深入理解RAG中的嵌入模型Embedding Model

小喵学AI 2025年03月05日 11:50 北京

小喵学AI
专注于分享C++/Python编程、计算机视觉、自然语言处理、大模型等深度学习与人工智...
153篇原创内容

公众号

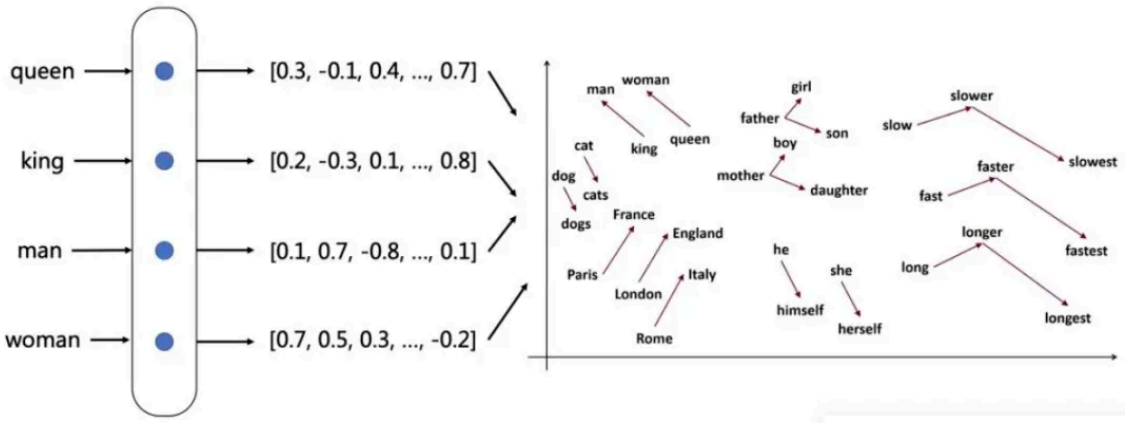
在当前流行的RAG引擎（例如RAGFlow、Qanything、Dify、FastGPT等）中，嵌入模型（Embedding Model）是必不可少的关键组件。在RAG引擎中究竟扮演着怎样的角色呢？本文笔者进行了总结，与大家分享~

什么是Embedding?

在学习嵌入模型之前，我们需要先了解什么是Embedding。简单来说，Embedding是一种将离散的非结构化数据（如文本中的单词、句子或文档）转换为连续向量的技术。

在自然语言处理（NLP）领域，Embedding通常用于将文本映射为固定长度的实数向量，以便计算机能够更好地处理和理解这些数据。每个单词或句子都可以用一个包含其语义信息的向量来表示。

Embedding常用于将文本数据映射为固定长度的实数向量，从而使计算机能够更好地处理和理解这些数据。每个单词或句子都可以用一个包含其语义信息的实数向量来表示。



以“人骑自动车”为例，在计算机中，单词是以文字形式存在的，但计算机无法直接理解这些内容。Embedding的作用就是将每个单词转化为向量，例如：

- “人”可以表示为 [0.2, 0.3, 0.4]
- “骑”可以表示为 [0.5, 0.6, 0.7]
- “自行车”可以表示为 [0.8, 0.9, 1.0]

通过这些向量，计算机可以执行各种计算，比如分析“人”和“自行车”之间的关系，或者判断“骑”这个动作与两者之间的关联性。

此外，Embedding还可以帮助计算机更好地处理和理解自然语言中的复杂关系。例如：

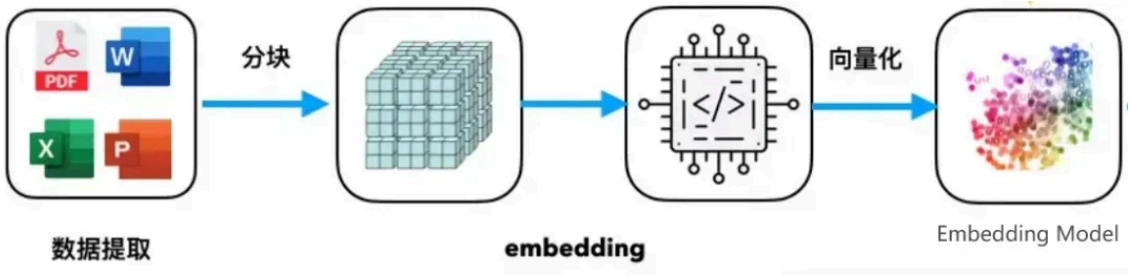
- 相似的词（如“人”和“骑手”）在向量空间中会比较接近。
- 不相似的词（如“人”和“汽车”）则会比较远。

「那么为什么需要向量呢？」

因为计算机只能处理数字，无法直接理解文字。通过将文本转换为向量，相当于为数据分配了一个数学空间中的“地址”，使计算机能够更高效地理解和分析数据。

什么是Embedding Model?

