高级 RAG：架构、技术、应用和发展前景

> 来源：首席AI分享圈 https://www.aisharenet.com/

检索增强生成（Retrieval-augmented generation，RAG）已经成为 AI 领域的重要框架，极大提升了大语言模型（LLMs）在使用外部知识源生成响应时的准确性和相关性。据 Databricks 的数据显示，企业中 60% 的 LLM 应用都采用了检索增强生成（RAG），其中 30% 使用了多步骤流程。RAG 之所以受到广泛关注，是因为其生成的响应比仅依赖微调模型的响应几乎 提升了 43% 的准确性，表明了 RAG 在提高 AI 生成内容质量和可靠性方面的巨大潜力。

然而，传统 RAG 方法在应对复杂查询、理解细微语境以及处理多种数据类型时仍面临一些挑战。这些局限性推动了高级 RAG 的诞生，旨在提升 AI 在信息检索和生成上的能力。值得注意的是，一些公司 已将 RAG 集成到大约 60% 的产品中，显示了其在实际应用中的重要性和有效性。

该领域的一大突破是多模态 RAG 和知识图谱的引入。多模态 RAG 扩展了 RAG 的能力，不仅可以处理文本，还能处理包括图像、音频和视频在内的多种数据。这使得 AI 系统在与用户交互时更加全面并具备更强的语境理解能力。而知识图谱则通过结构化的知识表示，提升了信息检索过程以及生成内容的连贯性和准确性。微软的研究 表明，GraphRAG 所需的 Token 数量比其他方法减少了 26% 到 97%，显示出更高的效率和计算成本的降低。

这些 RAG 技术的进步在多个基准测试和实际应用中都带来了显著的性能提升。例如，知识图谱 在 RobustQA 测试中取得了 86.31% 的准确率，大大超过了其他 RAG 方法。此外，Sequeda 和 Allemang 的后续研究发现，结合本体可以降低 20% 的错误率。企业也从这些进展中获益良多，LinkedIn 报告称，通过 RAG 加知识图谱的方法，将客户支持的解决时间减少了 28.6%。

本文将深入探讨高级 RAG 的演进，探索多模态 RAG 和知识图谱 RAG 的复杂性及其对 AI 驱动的信息检索和生成的提升效果。我们还将讨论这些创新在不同行业中的应用潜力以及在推广和应用这些技术时面临的挑战。

- [什么是检索增强生成（RAG），以及它对大语言模型（LLM）为何重要？]
- [RAG 架构的类型]
- [从基础 RAG 到高级 RAG：如何克服局限和提升能力]
- [企业中的高级 RAG 系统组成和流程]

- [高级 RAG 技术]
- [高级 RAG 的应用和案例分析]
- [如何使用高级 RAG 构建对话工具？]
- [如何构建高级 RAG 应用？]
- [知识图谱在高级 RAG 中的崛起]
- [高级 RAG：通过多模态检索增强生成扩展视野]
- [LeewayHertz 的 GenAI 协作平台 ZBrain 如何在高级 RAG 系统中脱颖而出？]

什么是检索增强生成（RAG），以及它对大语言模型（LLM）为何重要?

大语言模型（LLM）已成为 AI 应用的核心，从虚拟助手到复杂的数据分析工具均依赖其强大功能。但尽管这些模型能力出众，它们在提供最新和准确信息方面仍存在局限。这时，检索增强生成（RAG）为 LLM 提供了强有力的补充。

什么是检索增强生成（RAG）？

检索增强生成（RAG）是一种高级技术，它通过整合外部知识源来提升大语言模型（LLM）的生成能力。LLM 基于大量数据集进行训练，拥有数十亿参数，能够执行如回答问题、语言翻译和文本补全等多种任务。RAG 则更进一步，通过引用权威且领域专精的知识库，提高生成内容的相关性、准确性和实用性，而无需重新训练模型。这种成本效益高且高效的方法，已成为企业优化其 AI 系统的理想选择。

RAG（检索增强生成）如何帮助大语言模型（LLM）解决核心问题?

大语言模型（LLM）在驱动智能聊天机器人和其他自然语言处理（NLP）应用中扮演着关键角色。通过广泛的训练，它们试图在各种情境中提供准确的答案。然而，LLM 本身存在一些不足，面临多重挑战：

1. **错误信息**：当 LLM 知识不足时，可能会生成不准确的答案。
2. **信息过时**：训练数据是静态的，因此模型生成的回答可能已经过时。
3. **非权威信息源**：生成的回答有时可能来自不可靠的来源，影响可信度。
4. **术语混淆**：不同数据来源对同一术语的使用不一致，容易引发误解。

RAG 通过为 LLM 提供外部权威数据来源，提升模型的回答准确性与实时性，进而解决上述问题。以下几点解释了为什么 RAG 对 LLM 的发展如此重要：

1. **提升准确性与相关性**：由于训练数据静态化，LLM 有时会给出不准确或不相关的答案。RAG 从权威来源中提取最新、最相关的信息，确保模型回答更为准确且符合当前情境。
2. **突破静态数据的限制**：LLM 的训练数据有时已过时，无法反映最新的研究或新闻。RAG 使 LLM 能够访问最新的数据，保持信息的时效性和相关性。
3. **提升用户信任**：LLM 可能会生成所谓的“幻觉”——即自信但错误的回答。RAG 通过让 LLM 引用来源，提供可验证的信息，增强透明度与用户信任感。
4. **节约成本**：用新数据重新训练 LLM 代价高昂。RAG 无需重新训练整个模型，只需利用外部数据源，提供了一种更为经济高效的替代方案，使得高级 AI 技术更为广泛应用。
5. **增强开发者控制权与灵活性**：RAG 为开发者提供了更多的自由度，能够灵活指定知识来源，快速适应需求变化，并确保敏感信息的适当处理，从而支持广泛应用，提高 AI 系统的效果。
6. **提供定制化答案**：传统 LLM 往往给出过于笼统的答案。RAG 将 LLM 与组织的内部数据库、产品信息和用户手册相结合，提供更加具体和相关的回答，大幅提升客户支持和互动体验。

RAG（检索增强生成）通过整合外部知识库，使得 LLM 能够生成更准确、实时且符合上下文的回答。这对依赖 AI 的组织而言极为重要，从客户服务到数据分析，RAG 不仅提升了效率，也增强了用户对 AI 系统的信任。

RAG 架构的类型

检索增强生成（RAG）代表了 AI 技术的一次重大进步，它将语言模型与外部知识检索系统结合在一起。通过从庞大的外部数据源中获取详细且相关的信息，这种混合方法提升了 AI 响应生成的能力。了解不同类型的 RAG 架构有助于我们根据具体需求更好地发挥其优势。以下是对三种主要 RAG 架构的深入探讨：

1. Naive RAG

Naive RAG 是最基础的检索增强生成方法。它的原理很简单，系统会根据用户的查询从知识库中提取相关的信息块，然后使用这些信息块作为上下文，通过语言模型生成答案。

特点：

- **检索机制**：采用简单的检索方法，通常通过关键词匹配或基本语义相似度，从预先建立的索引中提取相关文档块。
- **上下文融合**：检索到的文档与用户的查询进行拼接，并输入语言模型生成答案。这种融合为模型提供了更丰富的上下文，从而生成更加相关的答案。

- **处理流程**：系统遵循固定流程：检索、拼接、生成。模型不会对提取的信息进行修改，而是直接使用这些信息生成答案。

2. Advanced RAG

Advanced RAG 基于 Naive RAG 的基础，采用更先进的技术提升检索的准确性和上下文的相关性。它通过结合先进的机制，克服了 Naive RAG 的一些局限性，从而更好地处理和利用上下文信息。

特点：

- **增强检索**：利用高级检索策略，如查询扩展（在初始查询中添加相关术语）和迭代检索（多阶段优化文档），提升检索信息的质量和相关性。
- **上下文优化**：通过注意力机制等技术，有选择性地聚焦于最相关的上下文部分，帮助语言模型生成更准确且上下文更加精准的回答。
- **优化策略**：采用优化策略，如相关性评分和上下文增强，确保模型能获取最相关且优质的信息生成回答。

3. Modular RAG

Modular RAG 是最灵活、最具定制化的 RAG 架构。它将检索和生成过程分解成独立的模块，允许根据具体应用的需求进行优化和更换。

特点：

- **模块化设计**：将 RAG 的流程分解为不同模块，如查询扩展、检索、重排序和生成。每个模块可以独立优化，按需替换。
- **灵活定制**：允许高度定制化，开发人员可以在每个步骤中尝试不同配置和技术，以找到最佳方案。该方法为各种应用场景提供了定制化的解决方案。
- **集成与适应**：该架构能够整合额外功能，如记忆模块（用于记录过去的互动）或搜索模块（从搜索引擎或知识图谱中提取数据）。这种适应性使得 RAG 系统能够灵活调整，满足特定的需求。

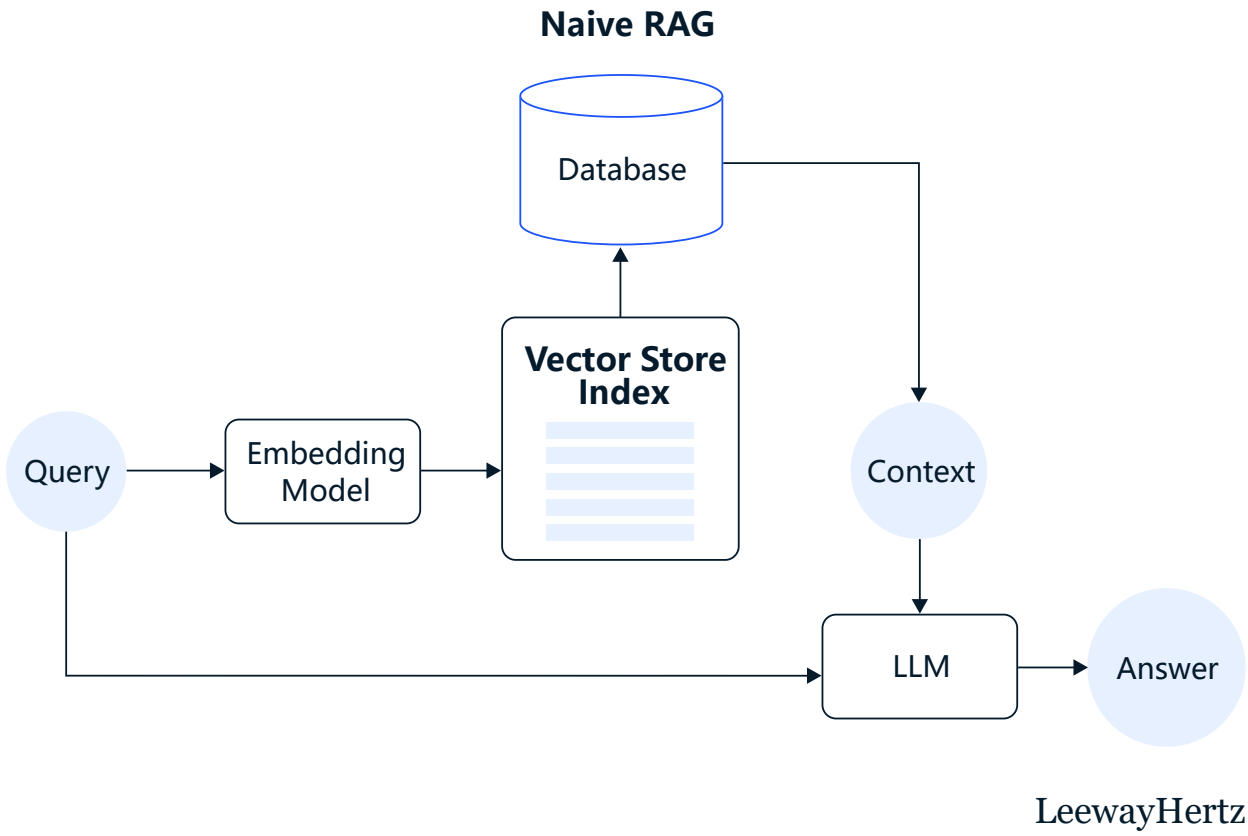
理解这些类型和特点，对于选择和实施最合适的 RAG 架构至关重要。

从基础到高级 RAG：突破限制，提升能力

检索增强生成（Retrieval-augmented generation，RAG）在 自然语言处理（NLP） 中已成为一种非常有效的方法，它结合了信息检索和文本生成，能够生成更准确且更符合上下文的输出。但随着技术的发展，初期的“基础”RAG 系统也暴露出了一些缺陷，这也

