

要不断地从书架上取书再放回去，这会导致你的效率较低。同样的道理，如果显存容量不够，训练和推理速度也会大大降低。

② 卡间通信的瓶颈。

当我们使用多张 GPU 卡进行分布式训练时，这些卡之间需要交换数据。卡间通信速度慢会影响整体的训练效率。NVIDIA 的 NVLink 技术能够提高 GPU 之间的数据传输速率，但即使有了这种技术，数据传输仍然需要优化。就像在篮球比赛中，如果队员之间传球不够快、配合不够默契，整个团队的表现都会受到影响。NVLink 技术就像一个高速通道，能够让数据在不同的 GPU 之间快速传递，但我们还需要优化数据传输的方式，以进一步提高效率。

③ 数据传输和存算一体。

CPU 和 GPU 在处理数据时，一个主要区别是数据的传输和存储方式。CPU 的数据存储在内存中，通过缓存进行快速访问，但整体的数据传输速率受限于内存总线的带宽。GPU 的数据存储在显存中，直接与计算单元连接。虽然显存容量较小，但数据传输速率非常快。这就像在家里，CPU 是一个大书柜，书很多但取用速度慢；GPU 是一个小书架，书少但取用速度快。然而，当显存容量不足时，GPU 也会面临数据交换的瓶颈。

④ 优化计算框架。

为解决硬件限制，研究者开发了多种计算框架和工程方法，如 DeepSpeed。这些框架考虑了 GPU 型号、显存大小、核心数等硬件参数，旨在提高硬件利用效率。同时，一些优化算法如 Flash Attention 也被开发出来，以提高计算效率。

在 2023 年上半年，微调技术曾一度备受青睐，但实际应用表明其效果并不稳定，且存在多种潜在问题。随后，有人提出了一个新的解决方案——检索增强生成（RAG），这项技术引起人们广泛关注。



5.4.3 RAG 技术

1. RAG 概述

RAG 是一种结合了向量语义搜索和大语言模型的技术，旨在提高智能问答系统回答的准确性和可靠性，降低模型“幻觉”，并增加特定领域知识或实时信息。典型的应用实例如 ChatPDF 和 ChatDOC。在这些应用中，用户可以上传 PDF 文档或 DOC 文档，系统会将文档分块并向量化存储，当用户提出问题时，系统实时检索相关的文档内容，并将其作为背景知识提供给模型。这使模型能够根据特定文档内容生成回答，提高回答的针对性和准确性。

RAG 的核心在于通过检索外部知识库的相关信息，来补充预训练大语言模型在生成回答时的知识缺口，有效解决知识幻觉和专有知识缺乏的问题。知识幻觉指模型生成的回答看似合理但实际上不准确或缺乏事实依据；专有知识缺乏是指模型无法回答涉及特定领域或组织内部专有信息的问题。RAG 通过在生成过程中引入相关的检索结果作为上下文，使模型能够基于更丰富、更准确的信息生成回答，从而提高回答的准确性和可靠性。