在自然语言处理（NLP）中，嵌入模型（Embedding Model） 是一种将词语、句子或文档转换成数字向量的技术。它通过将高维、离散的输入数据（如文本、图像、声音等）映射到低维、连续的向量空间中，使得计算机能够更好地理解和处理这些数据。



Embedding Model就像是给每个词或句子分配一个独特的“指纹”，这个“指纹”能够在数学空间中表示这个词或句子的含义。具体来说，这种模型将每个词语或句子转换成一个固定长度的数字向量。通过这种方式，计算机可以对文本进行各种数学计算，例如：

- 比较词语的相似性：通过计算两个词语向量之间的距离（如余弦相似度），可以判断它们在语义上的相似程度。
- 分析句子的意义：通过对句子中的所有词语向量进行聚合（如平均值或加权和），可以得到整个句子的向量表示，并进一步分析其语义信息。

这种技术在许多NLP任务中具有重要意义，以下是几个典型的应用示例：

- 语义搜索：通过计算查询向量与文档库中各文档向量的相似度，找到与查询最相关的文档或段落。例如，用户输入“如何制作披萨？”，系统会返回最相关的烹饪指南。
- 情感分析：判断一段文本的情感倾向（如正面、负面或中性）。例如，对于一篇产品评论“这款手机性能出色，但电池续航一般”，系统可以分析出该评论整体上是正面的，但也存在一些负面因素。
- 机器翻译：将一种语言的文本转换为另一种语言。例如，用户输入“我喜欢猫”，系统将其转换为对应的英文翻译“I like cats”。

- 问答系统：根据用户的问题，从知识库中检索相关信息并生成回答。例如，用户提问“太阳有多大？”，系统通过嵌入模型找到相关天文学文档，并生成详细的回答。
- 文本分类：将文本归类到预定义的类别中。例如，新闻文章可以被自动分类为政治、体育、科技等不同类别，基于其内容的向量表示。
- 命名实体识别（NER）：识别文本中的特定实体（如人名、地名、组织名等）。例如，在一段文字“李华在北京大学学习”中，系统可以识别出“李华”是人名，“北京大学”是组织名。

Embedding Model的作用

在RAG引擎中，嵌入模型（Embedding Model）扮演着至关重要的角色。它用于将文本转换为向量表示，以便进行高效的信息检索和文本生成。以下是Embedding Model在RAG引擎中的具体作用和示例：

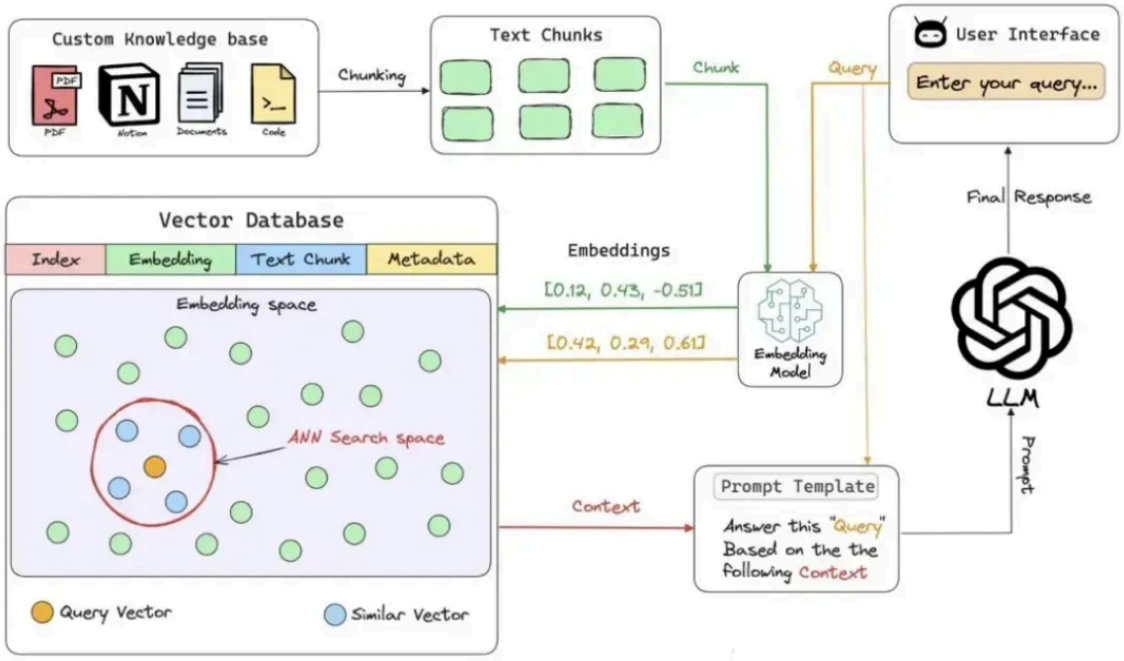
1. 文本向量化
   - 作用：将用户的问题和大规模文档库中的文本转换为向量表示。
   - 举例：在RAG引擎中，用户输入一个问题，如“如何制作意大利面？”，Embedding Model会将这个问题转换为一个高维向量。
2. 信息检索
   - 作用：使用用户的查询向量在文档库的向量表示中检索最相似的文档。
   - 举例：RAG引擎会计算用户问题向量与文档库中每个文档向量的相似度，然后返回最相关的文档，这些文档可能包含制作意大利面的步骤。
3. 上下文融合
   - 作用：将检索到的文档与用户的问题结合，形成一个新的上下文，用于生成回答。
   - 举例：检索到的关于意大利面的文档会被Embedding Model转换为向量，并与问题向量一起作为上下文输入到生成模型中。
4. 生成回答
   - 作用：利用融合了检索文档的上下文，生成模型生成一个连贯、准确的回答。
   - 举例：RAG引擎结合用户的问题和检索到的文档，生成一个详细的意大利面制作指南作为回答。
5. 优化检索质量
   - 作用：通过微调Embedding Model，提高检索的相关性和准确性。
   - 举例：如果RAG引擎在特定领域（如医学或法律）中使用，可以通过领域特定的数据对Embedding模型进行微调，以提高检索的质量。
6. 多语言支持
   - 作用：在多语言环境中，Embedding Model可以处理和理解不同语言的文本。