推动了更高级版本的出现。基础 RAG 向高级 RAG 的进化，意味着我们正逐步克服这些缺陷，极大地提升了 RAG 系统的整体能力。

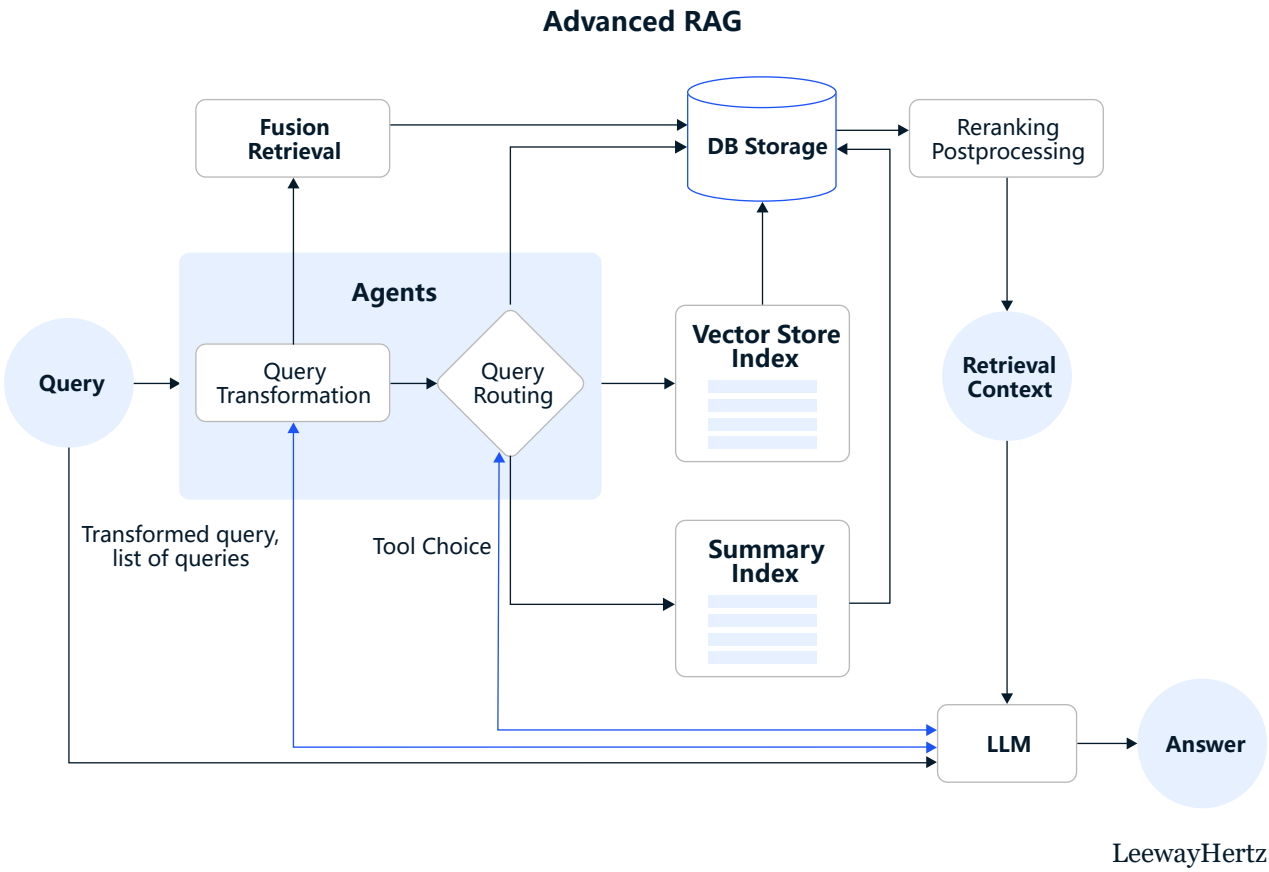
基础 RAG 的局限性



基础的 RAG 框架是将检索和生成结合起来进行 NLP 的初步尝试。虽然这种方法很有创新性，但仍然面临一些局限：

1. **简单的检索方法**：基础 RAG 系统大多依赖于简单的关键词匹配，这种方法难以理解查询中的细微差别和上下文，从而检索出不够相关或部分相关的信息。
2. **难以理解上下文**：这些系统难以正确理解用户查询的背景。例如，基础 RAG 系统可能会检索出包含查询关键词的文档，但未能抓住用户的真正意图或上下文，从而无法准确满足用户需求。
3. **处理复杂查询的能力有限**：面对复杂或多步骤的查询时，基础 RAG 系统表现不佳。它们在理解上下文和精准检索方面的局限使其难以有效处理复杂问题。
4. **静态的知识库**：基础 RAG 系统依赖于静态的知识库，缺乏动态更新的机制，信息可能会随着时间推移而过时，影响响应的准确性和相关性。
5. **缺乏迭代优化**：基础 RAG 缺乏基于反馈进行优化的机制，无法通过迭代学习改进性能，随着时间的推移表现会停滞。

向高级 RAG 的过渡



随着技术的发展，针对基础 RAG 系统的缺陷，我们已经有了更复杂的解决方案。高级 RAG 系统通过以下几种方式克服了这些挑战：

1. **更复杂的检索算法**：高级 RAG 系统使用语义搜索和上下文理解等复杂技术，语义搜索能够超越关键词匹配，理解查询背后的真正含义，从而提高检索结果的相关性。
2. **增强的上下文整合**：这些系统会结合上下文和相关性权重来整合检索结果，确保不仅信息准确，而且在上下文中也是合适的，能更好地回应用户的查询和意图。
3. **迭代优化与反馈机制**：
   高级 RAG 系统采用了迭代优化过程，能够通过结合用户反馈持续提高准确性和相关性，随着时间的推移不断优化。
4. **动态知识更新**：
   高级 RAG 系统能够动态更新知识库，持续引入最新的信息，确保系统始终反映最新的趋势和进展。
5. **复杂的上下文理解**：
   借助更先进的 NLP 技术，高级 RAG 系统对查询和上下文有更深的理解，能够分析语义上的细微差别、上下文提示以及用户意图，生成更加连贯且相关的回应。

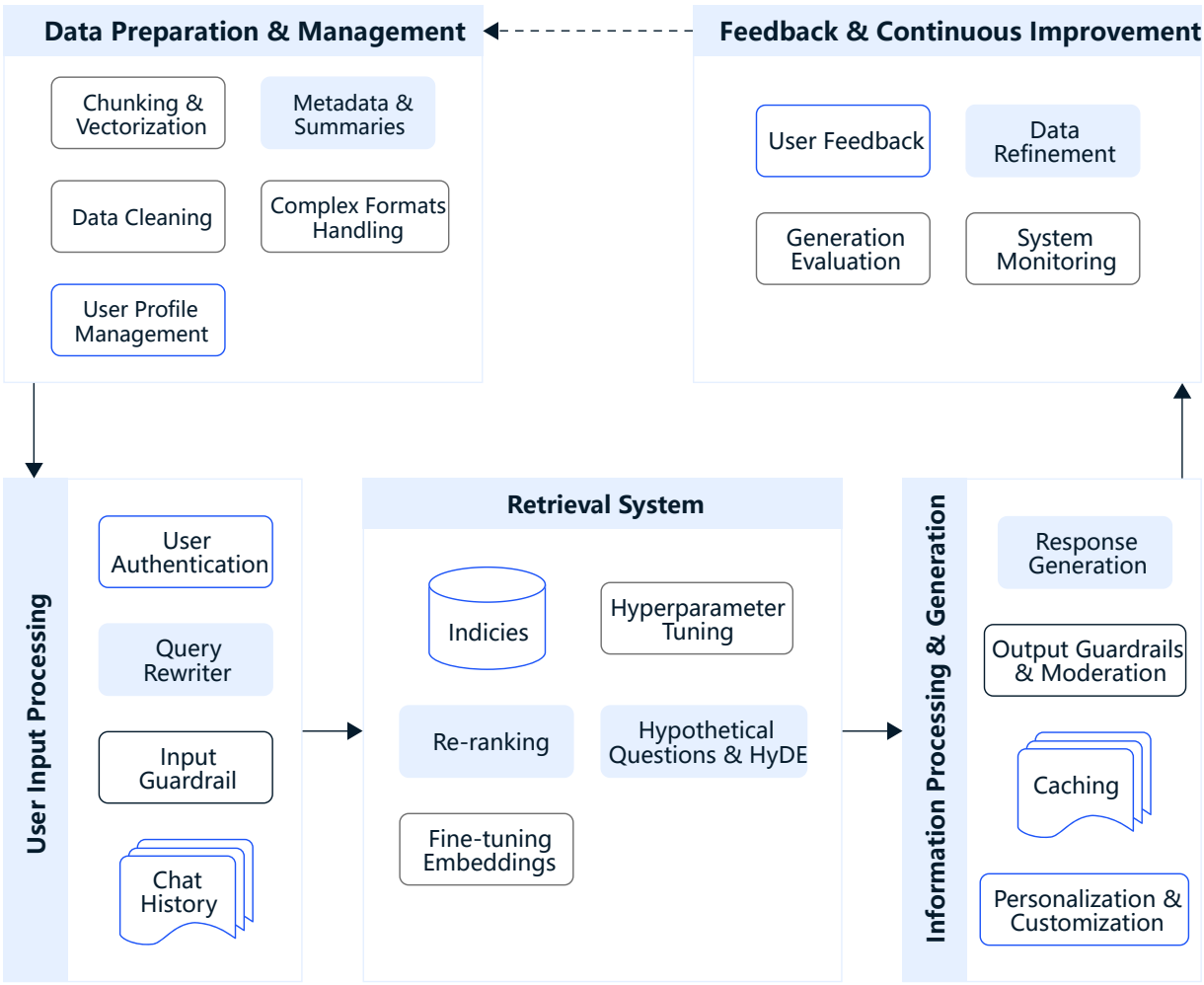
高级 RAG 系统在各组件上的改进

从基础到高级 RAG 的进化意味着系统在存储、检索、增强和生成四个关键组件上都实现了显著改进。

- **存储**：高级 RAG 系统通过语义索引存储数据，按数据的意义而不是简单的关键词进行组织，使得信息检索更加高效。
- **检索**：通过语义搜索和上下文检索的提升，系统不仅能找到相关数据，还能理解用户的意图和上下文。
- **增强**：高级 RAG 系统的增强模块通过动态学习和适应机制，根据用户的交互不断优化，从而生成更加个性化和准确的回应。
- **生成**：生成模块利用复杂的上下文理解和迭代优化，能够生成更连贯且符合上下文的回应。

从基础 RAG 到高级 RAG 的演变是一次重要的飞跃。通过使用复杂的检索技术、增强的上下文整合和动态学习机制，高级 RAG 系统提供了更为准确且具上下文感知的信息检索和生成方式。这种进步提高了 AI 互动的质量，并为更精细和高效的沟通奠定了基础。

企业级高级 RAG 系统的组成部分与工作流程



LeewayHertz

在企业应用领域，越来越需要能够智能检索和生成相关信息的系统。检索增强生成（RAG）系统应运而生，它结合了信息检索的准确性和大语言模型（LLMs）的生成能

力，成为强大的解决方案。但要构建适合企业复杂需求的高级 RAG 系统，必须精心设计其架构。

核心架构组成部分

一个高级的检索增强生成（RAG）系统需要多个核心组成部分，它们共同作用，以确保系统的效率和效果。这些组成部分涵盖了数据管理、用户输入处理、信息检索和生成，以及持续的系统性能提升。以下是这些关键部分的详细分解：

1. **数据准备与管理**

高级 RAG 系统的基础是数据的准备和管理，这涉及多个关键环节：

- **数据分块与向量化：** 数据被分成更易管理的小块，并转化为向量表示，这对于提高检索效率和准确性至关重要。
- **元数据与摘要生成：** 生成元数据和摘要可以快速参考，减少检索时间。
- **数据清理：** 保证数据干净、有序，且无噪音，是确保检索信息准确的关键。
- **处理复杂数据格式：** 系统处理复杂数据格式的能力，确保企业的各种数据类型都能被有效利用。
- **用户配置管理：** 在企业环境中，个性化非常重要，通过管理用户配置，可以根据个人需求定制响应，优化用户体验。