2. RAG 的工作原理

RAG 的工作流程包括两个主要阶段：数据准备阶段和检索生成阶段。

(1) 数据准备阶段

如图 5.8 所示，数据准备阶段涉及两个步骤：数据处理和向量化。

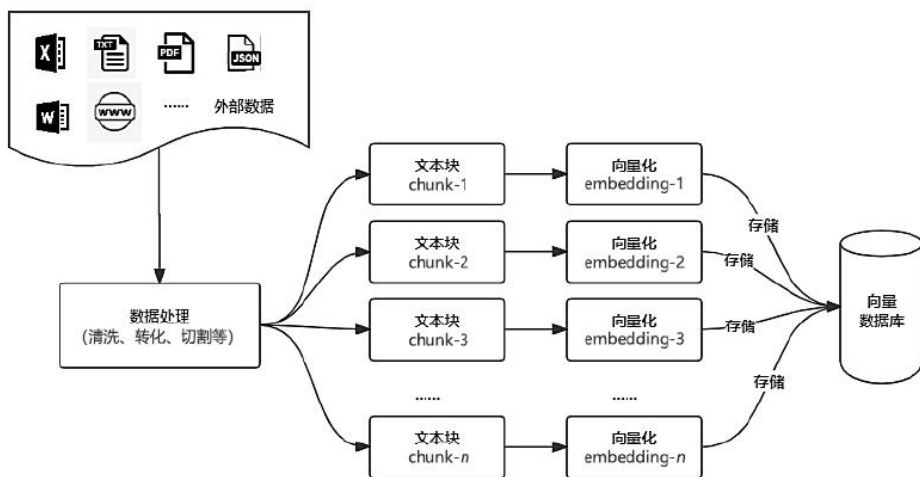


图 5.8 RAG 的数据准备阶段

① 数据处理。

外部数据比如各种类型的文件、网页等数据，先经过清洗、转化、切割，再被分成若干小块 (chunk)。每个 chunk 代表数据中的一个知识点或段落。分块策略可以基于字数、语义、标点符号等。例如，可以按照每 500 个字符分块，或按照语义相关性分块。这一步的目的是将庞大的数据内容细化为易于管理和检索的小单元。

② 向量化。

向量化的目的是将数据内容转化为可语义计算的形式，以便于后续的搜索和匹配。每个 chunk 通过向量化转换为文本向量，最终存储在向量数据库中。

(2) 检索生成阶段

RAG 的检索生成阶段如图 5.9 所示。检索生成阶段涉及两个步骤：信息检索和文本生成。

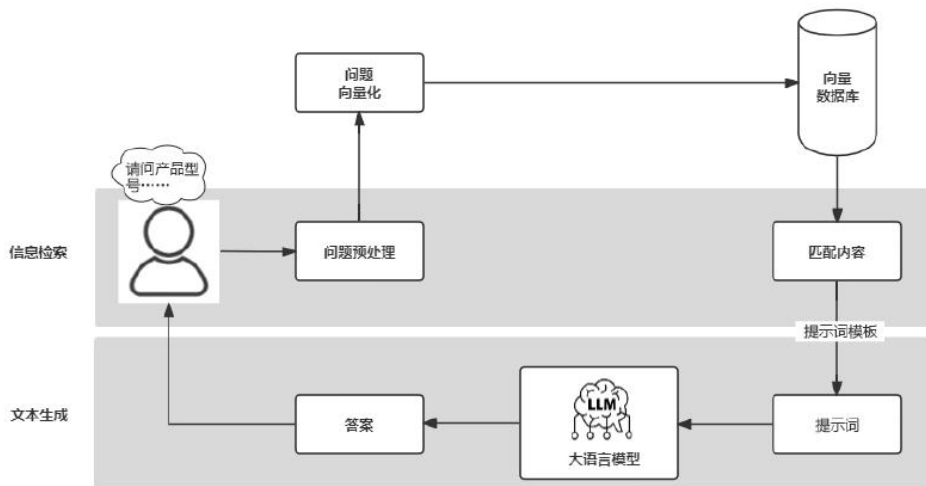


图 5.9 RAG 的检索生成阶段

① 信息检索。

系统接收用户问题并进行预处理，如文本清洗、标准化等，确保输入数据的质量。接着，系统对问题进行向量化，即将问题转换为向量表示。接下来，系统在之前构造好的向量数据库里，利用向量语义搜索技术，根据问题的语义特征找到与之最相关的内容，即匹配内容。

② 文本生成。

系统将检索到的匹配内容和预制的提示词结合形成新的提示词。这个提示词通常包括问题或任务描述、检索到的相关信息、原始问题，以及对答案格式和处理不确定性的指导。示例如下：

已知检索的信息：

{context(匹配内容)}

根据上述已知信息，专业地回答用户的问题。如果无法从中得到答案，请说“根据已知信息无法回答该问题”或“没有提供足够的相关信息”，不允许在答案中添加编造成分，答案请使用中文。

问题是：{question(问题)}

构建好的提示词随后被输入大语言模型中，大语言模型基于提供的上下文和指令，生成答案，返回给用户。

3. RAG 的优势和局限性

RAG 的主要优势如下。

- (1) 准确性好：通过引入外部知识，减少知识幻觉。
- (2) 专业性好：能够处理特定领域的专业问题。
- (3) 实时性好：可以利用最新更新的知识库内容。
- (4) 灵活性好：适应各种类型的查询，包括开放性问题。

