、T. Ward和W. Zhu，“BLEU：一种机器翻译自动评估方法”，载于《计算语言学协会第40届年会论文集》，2002年7月6日至12日，美国宾夕法尼亚州费城。ACL，2002年，第311–318页。[在线]。可访问：https://aclanthology.org/P02-1040/ [17] C.-Y. Lin，“ROUGE：用于摘要自动评估的软件包”，载于《文本摘要拓展》。西班牙巴塞罗那：计算语言学协会，2004年7月，第74–81页。 [在线]。可访问：https://aclanthology.org/W 04 1013 - T. Goyal、J. J. Li和G. Durrett，“GPT-3

[18] 时代的新闻摘要与评估”，CoRR，第abs/2209.12356卷，2022年。 [在线]。可访问：https://doi.org/ 10 48550 2209 12356 。/ arXiv. . C. Liu、R. Lowe、I. Serban、M. Noseworthy、L. Charlin和J.

[19] Pineau，“如何不评估对话系统：对用于对话响应生成的无监督评估指标的实证研究”，载于《2016年自然语言处理经验方法会议论文集》，EMNLP 2016，美国得克萨斯州奥斯汀，2016年11月1日至4日，J. Su、X. Carreras和K. Duh编。计算语言学协会，2016年，第2122–2132页。 [在线]。可访问：https://doi.org/ 10 18653 1 16 1230 。/v /d - N. Wu、M.

[20] Gong、L. Shou、S. Liang和D. Jiang，“大语言模型是用于摘要评估的多样化角色扮演者”，载于《自然语言处理与中文计算——第12届中国计算机学会全国会议》，NLPCC 2023，中国佛山，2023年10月12日至15日，论文集，第I部分，计算机科学讲义系列，F. Liu、N. Duan、Q. Xu和Y. Hong编，第14302卷。Springer，2023年，第695–707页。 [在线]。可访问：https: //doi.org/ 10 1007 978 3 031 44693 1 54 。/ - - - - ReplicationPackage，“

[21] QAMR复现包”，https://github.com/ parham-box/SANER-QAMR， 2025, [在线；访问日期为1月 6, 2026]。[22] P. Laban、A. R. Fabbri、C. Xiong和C. Wu，“大海捞针式摘要：对长上下文大语言模型和检索增强生成系统的挑战”，载于《2024年自然语言处理经验方法会议论文集》，EMNLP 2024，美国佛罗里达州迈阿密，2024年11月12日至16日，Y. Al-Onaizan、M. Bansal和Y. Chen编。计算语言学协会，2024年，第9885–9903页。 [在线]。可访问：https://aclanthology.org/ 2024 552 .

[23] emnlp-main。S. Hassani、M. Sabetzadeh、D. Amyot和J. Liao，“重新思考法律合规自动化：大语言模型带来的机遇”，载于《第32届IEEE国际需求工程会议》，RE 2024，冰岛雷克雅未克，2024年6月24日至28日，G. Liebel、I. Hadar和P. Spoletini编。IEEE，2024年，第432–440页。 [在线]。可访问：https://doi.org/ 10 1109 59067 2024 00051 。/RE . . N. F.

[24] Liu、K. Lin、J. Hewitt、A. Paranjape、M. Bevilacqua、F. Petroni和P. Liang，“迷失在中间：语言模型如何使用长上下文”，《计算语言学协会会刊》，第 12， 157 173, 卷，第–页。2024 [ ] 10 1162 在线。可访问：https://doi.org/。/tacl a 00638

[25] T. Mikolov、I. Sutskever、K. Chen、G. S. Corrado和J. Dean，“词语和短语的分布式表示及其组合性”，载于《神经信息处理系统进展26：第27届神经信息处理系统年会论文集》。会议于2013年12月5日至8日在美国内华达州太浩湖举行，C. J. C. Burges、L. Bottou、Z. Ghahramani和K. Q. Weinberger编，2013年，第3111–3119页。 [在线]。可访问：https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html[26] P. Khamsepour、M. Cole、I. Ashraf、S. Puri、M. Sabetzadeh和S. Nejati，“QAMR工作示例”，https://github.com/parham-box/SANER-QAMR/blob/main/Example.pdf， 2025, [在线；访问日期为1月 6, 2026]。