2. **用户输入处理**

用户输入处理模块在确保系统能有效处理查询中起到了至关重要的作用：

- **用户认证：** 企业系统的安全性非常重要，认证机制保证只有授权用户才能使用 RAG 系统。
- **查询优化器：** 用户的查询结构可能并不适合检索，优化器对查询进行优化，提高检索的相关性和准确性。
- **输入防护机制：** 防护机制可保护系统免受无关或恶意输入的干扰，确保检索过程的可靠性。
- **聊天历史利用：** 通过参考先前的对话，系统能更好地理解和响应当前查询，生成更加精准且符合上下文的回答。

3. **信息检索系统**

信息检索系统是 RAG 架构的核心，它负责从预先处理好的数据索引中检索最相关的信息：

- **数据索引：** 高效的索引技术能确保快速且精准的信息检索，先进的索引方式可支持处理大量企业数据。

- **超参数调优：** 对检索模型的参数进行调整，以优化其性能，确保检索到最相关的结果。
- **结果重排序：** 在检索之后，系统会重新排序结果，确保最相关的信息优先展示，提升响应质量。
- **嵌入优化：** 通过调整嵌入向量，系统能够更好地将查询与相关数据匹配，从而提高检索的准确性。
- **假设问题与 HyDE 技术：** 使用 HyDE（假设文档嵌入）技术生成假设问题和答案对，能够更好地应对查询和文档不对称时的信息检索。

4. **信息生成与处理**

当相关信息被检索到之后，系统需要生成连贯且上下文相关的回应：

- **响应生成：** 利用先进的大语言模型（LLMs），该模块将检索到的信息合成为全面且准确的回应。
- **输出防护和审核：** 为了保证生成的回应符合规范，系统会使用各种规则进行审核。
- **数据缓存：** 经常访问的数据或回答会被缓存，从而减少检索时间，提升系统效率。
- **个性化生成：** 系统会根据用户的需求和配置，定制生成的内容，确保回应的相关性和准确性。

5. **反馈与系统优化**

高级 RAG 系统应具备自我学习和改进能力，反馈机制对于持续优化至关重要：

- **用户反馈：** 通过收集和分析用户反馈，系统可以找出需要改进的地方，并不断演进以更好地满足用户需求。
- **数据优化：** 基于用户反馈和新发现，系统中的数据会不断优化，确保信息的质量和相关性。
- **生成质量评估：** 系统会定期评估生成内容的质量，以便持续优化。
- **系统监控：** 持续监控系统性能，确保其高效运行，并能应对需求变化或数据模式的变化。

与企业系统的集成

要让高级 RAG 系统在企业环境中发挥最佳效果，与现有系统的无缝集成非常重要：

- **CRM 和 ERP 系统集成：** 将高级 RAG 系统与客户关系管理 (CRM) 和企业资源规划 (ERP) 系统对接，可以高效地访问和利用关键业务数据，从而提升生成准确且与上下文相关的回答的能力。
- **API 和微服务架构：** 使用灵活的 API 和微服务架构可以使 RAG 系统轻松集成到现有的企业软件中，实现模块化升级和扩展。

安全性和合规性

由于企业数据的敏感性，安全性和合规性显得尤为重要：

- **数据安全协议：** 采用强大的数据加密和安全的数据处理措施，以保护敏感信息，并确保符合如 GDPR 等数据保护法规。
- **访问控制和认证：** 实施安全的用户认证和基于角色的访问控制机制，以确保只有授权人员可以访问或修改系统。

可扩展性和性能优化

企业级 RAG 系统需要具备良好的可扩展性，并能在高负载情况下保持良好的性能：

- **云原生架构：** 采用云原生架构提供按需扩展资源的灵活性，保证系统的高可用性和性能优化。
- **负载均衡和资源管理：** 高效的负载均衡和资源管理策略帮助系统处理大量用户请求和数据，同时保持最佳性能。

分析和报告

高级 RAG 系统还应具备强大的分析和报告功能：

- **性能监控：** 通过集成高级分析工具，实时监控系统的性能、用户互动情况以及系统健康状态，这对于维持系统效率至关重要。
- **商业智能集成：** 与商业智能工具的集成可以提供有价值的见解，帮助决策制定并推动商业战略的发展。

企业级的高级 RAG 系统代表了尖端的 AI 技术、强大的数据处理机制、安全且可扩展的基础设施以及无缝的集成能力的综合体。通过融合这些元素，企业能够构建出既能高效检索和生成信息，又能成为企业技术体系核心部分的 RAG 系统。这些系统不仅带来了显著的业务价值，还改善了决策过程，并提升了整体运营效率。

高级 RAG 技术

高级检索增强生成 (RAG) 涵盖了一系列技术手段，旨在提升各个处理阶段的效率和准确性。这些高级 RAG 系统通过在从索引和查询转换到检索和生成的不同阶段应用先进技术，能够更好地管理数据并提供更精准、符合上下文的回应。下面是一些用于优化 RAG 过程每个阶段的高级技术：

1. 索引

索引是一个关键过程，它提高了大语言模型 (LLMs) 系统的准确性和效率。索引不仅仅是存储数据，还包括系统化地组织和优化数据，以确保信息易于访问和理解，同时保持重要的上下文。有效的索引有助于精确和高效地检索数据，使 LLMs 能够提供相关和准确的回应。索引过程中使用的一些技术包括：

技术 1: 通过块优化来优化文本块
块优化的目的是调整文本块的大小和结构，以便在保持上下文的同时，避免文本块过大或过小，从而提高检索效果。

技术 2: 使用高级嵌入模型将文本转换为向量

创建文本块后，接下来的步骤是将这些块转换为向量表示。这一过程将文本转化为能够捕捉其语义含义的数值向量。像 BGE-large 或 E5 嵌入系列这样的模型能够有效地表现文本的细微差别。这些向量表示在后续的检索和语义匹配中至关重要。

技术 3: 通过嵌入微调来增强语义匹配
嵌入微调的目的是改进嵌入模型对索引数据的语义理解，从而提高检索信息与用户查询之间的匹配准确性。

技术 4: 通过多表示来提高检索效率
多表示技术将文档转换为轻量级的检索单元，例如摘要，以加速检索过程，并提高在处理大文档时的准确性。

技术 5: 使用层次索引来组织数据
层次索引通过像 RAPTOR 这样的模型将数据结构化为多个层次，从详细到一般，提供广泛和精确的上下文信息，提升检索效果。

技术 6: 通过元数据附加来增强数据检索
元数据附加技术向每个数据块添加额外信息，以改善分析和分类能力，使得数据检索更加系统和符合上下文。

2. 查询转换

查询转换旨在优化用户输入，提高信息检索的质量。通过利用 LLMs，转换过程能够将复杂或模糊的查询变得更加清晰和具体，从而提升整体搜索的效率和准确性。

技术 1: 使用 HyDE (假设文档嵌入) 提升查询清晰度
HyDE 通过生成假设数据来增强问题与参考内容之间的语义相似性，从而改善信息检索的相关性和准确性。

技术 2: 通过多步查询简化复杂查询
多步查询将复杂问题拆解成更简单的子问题，分别检索每个子问题的答案，并将结果汇

总，以提供更准确和全面的回应。

技术 3: 通过回溯提示增强上下文
回溯提示技术从复杂的原始查询生成一个更广泛的一般性查询，这样的上下文有助于为具体查询提供基础，通过结合原始查询和广泛查询的结果来改善最终回应。

技术 4: 通过查询重写提高检索效果
查询重写技术利用 LLM 来重新表述初始查询，以提高检索的效果。LangChain 和 LlamaIndex 都采用了这种技术，其中 LlamaIndex 提供了特别强大的实现，大幅提升了检索效果。

3. 查询路由

查询路由的作用是根据查询的特性，将查询发送到最合适的数据源，从而优化检索过程，确保每个查询都由最合适的系统组件处理。

技术 1: 逻辑路由
逻辑路由通过分析查询的结构来选择最适合的数据源或索引，从而优化检索。这种方法确保查询由最适合提供准确答案的数据源处理。

技术 2: 语义路由
语义路由通过分析查询的语义意义，指导查询到正确的数据源或索引。它通过理解查询的上下文和含义，尤其是对于复杂或细微的问题，提高了检索的准确性。

4. 预检索和数据索引技术

预检索优化提升了数据索引或知识库中信息的质量和可检索性。具体的优化方法依据数据的性质、来源和规模而不同。例如，提高信息密度可以用更少的 Token 生成更准确的响应，从而改善用户体验并降低成本。然而，适用于某一系统的优化方法可能不适用于其他系统。大语言模型 (LLMs) 提供了测试和调整这些优化的工具，可以量身定制方法，提高在不同领域和应用中的检索效果。

技术 1: 使用 LLMs 提高信息密度
优化 RAG 系统的一个基础步骤是提升数据在索引前的质量。通过利用 LLMs 进行数据清理、标记和总结，可以提高信息密度，从而带来更准确和高效的数据处理结果。

技术 2: 层次索引检索
层次索引检索通过创建文档摘要作为第一层过滤器来简化搜索过程。这种多层次的方法确保只有最相关的数据在检索阶段被考虑，从而提高检索效率和准确性。

技术 3: 通过假设问答对提高检索对称性
为了解决查询和文档之间的不对称问题，这种技术使用 LLMs 从文档中生成假设的问答

对。通过将这些问答对嵌入检索，系统能更好地匹配用户查询，从而提高语义相似性，减少检索错误。

技术 4: 使用 LLMs 去重
重复信息可能对 RAG 系统有利也有弊。使用 LLMs 去重数据块，可以优化数据索引，减少噪声，提高生成准确响应的可能性。

技术 5: 测试和优化分块策略
有效的分块策略对检索至关重要。通过进行不同分块大小和重叠比率的 A/B 测试，可以找到适合特定用例的最佳平衡。这有助于在保留足够上下文的同时，不至于使相关信息过于分散或稀释。

技术 6: 使用滑动窗口索引
滑动窗口索引通过在索引过程中重叠数据块，确保片段之间不会丢失重要的上下文信息。这种方法保持了数据的连续性，提高了检索信息的相关性和准确性。

技术 7: 提高数据粒度
提升数据粒度主要通过应用数据清理技术，去除无关信息，只保留最准确和最新的内容在索引中。这能提高检索质量，确保只考虑相关的信息。

技术 8: 添加元数据
添加元数据，如日期、目的或章节，可以提高检索的精准度，使系统能够更有效地聚焦于最相关的数据，改善整体检索效果。

技术 9: 优化索引结构
优化索引结构涉及调整分块大小和采用多重索引策略，例如句子窗口检索，以增强数据的存储和检索方式。通过嵌入单个句子同时保持上下文窗口，这种方法在推理过程中能实现更丰富和更具上下文准确性的检索。

5. 检索技术

在检索阶段，系统收集回答用户查询所需的信息。先进的检索技术确保检索到的内容既全面又具备完整的上下文，为后续的处理步骤奠定坚实的基础。

技术 1: 使用 LLMs 优化搜索查询
LLMs 可以优化用户的搜索查询，使其更符合搜索系统的要求，无论是简单的搜索还是复杂的对话查询。这种优化确保检索过程更有针对性和高效。

技术 2: 使用 HyDE 修正查询与文档的不对称
通过生成假设的回答文档，HyDE 技术提高了检索中的语义相似性，解决了短查询和长文档之间的不对称问题。

技术 3: 实施查询路由或 RAG 决策模式
在使用多个数据源的系统中，查询路由将搜索指向合适的数据库，从而优化检索效率。

RAG 决策模式进一步优化这个过程，通过确定何时需要检索来节省资源，当大语言模型可以独立响应时。

技术 4: 使用递归检索器进行深度探索
递归检索器基于前一个结果进行进一步查询，适合深入探索相关数据，获取详细或全面的信息。

技术 5: 使用路由检索器优化数据源选择
路由检索器利用 LLM 动态选择最适合的数据源或查询工具，根据查询的上下文提高检索过程的效果。

技术 6: 使用自动检索器自动生成查询
自动检索器利用 LLM 自动生成元数据过滤器或查询语句，从而简化数据库查询流程，优化信息检索。

技术 7: 使用融合检索器结合结果
融合检索器将来自多个查询和索引的结果合并，提供全面且不重复的信息视图，确保检索的全面性。

技术 8: 使用自动合并检索器聚合数据上下文
自动合并检索器将多个数据片段合并为一个统一的上下文，通过整合较小的上下文提高信息的相关性和完整性。

技术 9: 微调嵌入模型
微调嵌入模型使其更适应特定领域，提高处理专业术语的能力。这种方法通过更紧密地对齐领域特定内容，增强了检索信息的相关性和准确性。

技术 10: 实施动态嵌入
动态嵌入通过根据上下文调整词向量，超越静态表示，提供更细致的语言理解。这种方法，如 OpenAI 的 embeddings-ada-02 模型，能更准确地捕捉上下文含义，从而提供更准确的检索结果。