然而，RAG 也存在以下局限性。

- (1) 计算资源需求高：实时检索和知识整合需要较高的计算资源。
- (2) 存在知识库质量依赖：系统性能在很大程度上取决于知识库的质量和全面性。
- (3) 存在潜在的检索偏差：检索结果可能不完全匹配用户意图，影响回答质量。

4. RAG 的适用场合

RAG 特别适用于以下场合。

(1) 动态知识环境：在需要频繁更新知识库或处理最新信息的场景中，RAG 表现出色，如新闻分析、市场趋势预测等领域。

(2) 开放域问答：当系统需要回答广泛且不可预测的问题时，RAG 能够灵活地检索和整合相关信息。

(3) 专业领域应用：在医疗、法律、金融等专业领域，RAG 可以有效结合专业知识库和语言模型，提供准确的专业回答。

(4) 大规模信息处理：对于需要从海量文档中快速提取信息的场景，如企业知识管理、学术研究等，RAG 能够显著提高效率。

(5) 个性化服务：在需要根据用户背景或历史交互提供定制化回答的应用中，RAG 可以有效整合用户相关信息。

5. RAG 的实际应用场景

RAG 在多个领域有广泛应用。除了前面提到的 ChatPDF、ChatDOC 这类文档问答系统，RAG 还有以下应用实例。

(1) 客户服务系统：帮助客服人员快速检索产品信息，提供准确的客户支持。

(2) 医疗诊断辅助：通过检索最新的医学文献和病例，辅助医生制订治疗方案。

(3) 法律咨询系统：协助律师快速检索相关法律条文、判例和解释，并提供客观的法律建议。系统能够根据用户的具体问题，从大量法律文献中检索相关信息，并结合大语言模型的理解能力，生成符合法律专业要求的回答。

(4) 企业知识管理系统：有效整合和利用企业内部的庞大知识库。例如，员工可以通过系统快速检索公司政策、操作手册、项目文档和最佳实践等信息。当员工提出关于公司流程、技术规范或项目历史的问题时，系统能够从企业知识库中检索相关信息，并生成简洁明了的回答。这不仅提高了信息获取的效率，还能确保企业知识的一致性传播，支持决策制订和问题解决。

(5) 科研文献助手：帮助研究人员快速定位和综合大量学术文献中的关键信息。研究人员可以提出具体的研究问题，系统会从最新的学术论文、研究报告中检索相关内容，并生成简明扼要的总结或分析。

5.5 模型评估

随着大语言模型的发展，我们常常疑惑：这么多模型，如何评估哪一个更好？

5.5.1 模型评估的挑战

目前，针对大语言模型的评估方法还不够完善，尤其在生成式人工智能领域。现有的评估方法主要分为以下 3 类，它们各有其优缺点。

(1) 客观题测试

这是传统机器学习中的评估方法，模型根据一系列预设的题目或数据集进行测试。这种方法的优势在于自动化程度高，但问题在于模型可能在训练过程中已经接触过这些测试数据，导致“刷题”现象，即模型并非真正理解问题，而是通过记忆相似的输入输出进行推断，从而影响评估的公正性。

(2) 人工测评

评估人员通过人为交互测试模型的输出。这种方法贴近实际应用场景，能够捕捉更丰富的模型表现，但它的局限在于覆盖面有限，评估标准主观，不同测评者可能对同一模型的表现有不同的理解和评价，因此在一致性上存在问题。

(3) AI 裁判评估

让其他 AI 模型（如 GPT-4）作为裁判，来评估模型的表现。这种方法的使用比较少见，但它有潜力实现更加一致和快速的评估。不过，这类评估方法也存在模型本身的偏见和限制，可能会影响评估结果的客观性。

总的来说，无论采用哪种评估方法，都存在局限性。评估结果通常只能作为参考，而不是绝对标准。为了获得更准确的评估结果，往往需要结合多种评测方法，并根据具体应用场景进行综合判断。