- 举例：如果用户用中文提问，而文档库包含英语内容，Embedding Model需要能够处理两种语言的文本，并将它们转换为统一的向量空间，以便进行有效的检索。

7. 处理长文本

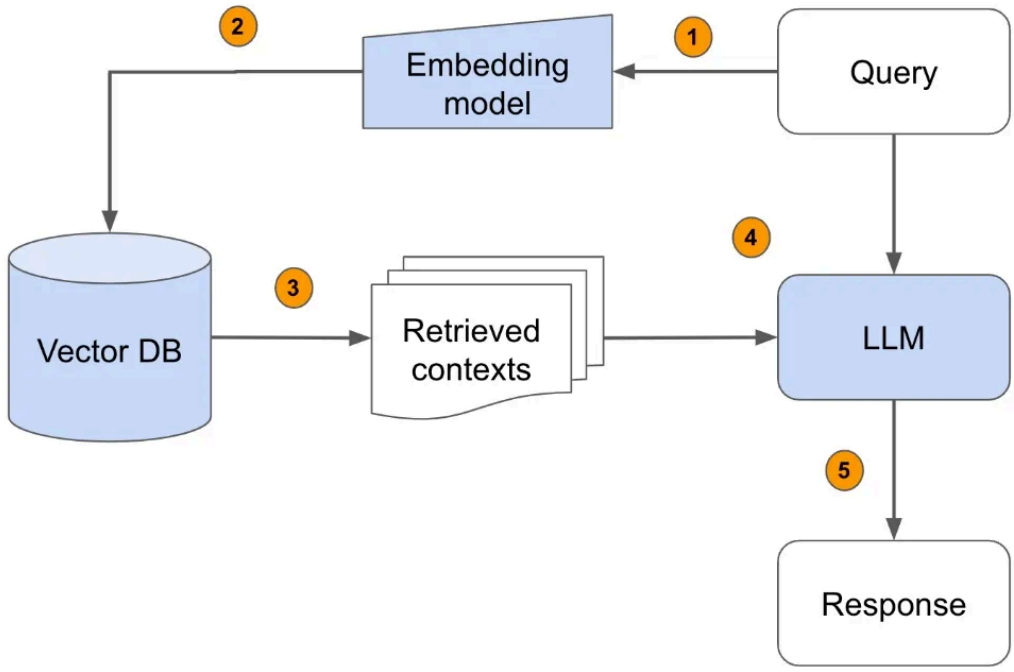
- 作用：将长文本分割成多个片段，并为每个片段生成Embedding，以便在RAG引擎中进行检索。
- 举例：对于长篇文章或报告，Embedding Model可以将其分割成多个部分，每个部分都生成一个向量，这样可以在不损失太多语义信息的情况下提高检索效率。

通过以上几点，Embedding Model在RAG引擎中提供了一个桥梁，连接了用户查询和大量文本数据，使得信息检索和文本生成成为可能。如下图所示，Embedding Model正处于整个RAG系统的中心位置。



RAG引擎中的工作流

以下是一个RAG引擎中工作流的示意图，此流程基本与各大RAG引擎相匹配。虽然各个不同的RAG引擎内部算法可能有所区别，但整体工作流程大同小异。



「流程说明」

1. 查询嵌入化：将用户输入的查询传递给嵌入模型，并在语义上将查询内容表示为嵌入的查询向量。
2. 向量数据库查询：将嵌入式查询向量传递给向量数据库。
3. 检索相关上下文：检索前k个相关上下文——通过计算查询嵌入和知识库中所有嵌入块之间的距离（如余弦相似度）来衡量检索结果。
4. 上下文融合：将查询文本和检索到的上下文文本传递给对话大模型（LLM）。
5. 生成回答：LLM 将使用提供的内容生成回答内容。

RAG 3　　大模型 29

RAG · 目录

上一篇
一文读懂大模型RAG：检索、增强与生成的技术详解

下一篇
RAG检索增强之Reranker重排序模型详解！