技术 11: 利用混合搜索
混合搜索结合了向量搜索和传统的关键词匹配，允许同时进行语义相似性和精确术语识别。这种方法在需要精确术语识别的场景中尤为有效，确保全面和准确的检索。

6. 检索后的技术

在获取到相关内容之后，检索后的阶段主要关注如何将这些内容有效地整合在一起。这一步骤包括向大语言模型（LLM）提供精确且简洁的上下文信息，确保系统拥有生成连贯、准确回答所需的所有细节。这种整合的质量直接决定了最终输出的相关性和清晰度。

技术 1: 通过重排序优化搜索结果
在检索之后，重排序模型可以重新排列搜索结果，将最相关的文档放在更靠近查询的位置，从而提升提供给 LLM 的信息质量，进而改善最终响应的生成效果。重排序不仅减少了需要提供给 LLM 的文档数量，还起到了过滤的作用，提升了语言处理的准确性。

技术 2: 通过上下文提示压缩优化搜索结果
LLM 可以在生成最终提示之前对检索到的信息进行过滤和压缩。压缩通过减少冗余的背景信息和去除无关噪音，帮助 LLM 更加关注关键信息。这种优化提高了响应质量，使其集中于重要细节。像 LLMLingua 这样的框架进一步改进了这一过程，通过移除无关的 tokens，使提示更加简洁有效。

技术 3: 通过纠正 RAG 对检索文档进行评分和过滤
在将内容输入 LLM 之前，需要选择和过滤文档，去除无关或准确度低的文档。这项技术确保只使用高质量、相关的信息，从而提高响应的准确性和可靠性。纠正 RAG 利用如 T5-Large 这样的模型来评估检索文档的相关性，并过滤掉那些低于预设阈值的文档，确保只有有价值的信息参与最终响应的生成。

7. 生成技术

在生成阶段，检索到的信息会被评估和重新排序，以确定最重要的内容。先进的技术在这一阶段涉及选择那些能够提高响应相关性和可靠性的关键细节。这一过程确保生成的内容不仅回答了查询，还能以有意义的方式得到检索数据的良好支持。

技术 1: 使用 Chain-of-Thought 提示减少噪音
Chain-of-thought 提示帮助 LLM 处理嘈杂或无关的背景信息，即使数据中存在干扰，也能提高生成准确响应的可能性。

技术 2: 通过自我 RAG 使系统具备自我反思能力
自我 RAG 涉及训练模型在生成过程中使用反思 tokens，这样可以实时评估和改进自己的输出，根据事实性和质量选择最佳回应。

技术 3: 通过微调忽略无关背景
专门对 RAG 系统进行微调，以增强 LLM 忽略无关背景的能力，确保只有相关信息影响最终响应。

技术 4: 利用自然语言推理增强 LLM 对无关背景的鲁棒性
集成自然语言推理（NLI）模型有助于通过将检索的背景与生成的答案进行比较，来过滤掉无关的背景信息，确保只有相关的信息影响最终输出。

技术 5: 使用 FLARE 控制数据检索
FLARE（灵活语言模型适应用于检索增强）是一种基于提示工程的方法，确保 LLM 仅在必要时检索数据。它不断调整查询并检查低概率关键词，触发相关文档的检索，以提升响应的准确性。

技术 6: 使用 ITER-RETGEN 提升响应质量
ITER-RETGEN（迭代检索-生成）通过迭代执行生成过程来提高响应质量。每次迭代都使用前一次的结果作为背景，检索更相关的信息，从而持续改进最终响应的质量和相关性。

技术 7: 使用 ToC（澄清树）澄清问题
ToC 递归地生成具体问题，以澄清初始查询中的模糊之处。这种方法通过不断评估和完善原始问题，精细化问答过程，使最终响应更加详细和准确。

8. 评估

在高级检索增强生成（RAG）技术中，评估过程至关重要，用于确保检索和综合的信息既准确又与用户的查询相关。评估过程包括两个关键组成部分：质量评分和所需能力。

质量评分专注于测量内容的精确度和相关性：

- **背景相关性:** 评估检索或生成的信息在查询的具体背景中的适用程度。确保响应准确且符合用户的需求。
- **答案忠实度:** 检查生成的答案是否准确反映了检索的数据，没有引入错误或误导信息。这对于保持系统输出的可靠性至关重要。
- **答案相关性:** 评估生成的响应是否直接有效地回答了用户的查询，确保答案既有用又符合问题的要点。

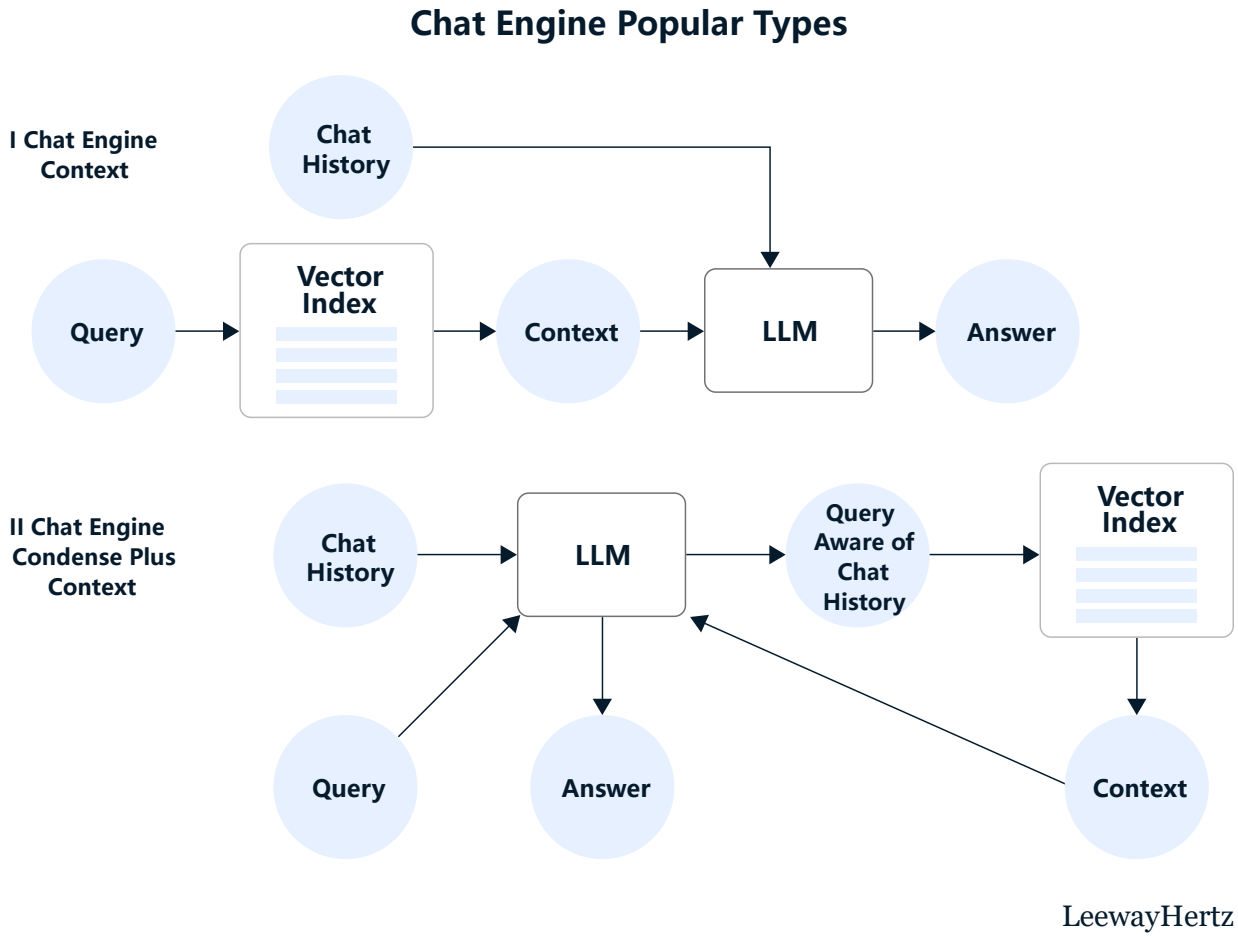
所需能力是系统必须具备的能力，以提供高质量的结果：

- **噪音鲁棒性:** 衡量系统过滤无关或嘈杂数据的能力，确保这些干扰不会影响最终响应的质量。
- **负面拒绝:** 测试系统识别和排除错误或无关信息的效果，防止其污染生成的输出。
- **信息整合:** 评估系统将多个相关信息整合成连贯、全面响应的能力，为用户提供一个完整的答案。
- **反事实鲁棒性:** 检查系统在处理假设性或反事实场景中的表现，确保即使在处理推测性问题时，响应仍然准确可靠。

这些评估组件共同确保高级 RAG 系统提供的响应既准确又相关，具备鲁棒性、可靠性，并且针对用户的特定需求进行了定制。

附加技术

聊天引擎：提升 RAG 系统中的对话能力



将聊天引擎集成到高级检索增强生成 (RAG) 系统中，可以提升系统处理后续问题和维持对话上下文的能力，这与传统的聊天机器人技术类似。不同的实现方式提供了不同的复杂程度：

- **上下文聊天引擎：** 这种基础方法通过检索与用户查询相关的上下文，包括之前的聊天记录，来指导大语言模型 (LLM) 的回应。这样可以确保对话的连贯性和上下文的适当性。
- **浓缩加上下文模式：** 这是一种更高级的方法，将每次交互的聊天记录和最新消息浓缩成一个优化的查询。这个精炼的查询可以获取相关的上下文，并与原始用户消息结合，提供给 LLM，以生成更准确、更具上下文的回应。

这些实现方式有助于提升 RAG 系统中对话的连贯性和相关性，并根据不同的需求提供不同层次的复杂性。

参考文献引用：确保来源准确

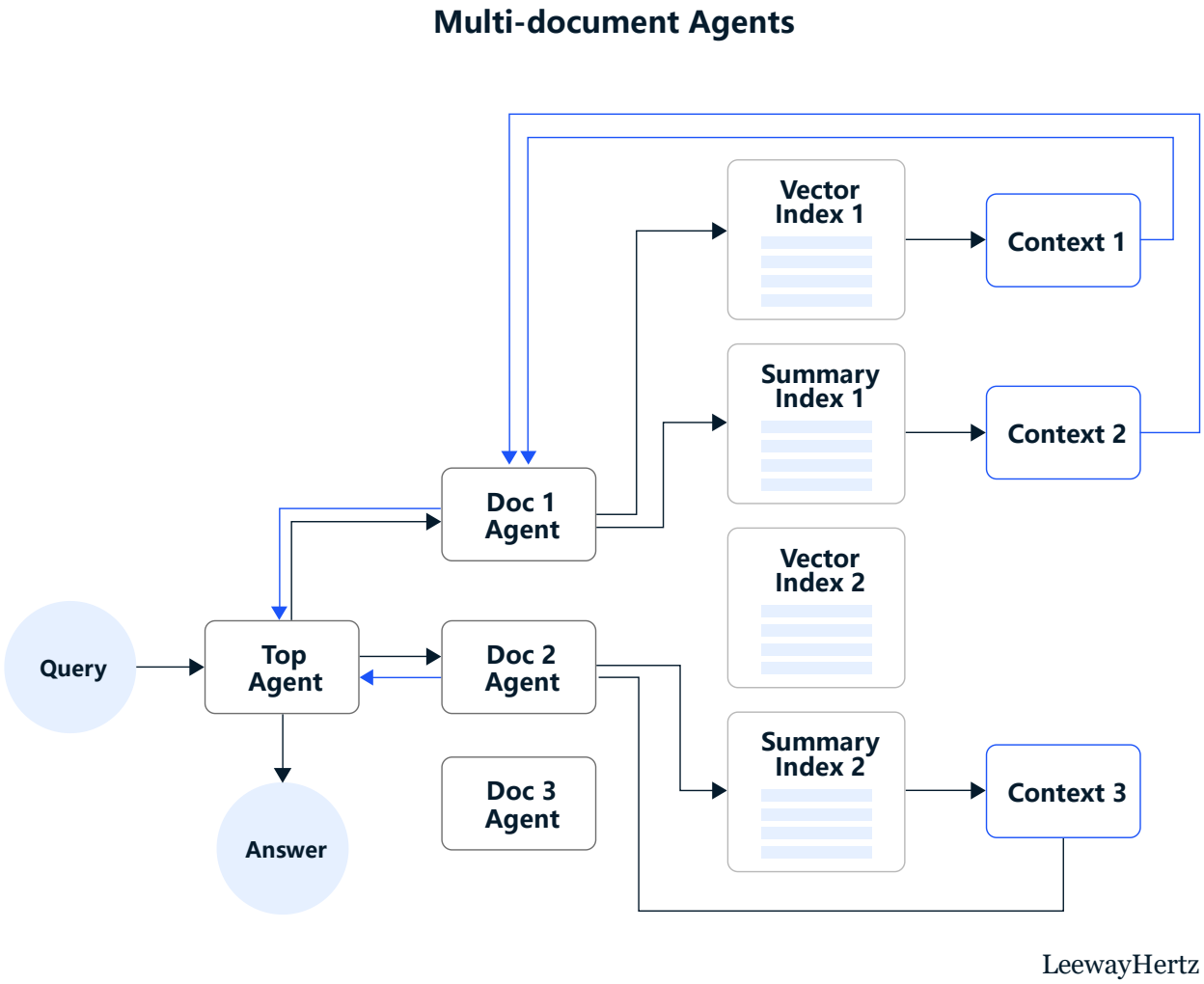
确保参考文献的准确性非常重要，特别是当多个来源对生成的回答有贡献时。可以通过以下几种方法实现：