[27] ——，“QAMR完整提示词”，https://github.com/parham-box/SANER-QAMR/blob/main/Prompts.pdf，2025， [在线；访问日期为1月 6, 2026]。[28] X. Ma、Y. Gong、P. He、H. Zhao和N. Duan，“面向检索增强型大语言模型的查询改写”，CoRR，第abs/2305.14283卷，2023年。 [在线]。可访问：https://doi.org/10.48550/ arXiv.2305.14283[29] J. G. Carbonell和J. Goldstein，“MMR的应用：基于多样性的重排序，用于重新排列文档并生成摘要”，SIGIR Forum，第51卷第2期，第209–210页，2017年。 [在线]。可访问：https://doi.org/ 10 1145 3130348 3130369 。/ 。Langchain, [30] https://www.langchain.com， 2024, [在线；访问日期为1月6, 2026]。[31] AI@Meta，“Llama 3模型卡”，https://github.com/meta-llama/llama3/blob/main/MODEL CARD.md，2024年。[32] T. Wolf、L. Debut、V. Sanh、J. Chaumond、C. Delangue、A. Moi、P. Cistac、T. Rault、R. Louf、M. Funtowicz和J. Brew，“HuggingFace的transformers：最先进的自然语言处理”，CoRR，第abs/1910.03771卷，2019年。 [在线]。可访问：http://arxiv.org/abs/1910. 03771[33] HuggingFace，“Bitsandbytes 库”，https://huggingface.co/docs/ bitsandbytes/main/en/index， 2024, [在线；访问日期为1月 6, 2026]。[34] S. Xiao、Z. Liu、P. Zhang和N. Muennighoff，“C-Pack：推进通用中文嵌入的资源包”，CoRR，第abs/2309.07597卷，2023年。 [在线]。可访问：https://doi.org/10.48550/ arXiv.2309.07597[35] N. Muennighoff、N. Tazi、L. Magne和N. Reimers，“MTEB：大规模文本嵌入基准”，载于《第17届计算语言学协会欧洲分会会议论文集》，EACL 2023，克罗地亚杜布罗夫尼克，2023年5月2日至6日，A. Vlachos和I. Augenstein编。计算语言学协会，2023年，第2006–2029页。 [在线]。可访问：https://doi.org/10.18653/v1/2023。eacl-main.148[36]Langchain，“多向量检索器”，https://python.langchain.com/v0.1/ docs/modules/data connection/retrievers/multi vector/， 2024，[在线；访问日期为1月 6, 2026]。[37] C. vectorstore，https://www.trychroma.com， 2024，[在线；访问日期为1月 6, 2026]。[38] Langchain，“LangChain 表达式语言”，https://python.langchain. com/docs/concepts/# langchain-expression-language-lcel， 2024, [在线；访问日期为1月 6, 2026]。[39] F. Wilcoxon，“基于排序方法的个体比较”，载于《统计学突破：方法论与分布》。Springer，1992年，第196–202页。[在线]。可访问：https: //doi.org/10.1007/978-1-4612-4380-9 16

[40] A. Vargha 和 H. D. Delaney，《对 McGraw 和 Wong 的共同语言效应量统计 cl 的批评与改进》，《教育与行为统计学杂志》，第 25， 2， 101 132, 卷，第 期，第–页。线 2000 [ ] 10 3102 上。获取地址：https://doi.org/ 。 / 10769986025002101 [41] P. Khamsepour、M. Cole、I. Ashraf、S. Puri、M. Sabetzadeh 和 S. Ne- jati，《QAMR消融研究结果》，https://github.com/parham-box/ SANER-QAMR/blob/main/RQ 2 2025, / [ 6, 2026] Ablation-Study-Results.pdf，线上；访问于 1 月。[42] OpenAI，《GPT-4技术报告》，CoRR，第 abs/2303.08774 卷，2023 年。 [ ] 10 48550 2303 08774 线上。获取地址：[43] https://doi.org/ 。 /arXiv. 。D. Guo、Q. Zhu、D. Yang、Z. Xie、K. Dong、W. Zhang、G. Chen、X. Bi、Y. Wu、Y. K. Li、F. Luo、Y. Xiong 和 W. Liang，《Deepseek-coder：当大语言模型遇上编程——代码智能的崛起》，CoRR，第 abs/2401.14196 卷，2024 年。 [线上]。获取地址：https://doi.org/ 10 48550 2401 14196 。 / arXiv. 。W. Chiang、L. Zheng、Y. Sheng、A. N. Angelopoulos、T. Li、D. Li、B. Zhu、H. Zhang、M. I. Jordan、J. E. Gonzalez 和 I. Stoica，《Chatbot Arena：一个通过人类偏好评估大语言模型的开放平台》，载于第四十一届国际机器学习会议（ICML 2024），奥地利维也纳，2024年7月21日至27日。OpenReview.net，2024 年。 [线上]。获取地址：https://openreview.net/forum?id=3MW8GKNyzI [45] M. Borg、J. Bengtsson、H. ¨Osterling、A. Hagelborn、I. Gagner 和 P. Tomaszewski，《用于瑞典语练习的演进型对话代理中生成式对话模型的质量保证》，载于第一届人工智能工程国际会议论文集：人工智能软件工程（CAIN 2022），宾夕法尼亚州匹兹堡，2022年5月16日至17日，I. Crnkovic 编。ACM，2022 年，第 22–32 页。 [线上]。获取地址：https://doi.org/ 10 1145 3522664 3528592 。 / 。A. [44] Landschaft、D. Antweiler、S. Mackay、S. Kugler、S. R¨uping、S. Wrobel、T. H¨ores 和 H. Allende-Cid，《在基于 PRISMA 的医学系统性文献综述中实施并评估额外的 GPT-4 审稿人》，《国际医学信息学杂志》，第 189 卷，文章编号 105531，2024 年。 [线上]。获取地址：https://doi.org/10.1016/j. ijmedinf. 2024 105531 。Y. [46] Zhu、H. Yuan、S. Wang、J. Liu、W. Liu、C. Deng、Z. Dou 和 J. Wen，《用于信息检索的大语言模型：综述》，CoRR，第 abs/2308.07107 卷，2023 年。 [线上]。获取地址：https://doi.org/ 10 48550 2308 07107 。 /arXiv. 。Y. Liu、K. [47] Hashimoto、Y. Zhou、S. Yavuz、C. Xiong 和 P. S. Yu，《用于开放域问答的稠密层次检索》，载于计算语言学协会论文集：EMNLP 2021，线上活动／多米尼加共和国蓬塔卡纳，2021年11月16日至20日，M. Moens、X. Huang、L. Specia 和 S. W. Yih 编。计算语言学协会，2021 年，第 188–200 页。[线上]。获取地址：https://doi.org/10.18653/v1/2021.findings-emnlp.19 [48]