1. **直接来源标注：** 在语言模型 (LLM) 的提示中设置任务，要求在生成的回应中直接标明来源。这种方式可以明确标注原始来源。

2. **模糊匹配技术：** 采用模糊匹配技术，例如 LlamaIndex 使用的，来将生成内容的部分与源索引中的文本块对齐。模糊匹配可以提高内容的准确性，确保其反映来源信息。

通过应用这些策略，可以显著提升参考文献引用的准确性和可靠性，确保生成的回应既可信又有充分的支持。

在检索增强生成 (RAG) 中的代理



代理在提升检索增强生成 (RAG) 系统的性能中扮演着重要角色，它们通过为大语言模型 (LLM) 提供额外的工具和功能来拓展其应用范围。最初这些代理是通过 LLM API 引入的，使得 LLM 能够利用外部代码函数、API 甚至其他 LLM，从而增强其功能。

代理的一个重要应用是在多文档检索中。例如，最近 OpenAI 的助手展示了这一概念的进展。这些助手通过集成聊天记录、知识存储、文档上传接口以及将自然语言转换为可操作命令的功能调用 API 等特性，增强了传统 LLM 的功能。

代理的使用还扩展到了多个文档的管理，其中每个文档由一个专门的代理处理，例如总结和问答。一个中央的高级代理负责监督这些文档专用的代理，路由查询并整合回应。这种设置支持在多个文档间进行复杂的比较和分析，展示了先进的 RAG 技术。

回应合成器：制作最终答案

RAG 流程中的最后一步是将检索到的上下文和初始用户查询合成为回应。除了直接将上下文与查询结合并通过 LLM 处理之外，更精细的方法包括：

1. **迭代优化：** 将检索到的上下文分割成较小的部分，通过多次与 LLM 的交互来优化回应。
2. **上下文总结：** 将大量的上下文压缩到适合 LLM 提示的范围内，以确保回应保持集中和相关。
3. **多答案生成：** 从上下文的不同片段生成多个回应，然后将这些回应整合成一个统一的答案。

这些技术提升了 RAG 系统回应的质量和准确性，展示了回应合成中的高级方法的潜力。

采用这些先进的 RAG 技术可以显著提升系统的性能和可靠性。通过优化每个阶段的流程，从数据预处理到回应生成，企业可以创建更准确、高效和强大的 AI 应用。

高级 RAG 的应用与案例

高级检索增强生成 (RAG) 系统在许多领域都有广泛应用，通过其强大的数据处理和生成能力，提升了数据分析、决策制定和用户互动的效果。从市场研究到客户支持再到内容创作，高级 RAG 系统在多个方面都展现了显著的优势。下面介绍了这些系统在不同领域的具体应用：

1. 市场研究与竞争分析

- **数据整合**：RAG 系统能够从社交媒体、新闻文章和行业报告等多种来源整合并分析数据。
- **趋势识别**：通过处理大量数据，RAG 系统能够识别市场的新兴趋势和消费者行为的变化。
- **竞争者洞察**：系统提供详细的竞争对手策略和业绩分析，帮助企业进行自我评估和对标。
- **可操作的洞察**：企业可以利用这些报告进行战略规划和决策。

2. 客户支持与互动

- **上下文感知的回答**：RAG 系统通过检索知识库中的相关信息，给客户提供准确且符合上下文的回答。

- **减轻工作负担**：自动化处理常见问题，减轻了人工支持团队的压力，让他们可以处理更复杂的问题。
- **个性化服务**：系统通过分析客户的历史记录和偏好，定制化回应和互动，满足个人需求。
- **提升互动体验**：高质量的支持服务提升了客户满意度，并加强了客户关系。

3. 监管合规与风险管理

- **法规分析**：RAG 系统扫描和解读法律文件及监管指南，确保合规性。
- **风险识别**：通过对比内部政策与外部法规，系统迅速识别潜在的合规风险。
- **合规建议**：提供实际的建议，帮助企业填补合规空白，降低法律风险。
- **高效报告**：生成易于审计和检查的合规报告和摘要。

4. 产品开发与创新

- **客户反馈分析**：RAG 系统分析客户反馈，发现常见问题和痛点。
- **市场需求洞察**：跟踪新兴趋势和客户需求，指导产品开发。
- **创新建议**：根据数据分析提供潜在的产品功能和改进建议。
- **竞争定位**：帮助企业开发符合市场需求且在竞争中脱颖而出的产品。

5. 金融分析与预测

- **数据整合**：RAG 系统整合金融数据、市场情况和经济指标，进行全面分析。
- **趋势分析**：识别金融市场中的模式和趋势，辅助预测和投资决策。
- **投资建议**：提供有关投资机会和风险因素的实际建议。
- **战略规划**：通过准确预测和数据驱动的建议，支持战略财务决策。

6. 语义搜索与高效信息检索

- **上下文理解**: RAG 系统通过理解用户查询的背景和含义来进行语义搜索。
- **相关结果**: 从海量数据中检索出最相关和最准确的信息，提高搜索效率。
- **节省时间**: 优化数据检索流程，减少寻找信息的时间。
- **提升准确性**: 比传统的关键词搜索方法提供更精准的搜索结果。

7. 提升内容创作

- **趋势整合**: RAG 系统利用最新的数据，确保生成的内容符合当前市场趋势和观众兴趣。
- **自动生成内容**: 根据主题和目标受众自动生成内容创意和草稿。
- **增强参与度**: 生成更具吸引力和相关性的内容，提升用户互动效果。
- **及时更新**: 确保内容反映最新事件和市场动态，保持时效性。

8. 文本摘要

- **高效摘要**: RAG 系统能对长篇文档进行有效摘要，提炼出关键点和重要发现。
- **节省时间**: 为忙碌的高管和经理提供简洁的报告摘要，节省阅读时间。
- **重点突出**: 突出关键信息，帮助决策者快速把握要点。
- **决策效率提升**: 以易于理解的方式提供相关信息，提高决策效率。

9. 高级问答系统

- **精准回答**: RAG 系统从广泛的信息源中提取数据，生成对复杂问题的准确答案。
- **访问提升**: 提升对各个领域信息的访问，如医疗保健或金融。
- **上下文相关**: 根据用户的具体需求和问题，提供有针对性的答案。
- **复杂问题处理**: 通过整合多个信息来源，处理复杂问题。

10. 对话代理和聊天机器人

- **上下文信息**: RAG 系统通过提供相关的背景信息来增强聊天机器人和虚拟助手的互动能力。
- **提升准确性**: 确保对话代理的回答准确、信息丰富。
- **用户支持**: 提供智能且响应迅速的对话界面，提升用户帮助体验。
- **互动自然**: 实时检索相关数据，让互动更自然、更具吸引力。

11. 信息检索

- **高级搜索**: 通过 RAG 的检索和生成能力，提升搜索引擎的准确性。
- **信息片段生成**: 生成有效的信息片段，增强用户体验。
- **搜索结果增强**: 利用 RAG 系统生成的答案丰富搜索结果，提高查询解决效果。
- **知识引擎**: 利用公司数据回答内部问题，如人力资源政策或合规问题，便于信息获取。

12. 个性化推荐

- **分析客户数据**: 通过分析过往的购买记录和评价，生成个性化的产品推荐。
- **提升购物体验**: 根据个人喜好推荐产品，改善用户购物体验。
- **增加收入**: 根据客户行为推荐相关产品，提升销售额。
- **市场对接**: 将推荐内容与当前市场趋势对接，满足不断变化的客户需求。

13. 文本补全

- **上下文补全**: RAG 系统以符合上下文的方式完成部分文本。
- **提高效率**: 提供准确的补全，简化任务如邮件撰写或代码编写。
- **提升生产力**: 减少完成写作和编码任务的时间，提升生产力。
- **保持一致性**: 确保文本补全与现有内容和语调一致。

14. 数据分析

- **全面数据整合**: RAG 系统整合内部数据库、市场报告及外部来源的数据，提供全面视图并进行深入分析。
- **精准预测**: 通过分析最新数据、趋势和历史信息，提高预测的准确性。
- **洞察发现**: 分析综合数据集，识别和评估新机会，为增长和改进提供有价值的见解。
- **数据驱动推荐**: 通过分析全面数据集提供数据驱动的建议，支持战略决策，提升整体决策质量。

15. 翻译任务

- **检索翻译**: 从数据库中检索相关翻译，以帮助完成翻译任务。
- **上下文生成**: 根据上下文生成一致的翻译，并参考检索到的语料库。
- **提升准确性**: 利用多个来源的数据提高翻译的准确性。
- **提高效率**: 通过自动化和上下文感知的生成简化翻译过程。

16. 客户反馈分析

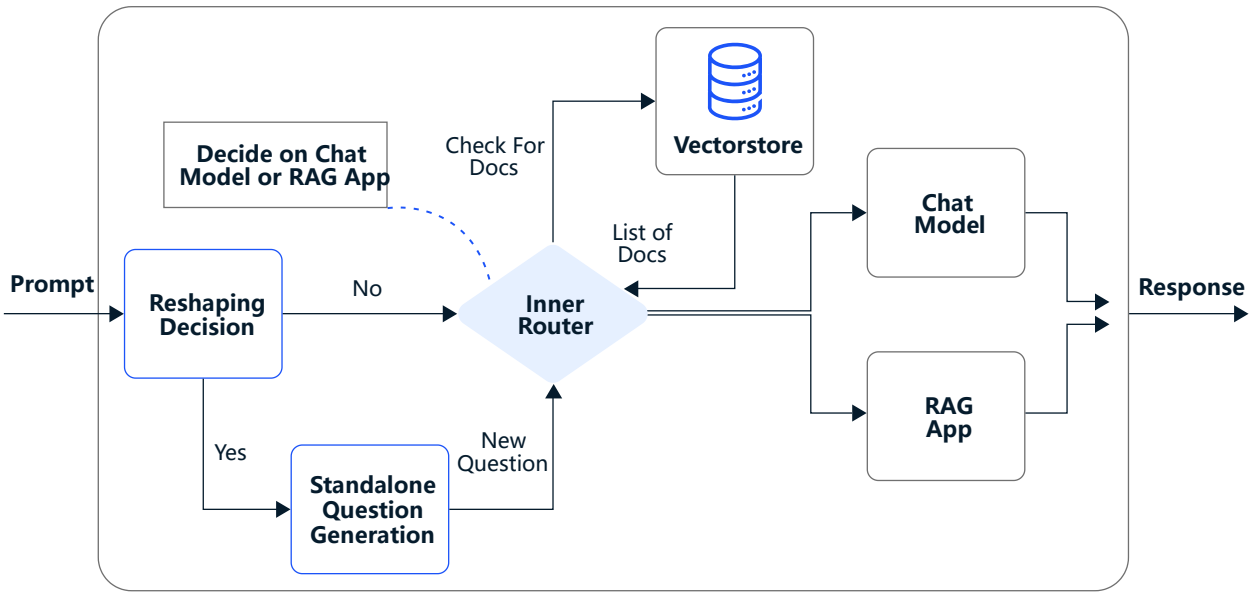
- **全面分析**: 从不同来源分析反馈，全面了解客户情绪和问题。
- **深入见解**: 提供详细的见解，揭示重复的主题和客户痛点。
- **数据整合**: 整合来自内部数据库、社交媒体和评论的反馈，进行全面分析。
- **信息化决策**: 基于客户反馈做出更快、更明智的决策，改进产品和服务。

这些应用展示了先进 RAG 系统的广泛可能性，体现了它们在提高效率、准确性和洞察力方面的能力。无论是提升客户支持、增强市场研究还是简化数据分析，先进的 RAG 系统都能提供推动战略决策和运营卓越的宝贵解决方案。

使用先进的 RAG 构建对话工具

对话 AI 工具在现代用户互动中扮演着至关重要的角色，它们能够在各种平台上提供生动和迅速的反馈。通过集成先进的检索增强生成（RAG）系统，我们可以将这些工具的能力提升到一个全新的高度。RAG 系统将强大的信息检索功能与先进的生成技术结合，确保对话既具备丰富的信息性，又保持自然的交流流畅性。当 RAG 系统融入对话 AI 工具时，它能为用户提供准确且上下文丰富的回答，同时保持自然的对话节奏。本节将探讨如何利用 RAG 构建高级对话工具，重点介绍构建这些系统时需要关注的关键要素，以及如何使它们在实际应用中有效且实用。

设计对话流程



LeewayHertz

任何对话工具的核心都是其对话流程——也就是系统如何处理用户输入并生成响应的步骤。对于基于高级 RAG 的工具，对话流程的设计需要精心策划，以充分发挥 RAG 系统的检索能力和语言模型的生成优势。这个流程通常包括几个关键阶段：

1. **问题评估和重塑**：
    - 系统首先会评估用户提出的问题，判断是否需要对其进行重塑，以便为准确的回答提供所需的背景。如果问题过于模糊或缺乏关键细节，系统可能会将其重新格式化为一个独立的查询，确保包含所有必要的信息。
2. **相关性检查和路由**：

    - 一旦问题被正确格式化，系统会在向量存储（一个包含索引信息的数据库）中查找相关数据。如果找到相关信息，问题会被转发到 RAG 应用程序，系统会检索所需的信息来生成回答。
    - 如果向量存储中没有相关信息，系统需要决定是继续使用仅由语言模型生成的回答，还是请求 RAG 系统说明无法提供满意的答案。
3. **生成响应**：
    - 根据前一步的决定，系统要么使用检索到的数据生成详细的答案，要么依赖语言模型的知识和对话历史来回应用户。这种方式确保工具能够处理实际问题，同时也适应更随意的开放式对话。

利用决策机制优化对话流程

构建高级 RAG 对话工具时，一个重要的方面是实现控制对话流程的决策机制。这些机制帮助系统智能地决定何时检索信息、何时依赖生成能力，以及何时告知用户没有相关数据。通过这些决策，工具能更加灵活，适应各种对话场景。

- **决策点 1：重塑还是继续?**
  系统首先决定用户的问题是否可以按原样处理，还是需要进行重塑。这一步确保系统理解用户的意图，并拥有所有必要的上下文，以便在生成回应前能够进行有效的检索或生成。
- **决策点 2：检索还是生成?**
  在需要重塑的情况下，系统会判断向量存储中是否有相关信息。如果找到相关数据，系统将使用 RAG 进行检索和回答生成。如果没有，系统需决定是否仅靠语言模型生成回答。
- **决策点 3：告知还是互动?**
  如果向量存储和语言模型都无法提供满意的答案，系统会告知用户没有相关信息，从而保持对话的透明度和可信度。

如何为对话型 RAG 设计有效的提示

提示在引导语言模型的对话行为中起着关键作用。设计有效的提示需要对背景信息、互动目标以及期望的风格和语气有明确的了解。例如：

- **背景提示**：提供相关的背景信息，确保语言模型能够在生成或调整问题时掌握必要的上下文。
- **目标导向提示**：明确每个提示的目的，比如调整问题、决定检索过程，或者生成回应。
- **风格和语气**：指定所需的风格（如正式、随意）和语气（如信息性、同理心），以确保语言模型的输出符合用户体验的期望。

使用高级的 RAG 技术构建对话工具需要一种综合的策略，将检索和生成的优势结合起来。通过精心设计对话流程、实施智能决策机制和制定有效的提示，开发者可以创建出既能提供准确且上下文丰富的回答，又能与用户进行自然、富有意义互动的 AI 工具。

如何构建高级 RAG 应用?

开始时构建一个基本的检索增强生成（RAG）应用是很好的，但要在更复杂的场景中充分发挥 RAG 的潜力，你需要超越基础。本节将介绍如何构建高级 RAG 应用，提升检索过程、提高响应准确性，并实施高级技术，如查询重写和多阶段检索。

在深入高级技术之前，我们简要回顾一下 RAG 应用的基本功能。RAG 应用将语言模型（LLM）的能力与外部知识库结合，用于回答用户查询。这个过程通常包括两个阶段：

1. **检索**：应用从向量数据库或其他知识库中搜索与用户查询相关的文本片段。
2. **阅读**：将检索到的文本传递给 LLM，根据这些上下文生成回应。

这种“先检索后阅读”的方法为 LLM 提供了所需的背景信息，从而在面对需要专业知识的查询时能提供更准确的答案。

构建高级 RAG 应用的步骤如下：

第 1 步：使用高级技术提升检索

检索阶段对最终响应的质量至关重要。在基础 RAG 应用中，检索过程相对简单，但在高级 RAG 应用中，你可以使用以下增强技术：

1. 多阶段检索

多阶段检索通过多个步骤精细化搜索，帮助锁定最相关的上下文。通常包括：

- **初步广泛搜索**：从广泛的检索开始，获取一系列可能相关的文档。
- **细化搜索**：基于初步结果进行更精确的检索，缩小到最相关的片段。

这种方法可以提高检索信息的精度，进而提供更准确的答案。

2. 查询重写

查询重写将用户的查询转换为更可能在检索中得到相关结果的格式。可以通过以下几种方式实现：

- **零样本重写**：在没有具体示例的情况下重写查询，依靠模型的语言理解能力。

- **少样本重写**：提供示例以帮助模型重写类似的查询，提高准确性。
- **定制重写器**：对专门用于查询重写的模型进行微调，以更好地处理特定领域的查询。

这些重写后的查询能更好地与知识库中的文档语言和结构匹配，从而提高检索的准确性。

3. 子查询分解

对于涉及多个问题或方面的复杂查询，将查询分解为多个子查询可以提升检索效果。每个子查询关注原问题的某个特定方面，这样系统就能为每个部分检索相关的上下文，并整合答案。

第 2 步：改进回应生成

在你提升了检索过程之后，下一步是优化大语言模型生成回应的方式：

1. 后退提示

面对复杂或多层次的问题时，生成一些额外的、更宽泛的查询可能会很有帮助。这些“后退”提示能够帮助检索更广泛的背景信息，从而让大语言模型生成更全面的回答。

2. 假设文档嵌入 (HyDE)

HyDE 是一种先进的技术，通过根据用户查询生成假设性文档来捕捉查询的意图，然后用这些文档在知识库中找到匹配的真实文档。这种方法特别适合当查询与相关背景在语义上不相似时使用。

第 3 步：整合反馈循环

为了不断提升 RAG 应用程序的性能，将反馈循环集成到系统中非常重要：

1. 用户反馈

加入一个机制，让用户可以评价回应的相关性和准确性。这些反馈可以用来调整检索和生成的过程。

2. 强化学习

利用强化学习技术，根据用户反馈和其他性能指标对模型进行训练。这能够让系统从错误中学习，并逐步提高准确性和相关性。

第 4 步：扩展与优化

随着 RAG 应用程序的不断进步，性能的扩展和优化变得越来越重要：

1. 分布式检索

为了应对大规模的知识库，实施分布式检索系统，这些系统可以在多个节点之间并行处理检索任务，从而降低延迟，提高处理速度。

2. 缓存策略

实施缓存策略以存储频繁访问的上下文块，减少重复检索的需求，并加快响应速度。

3. 模型优化

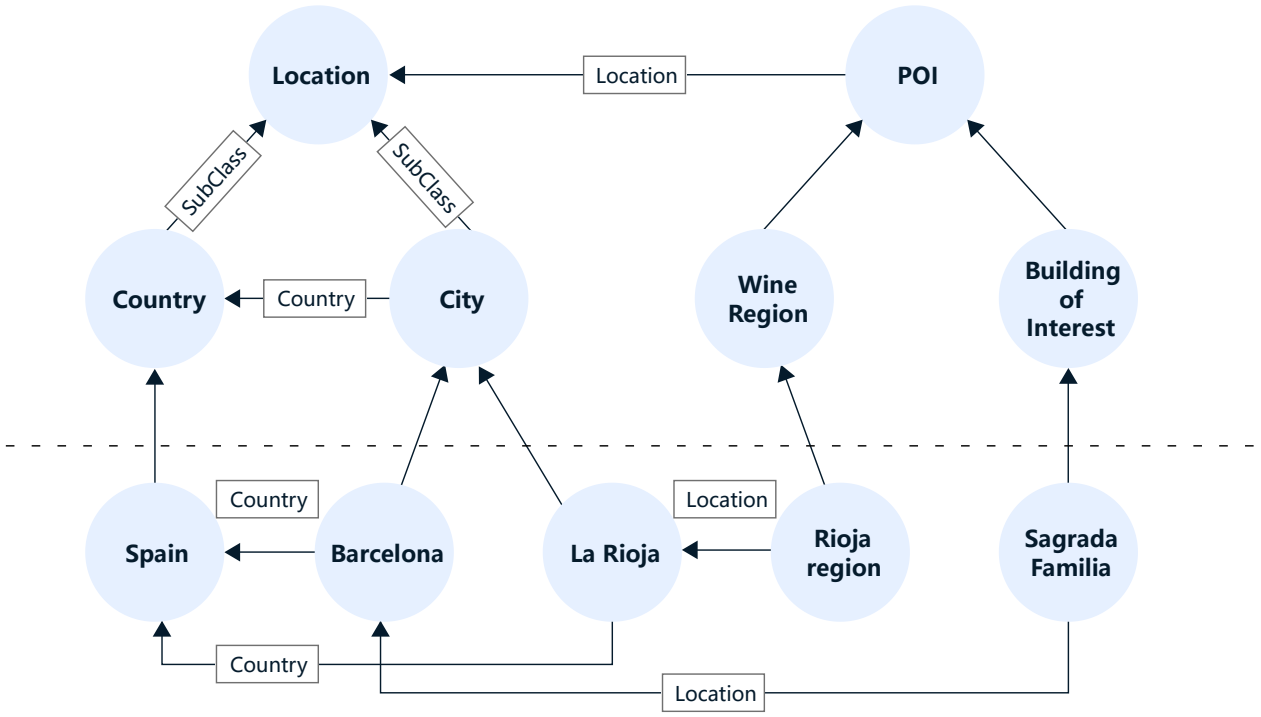
优化大语言模型和应用中使用的其他模型，以减少计算负担，同时保持准确性。模型蒸馏和量化等技术在这里非常有用。

构建一个高级的 RAG 应用程序需要深入了解检索机制和生成模型，同时具备实施和优化复杂技术的能力。按照上述步骤，你可以创建一个先进的 RAG 系统，超越用户的期望，提供高质量、上下文准确的回应，适用于各种应用场景。

知识图谱在高级 RAG 中的崛起

随着企业越来越依赖 AI 来处理复杂的数据驱动任务，知识图谱在高级检索增强生成（RAG）系统中的作用变得尤为重要。根据 Gartner 的说法，知识图谱是未来有望颠覆多个市场的前沿技术之一。Gartner 的新兴技术影响雷达 指出，知识图谱是高级 AI 应用的核心支持工具，它们为数据管理、推理能力和 AI 输出的可靠性提供了基础。这促使了知识图谱在医疗保健、金融、零售等各个行业的广泛应用。

什么是知识图谱?



LeewayHertz

知识图谱是一种结构化的信息表示形式，其中实体（节点）及其之间的关系（边）被明确地定义。这些实体可以是具体的对象（如人和地点）或抽象的概念。实体之间的关系帮助建立一个知识网络，使得数据检索、推理和推断更具人类认知的特点。知识图谱不仅仅是存储数据，而是捕捉了领域内丰富和细致的关系，这使得它成为 AI 应用中一个强大的工具。

利用知识图谱进行查询增强和规划

查询增强是解决 RAG 系统中提问不清晰问题的方法。其目的是为查询添加必要的上下文，确保即使是模糊的问题也能得到准确的解释。例如，在金融领域，像“实施金融监管的当前挑战是什么？”这样的问题可以被增强，加入如“AML 合规”或“KYC 过程”等具体实体，从而使检索过程集中于最相关的信息。

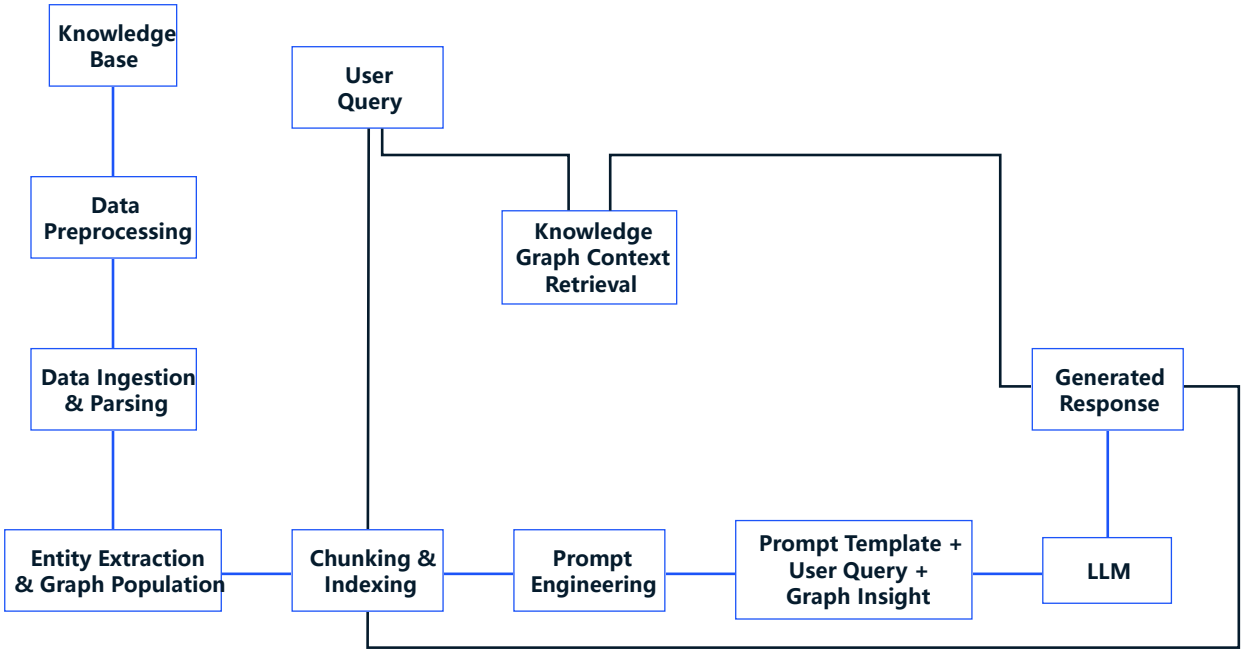
在法律领域，类似“合同相关的风险是什么？”的问题可以通过添加具体的合同类型，如“劳动合同”或“服务协议”，根据知识图谱提供的上下文进行增强。

查询规划则是通过生成子问题将复杂查询分解为可处理的部分。这确保了 RAG 系统可以检索和整合最相关的信息，提供全面的回答。例如，要回答“新的财务报告标准对公司有什么影响？”系统可能会首先检索各个报告标准的数据、实施时间表以及对不同领域的历史影响。

在医疗领域，一个像“最新的医疗设备进展是什么？”的问题可以分解为探索特定领域进展的子问题，如“植入设备”、“诊断设备”或“外科工具”，从而确保系统从每个子类别中获取详细和相关的信息。

通过查询增强和规划，知识图谱帮助优化和结构化查询，提高信息检索的准确性和相关性，最终为金融、法律和医疗等复杂领域提供更精准和有用的答案。

知识图谱在 RAG 中的作用



LeewayHertz

在检索增强生成（RAG）系统中，知识图谱通过提供结构化且丰富上下文的数据，增强了检索和生成过程。传统的 RAG 系统依赖非结构化文本和向量数据库，这可能导致信息检索不准确或不完整。通过整合知识图谱，RAG 系统能够：

1. **改善查询理解：** 知识图谱帮助系统更好地理解查询的上下文和关系，从而更准确地检索相关数据。
2. **增强答案生成：** 知识图谱提供的结构化数据可以生成更连贯、上下文相关的回答，减少 AI 错误的风险。
3. **实现复杂推理：** 知识图谱支持多跳推理，系统可以通过遍历多个关系来推断新知识或连接不同的信息。

知识图谱的主要组成部分

知识图谱主要由以下几个部分构成：

1. **节点：** 代表知识领域中的各种实体或概念，比如人物、地点或事物。
2. **边：** 描述节点之间的关系，说明这些实体是如何相互连接的。
3. **属性：** 与节点和边相关的附加信息或元数据，提供更多背景或细节。

4. **三元组：** 知识图谱的基本构成单元，包含一个主题、一个谓词和一个宾语（例如，“爱因斯坦” [主题] “发现” [谓词] “相对论” [宾语]），这些三元组构建了描述实体间关系的基本框架。

知识图谱-RAG 方法论

KG-RAG 方法论包括三个主要步骤：

1. **KG 构建：** 这一步是将非结构化的文本数据转化为结构化的知识图谱，确保数据的组织和关联性。
2. **检索：** 使用一种称为探索链（CoE）的新型检索算法，系统通过知识图谱进行数据检索。
3. **响应生成：** 最后，利用检索到的信息生成连贯且符合上下文的回答，结合了知识图谱的结构化数据和大语言模型的能力。

这个方法论突出了结构化知识在提升 RAG 系统检索和生成过程中的重要作用。

知识图谱在 RAG 中的好处

将知识图谱融入 RAG 系统带来了几个显著的优势：

1. **结构化知识表示：** 知识图谱以反映实体间复杂关系的方式组织信息，使得数据检索和使用变得更加高效。
2. **上下文理解：** 知识图谱通过捕捉实体之间的关系，提供了更加丰富的背景信息，使 RAG 系统能够生成更相关且连贯的回答。
3. **推理能力：** 知识图谱帮助系统通过分析实体间的关系推断出新的知识，从而生成更全面和准确的回答。
4. **知识整合：** 知识图谱能够整合来自不同来源的信息，提供更加全面的数据视图，帮助做出更好的决策。
5. **可解释性和透明性：** 知识图谱的结构化特性使得推理路径清晰易懂，便于解释结论的形成过程，提高系统的可信度。

将 KG 与 LLM-RAG 集成

将知识图谱与大语言模型（LLM）结合使用于 RAG 系统，可以增强整体的知识表示和推理能力。这种结合实现了动态知识融合，确保在推理时信息保持最新和相关，从而生成更加准确和有洞察力的回答。LLM 可以利用结构化数据和非结构化数据提供更优质的结果。

在思维链问答中使用知识图谱

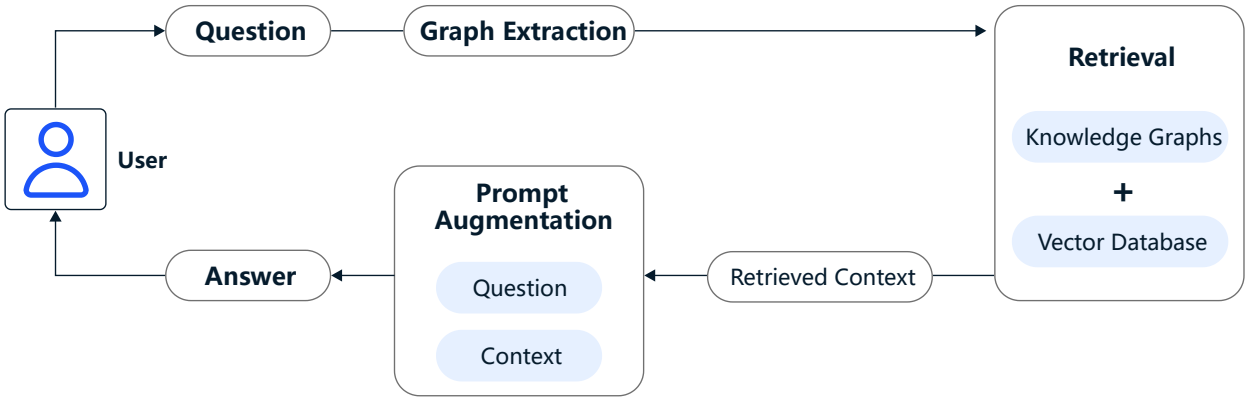
知识图谱在思维链问答中越来越受到重视，特别是与大语言模型（LLM）结合使用时。这种方法通过将复杂问题拆解为子问题，检索相关信息，并综合形成最终答案。知识图谱在这一过程中提供了结构化的信息，增强了 LLM 的推理能力。

举例来说，一个 LLM 代理可能会先使用知识图谱识别查询中的相关实体，然后从不同来源获取更多信息，最后生成一个全面的答案，这个答案反映了图谱中互联的知识。

知识图谱的实际应用

过去，知识图谱主要用于数据密集型领域，比如大数据分析和企业搜索系统，其作用是维护不同数据孤岛之间的一致性和统一性。但随着大语言模型驱动的 RAG 系统的发展，知识图谱找到了新的应用场景。现在，它们作为对概率性大语言模型的结构化补充，帮助减少虚假信息，提供更多上下文，并作为 AI 系统中的记忆和个性化机制。

介绍 GraphRAG



LeewayHertz

GraphRAG 是一种先进的检索方法，它将知识图谱与向量数据库结合在一个 RAG（检索增强生成）架构中。这种混合模型利用了两种系统的优势，提供更加准确、有背景信息且容易理解的 AI 解决方案。Gartner 已经指出 知识图谱在提升产品策略和创造新的 AI 应用场景方面的日益重要性。

GraphRAG 的特点包括：

1. **更高的准确性：** 通过结合结构化数据和非结构化数据，GraphRAG 能够提供更准确和全面的回答。
2. **可扩展性：** 这种方法简化了 RAG 应用的开发和维护，使其具有更好的可扩展性。
3. **可解释性：** GraphRAG 提供清晰的推理路径，增强了系统的透明度，使 AI 的输出结果更容易理解和信任。

GraphRAG 的优势

GraphRAG 相较于传统的 RAG 方法具有几个显著的优势：

1. **更高质量的回答：** 整合知识图谱后，GraphRAG 提高了 AI 生成回答的准确性和相关性，最近的基准测试显示准确性提高了三倍。
2. **成本效益：** GraphRAG 更具成本效益，所需的计算资源和训练数据较少，对于希望优化 AI 投资的组织来说是一个有吸引力的选择。
3. **更好的可扩展性：** 这种方法支持大规模 AI 应用，使组织能够轻松处理更复杂的查询和更大的数据集。
4. **改进的可解释性：** GraphRAG 的结构化方法提供了更清晰的推理路径，使 AI 决策过程更加透明，便于调试。
5. **揭示隐藏的关联：** 知识图谱能够揭示在大数据集中未被注意的关系，提供更深入的洞察，提升决策过程的质量。

常见的 GraphRAG 架构

几种 GraphRAG 架构作为将知识图谱有效整合到 RAG 系统中的方法正在出现：

1. **带有语义聚类的知识图谱：** 这种架构通过在生成回答前对相关信息进行聚类，来提高数据检索的相关性和准确性。
2. **知识图谱与向量数据库的集成：** 这种架构将两个系统结合起来，为大语言模型提供更丰富的背景，从而生成更全面和背景适宜的回答。
3. **知识图谱增强的问答系统：** 在这种架构中，知识图谱在向量检索后为大语言模型生成的答案添加事实信息，以确保回答的准确性和完整性。
4. **图谱增强的混合检索：** 这种方法结合了向量搜索、关键字搜索和图谱特定的查询，提供一个强大且灵活的检索系统，增强了大语言模型生成相关回答的能力。

GraphRAG 的新兴模式

随着 GraphRAG 的不断发展，几个新兴模式开始显现：

1. **查询增强：** 利用知识图谱优化和增强查询，确保检索到最相关的信息。
2. **答案增强：** 通过添加相关事实来提高大语言模型生成的回答的准确性和完整性。
3. **答案控制：** 利用知识图谱验证 AI 生成内容的准确性，减少错误或虚假信息的风险。

这些模式展示了 GraphRAG 如何改变 AI 系统处理复杂查询和生成回答的方式。

GraphRAG 的应用

1. **法律研究：** GraphRAG 能够在复杂的法律、判例和案例研究网络中导航，为法律专业人员提供强大的工具，帮助发现相关法律信息和潜在联系。
2. **医疗健康：** 在医疗领域，GraphRAG 帮助理解医学知识、患者历史和治疗选项中的复杂关系，从而提高诊断准确性和个性化治疗计划。
3. **金融分析：** GraphRAG 帮助分析复杂的金融网络和依赖关系，利用知识图谱中的互联数据提供市场趋势、风险管理和投资策略的洞察。
4. **社交网络分析：** GraphRAG 使得探索复杂的社会结构和互动变得可能，帮助研究人员和分析师理解社会网络中的关系和影响模式。
5. **知识管理：** GraphRAG 通过捕捉和利用组织内的关系和层级来提升企业知识库，提高决策过程，并促进企业内部的创新。

随着 AI 的进步，将知识图谱纳入检索增强生成系统变得越来越重要。知识图谱提供了一个组织和连接数据的强大框架，带来更精准、有背景的且易于解释的 AI 解决方案。GraphRAG 的出现展示了将知识图谱与传统向量方法结合的优势，提供了一种更全面、更高效的信息检索和回答生成的方法。

高级 RAG：通过多模态检索增强生成拓展视野

人工智能的进步一直伴随着突破性进展，这些进展不断扩展机器理解和生成的边界。传统的检索增强生成 (RAG) 系统主要关注文本数据，而多模态 RAG 的出现则标志着一个重要的技术飞跃。这种创新技术使得 AI 能够处理和整合多种数据形式——如文本、图像、音频和视频——以生成内容丰富且具有上下文的输出。通过利用多模态数据，这些先进的 AI 系统变得更加灵活、上下文敏感，并能提供更深入的见解和更准确的回应。本节将探讨多模态 RAG 的核心概念、操作机制以及潜在应用，突显其在下一代 AI 交互中的重要性。

理解多模态 RAG

多模态 RAG 是经典 RAG 框架的高级扩展，它将检索机制与多种数据类型的生成式 AI 结合起来。传统的 RAG 系统通过查询文本数据库获取信息，而多模态 RAG 则通过将文本、图像、音频和视频整合到检索和生成过程中，扩展了这一能力。这种扩展使 AI 模型能够利用更广泛的输入，从而生成更全面和更细致的结果。

多模态 RAG 如何运作?

多模态 RAG 的工作流程包括将不同类型的数据编码成结构化格式，通常是向量，这样 AI 模型就可以处理。这些向量存储在一个共享的嵌入空间中，其中包含来自不同模态的数据。当进行查询时，模型会从这些模态中检索相关信息，从而确保提供更加丰富和准确的回应。例如，对于一个关于历史事件的查询，系统可能会检索文本描述、相关图

像、专家评论的音频片段和视频镜头，这些信息共同构成一个更详细和信息丰富的回答。

实现多模态 RAG 的方法

实现多模态 RAG 可以采用多种方法，每种方法都有其独特的优点和挑战：

1. **单一多模态模型：**
   这种方法使用一个统一的模型，该模型被训练来将各种数据类型——如文本、图像、音频——编码到一个共同的向量空间中。模型然后可以无缝地在这些数据类型之间进行检索和生成。虽然这种方法通过使用单一模型简化了过程，但它需要复杂的训练，以确保准确地编码和检索多模态数据。
2. **基于文本的基础模态：**
   在这种方法中，非文本数据在编码和存储之前会被转换为文本描述。这种方法利用了当前最先进的文本模型的优势。然而，在转换过程中可能会有信息丢失，因为图像或音频中的细微差别可能在文本中无法完全体现。
3. **多个编码器：**
   这种方法使用不同的模型来编码不同的数据类型，每种数据类型由其专门的模型处理。检索过程中将整合这些结果。虽然这种方法允许更专业的编码和更准确的数据检索，但它增加了系统的复杂性，需要对多个模型及其相互作用进行细致的管理。

多模态 RAG 的架构

多模态 RAG 的架构建立在传统 RAG 的基础上，同时融合了处理多种数据类型的复杂性。核心架构包括以下几个关键组件：

1. **特定模态编码器：**
   每种数据模态，如文本、图像或音频，都由一个专门的编码器处理。这些编码器将原始数据转换为一个统一的嵌入空间，使所有模态能够以标准化的方式进行比较和检索。
2. **共享嵌入空间：**
   多模态 RAG 的一个关键组件是共享嵌入空间，其中存储了来自不同模态的编码向量。这个空间允许进行跨模态的比较和检索，使模型能够识别不同数据类型中的相关信息。
3. **检索器：**
   检索器组件负责查询共享嵌入空间，以找到跨模态的最相关数据点。它可以根据各种标准检索信息，如与输入查询的相关性或与空间内其他数据点的相似性。
4. **生成器：**
   一旦检索到相关信息，生成器组件会将这些数据整合到 AI 模型的回应中。生成器

通常是一个复杂的语言模型，能够将来自多个模态的见解编织成连贯且上下文准确的输出。

5. **融合机制：**
   融合机制负责将检索到的多模态数据合并成一个统一的表示，以供生成器使用。这个过程可能涉及选择最相关的模态或从不同来源合成信息，以创建一个全面的回应。

在 RAG 系统中管理不同模态的信息需要采用几个关键策略：

1. **统一嵌入空间：**
   将所有数据类型编码到一个共同的嵌入空间中，使系统能够高效地执行跨模态的检索操作。同时，这个嵌入空间也为不同来源的数据整合和对齐提供了基础。
2. **跨模态注意力机制：**
   使用跨模态注意力机制，确保模型能够集中关注检索数据中最相关的信息，不论这些信息来自哪个模态。这有助于在最终的响应中平衡各类数据的重要性。
3. **模态特定的后处理：**
   在检索完成后，可能需要对每种模态的数据进行特定的后处理，例如调整图像大小或规范化音频，以确保数据在整合和生成时达到最佳状态。

多模态 RAG 在聊天机器人中的应用

多模态 RAG 大大提升了聊天机器人的功能，使它们能够提供更加丰富和符合上下文的互动体验。传统聊天机器人主要依靠文本，这限制了它们对涉及视觉或听觉信息的响应能力。而多模态 RAG 使聊天机器人能够从图像、视频和音频片段中获取和整合信息，从而提供更全面和有趣的用户体验。

例如，使用多模态 RAG 的 客户支持聊天机器人 可以根据用户的查询展示教学视频、产品图片或音频指南，使用户获得更互动、更实用的帮助。这在零售、医疗和教育等领域尤为重要，因为这些领域的沟通往往需要多种信息形式的支持。

多模态 RAG 的应用拓展

多模态功能的引入正在为各行各业开辟新的机会：

- **医疗保健：**
  多模态 RAG 可以将文本医疗记录、放射学图像、实验室结果以及患者的音频描述结合起来，从而提高诊断系统的准确性和全面性。
- **金融：**
  在金融服务中，多模态 RAG 能够处理和分析包含表格、图表及解释文本的复杂文档，从而改善决策过程。

- **教育：**
  教育平台可以利用多模态 RAG 将文本、视频讲座、图示和互动模拟融合成一个完整的教学故事，提供更丰富的学习体验。

多模态 RAG 是一种重要的技术进步，它可能会改变 AI 系统与用户的互动和响应方式。通过将多种数据类型融入到检索和生成过程中，多模态 RAG 系统能够提供更丰富、更准确且更符合上下文的输出，从而在各个行业中开拓新的可能性。随着技术的发展，预计其应用将会扩展，进一步提升 AI 处理复杂多模态信息的能力。

LeewayHertz 的 GenAI 协调平台 ZBrain 如何在先进的 RAG 系统中脱颖而出？

对先进的 RAG、多模态 RAG 和知识图谱感到好奇？想象一下将这些强大功能整合到一个平台上，让你可以轻松构建高级 AI 应用。那就是 ZBrain。

ZBrain，由 LeewayHertz 开发，是一个全面的协调平台，旨在简化和加速企业级 AI 解决方案的开发和扩展。通过其用户友好的低代码环境，ZBrain 使组织能够快速创建、部署和扩展定制的生成式 AI (GenAI) 应用，减少了编码工作量。这个平台通过让企业利用自己的数据来构建高度定制和准确的 AI 应用，彻底改变了企业 AI 开发的流程。作为一个中央控制中心，ZBrain 能够无缝集成现有的技术栈，提高 GenAI 应用开发的效率。基于 ZBrain 构建的应用在自然语言处理 (NLP) 任务中表现出色，例如报告生成、翻译、数据分析、文本分类和总结。通过利用私人和上下文数据，ZBrain 确保响应高度相关且个性化，以满足特定的业务需求。

如何与先进的检索增强生成 (RAG) 系统对接

- **整合多样化的数据源：** ZBrain 整合了包括私人、公共和实时流在内的各种数据源，覆盖所有数据格式（结构化、半结构化和非结构化），提高了 AI 响应的准确性和相关性。
- **块级优化：** 该平台通过将信息拆分为易于处理的块，并应用最有效的检索策略，确保生成精确且量身定制的输出。
- **自动发现检索策略：** ZBrain 中的先进算法自动识别并应用最佳检索策略，根据数据和上下文减少手动操作，提高了数据检索的准确性。
- **防护措施和幻觉控制：** ZBrain 配备了防护措施和幻觉控制功能，以防止生成不准确或误导性的信息，确保高精度和可靠性。

多模态能力

- **处理多种数据格式:** ZBrain 在处理文本、图像、视频和音频等多种数据格式方面表现优异，可以提供全面而细致的响应。
- **跨数据类型的集成与分析:** 该平台能够整合和分析各种类型的数据，提供更丰富的见解和相关的回答。
- **改进的查询处理:** ZBrain 高效地管理和检索多种数据模态的信息，提高了对复杂问题的准确性和洞察力。

知识图谱

- **结构化数据框架:** ZBrain 将数据组织成一个结构化的网络，通过将相关概念连接起来，提高了检索的准确性，并提供了更深入的见解。
- **深入的数据洞察:** 知识图谱的互联特性使 ZBrain 能够提供细致入微的、具有上下文感知的回答，带来更加丰富和有意义的见解。
- **扩展的数据能力:** ZBrain 支持在块或文件级别扩展数据，更新元信息，并生成本体，从而改善数据的表示、组织和检索。

在企业 AI 解决方案开发中使用 ZBrain 的好处

ZBrain 为企业 AI 解决方案开发提供了许多优势，包括：

- **可扩展性**
  ZBrain 能够轻松扩展 AI 解决方案，处理不断增长的数据量和使用场景，且不影响性能。
- **高效集成**
  该平台可以轻松与现有技术栈集成，缩短部署时间和降低成本，加快 AI 的应用。
- **定制化**
  ZBrain 支持开发高度定制的 AI 应用程序，满足特定业务需求，与组织目标一致。
- **资源效率**
  其低代码特性减少了对大量开发人员的需求，适合技术团队规模较小的组织。
- **全面解决方案**
  从开发到部署，ZBrain 涵盖了 AI 应用程序的整个生命周期，是一个全面的解决方案。
- **云中立部署**
  ZBrain 是云中立的，允许应用程序在各种云平台上部署，提供灵活性以适应不同的组织需求和基础设施偏好。

ZBrain 的先进 RAG 系统、多模态支持和强大的知识图谱集成使其成为一个强大的平台，在广泛的应用中提供了更高的准确性、效率和洞察力。

附注

检索增强生成 (RAG) 的进步大幅提升了其能力，使其克服了之前的限制，并在 AI 驱动的信息检索和生成中开辟了新潜力。通过使用复杂的检索机制，先进的 RAG 可以访问大量数据，确保生成的回答既准确又富有相关的上下文。这一进步为更动态和互动的 AI 应用铺平了道路，使 RAG 成为客户服务、研究、知识管理和内容创作等领域的重要工具。这些先进的 RAG 技术的应用为企业提供了提升用户体验、简化流程和以更高准确性和效率解决复杂问题的机会。

多模态 RAG 和知识图谱 RAG 的引入进一步提升了这一框架的能力，推动了在各行业的广泛应用。多模态 RAG 结合了文本、视觉和其他数据形式，使大语言模型 (LLM) 能够生成更加全面和具有上下文感知的回应，从而改善用户体验，提供更丰富和细致的信息。而知识图谱 RAG 利用互联的数据结构来检索和生成语义丰富的内容，大大提高了信息的准确性和深度。这些 RAG 技术的进步预示着 AI 创新的新一波浪潮，为复杂的信息检索挑战提供了更智能和灵活的解决方案。