# OpenAI 平台和模型介绍

**OpenAI**简介

对于初学者和专业的开发者来说，理解OpenAI API的强大功能以及如何利用它构建LLM应用是至关重要的。OpenAI API为我们提供了一种直观的方法来处理涉及自然语言、代码、图像的任务，而无需深入了解底层机制。

应用范围广泛。OpenAI API不仅可以处理自然语言任务，还可以生成和编辑图像，将语音转换为文本。这意味着，从语义搜索到内容生成，再到分类任务，都可以通过这一API来实现。

模型的多样性。OpenAI提供了多种模型，每种模型都有其特定的功能和价格。这为开发者提供了选择的灵活性，以确保他们为特定的LLM应用找到最合适的模型。

微调与定制。除了预训练的模型，OpenAI还为开发者提供了微调自定义模型的能力，这意味着开发者可以根据具体的需求和数据来优化模型。

结合OpenAI API，LLM应用开发者可以更轻松地处理各种任务，从简单的文本生成到复杂的语义搜索。对于希望在这个领域取得突破的开发者来说，OpenAI API无疑是一个宝贵的资源。

OpenAI**的主要应用场景**

对于希望利用现代技术为其LLM应用增添动力的开发者来说，了解OpenAI的主要应用场景是非常有益的。以下列出了OpenAI在LLM应用开发中的几个主要用途：

**1. 内容生成**  
开发者可以使用OpenAI生成高质量的文本内容，从简单的句子到完整的文章。这对于那些希望自动化内容生产或生成特定格式文本的LLM应用尤其有用。

**2. 摘要**  
OpenAI能够从大量文本中提取关键信息并生成简洁的摘要。这对于需要快速理解文档主旨的应用，如新闻摘要或研究论文摘要生成等场景，具有巨大的价值。

**3. 分类、归类和情感分析**  
OpenAI可以帮助开发者对文本进行分类或归类，并对文本中的情感进行分析。这在社交媒体分析、评论系统或任何需要对文本进行情感判定的LLM应用中都是非常有用的。

**4. 数据提取**  
OpenAI可以从非结构化数据中提取关键信息，为开发者提供有价值的数据点。这可以应用于票据扫描、合同审查或任何需要从文本中提取特定信息的LLM应用。

**5. 翻译**  
OpenAI不仅可以理解文本，还可以将其翻译成其他语言。这为开发多语言LLM应用或需要快速翻译功能的项目提供了强大的支持。

窗体顶端

OpenAI核心概念解析

在深入研究如何使用OpenAI为LLM应用带来价值之前，了解其核心概念是至关重要的。以下为OpenAI中的一些核心概念，并解释了它们在LLM应用开发中的重要性。

**1. GPT模型**  
OpenAI的GPT（generative pre-trained transformer）模型经过训练，可以理解自然语言和代码。GPT模型根据输入提供文本输出，这些输入也被称为"prompts"。通过设计提示，开发者可以“编程”GPT模型，通常是通过提供任务的说明或示例。GPT模型适用于各种任务，包括内容或代码生成、摘要、对话、创意写作等。

**2. 嵌入向量（Embeddings）**  
嵌入是数据（例如文本）的向量表示，旨在保留其内容和/或含义的某些方面。相似的数据块在某种程度上会有更接近的嵌入，而不相关的数据则相反。OpenAI提供的文本嵌入模型接受文本字符串作为输入，并输出一个嵌入向量。嵌入对于搜索、聚类、推荐、异常检测、分类等都很有用。

**3. 标记（Tokens）**  
GPT和嵌入模型使用称为标记的文本块来处理文本。标记代表常见的字符序列。例如，字符串"tokenization"被分解为"token"和"ization"，而像"the"这样的短且常见的词被表示为一个标记。在句子中，每个词的第一个标记通常以一个空格字符开始。作为一个粗略的经验法则，对于英文文本，1个标记大约等于4个字符或0.75个词。

OpenAI工作流程探究

当开发者决定利用OpenAI为LLM应用增添功能时，首先需要了解其工作流程。本章节将详细探讨如何从头开始构建一个OpenAI应用。

**1. 应用构建准备**。首先，开发者需要为应用创建一个基础。例如，如果使用Python Flask框架，则需要准备相应的环境。当环境准备就绪后，可以下载官方提供的代码样本，如：

git clone https://github.com/openai/openai-quickstart-node.git

或者直接从官方链接下载压缩包。

**2. API密钥配置**。为了确保应用能够与OpenAI API进行通信，开发者需要一个API密钥。这可以通过在OpenAI官方网站上注册并获取。获取到的密钥需要被添加到应用中，确保数据交互的安全性。

**3. 应用运行**

一旦配置完毕，可以通过以下命令安装依赖并运行应用：

npm install

npm run dev

随后，开发者可以在浏览器中访问应用并进行测试。

**4. 代码深入解读**。真正理解应用的核心是理解其背后的代码。例如，在generate.js文件中，有一个用于生成提示的函数，这是与GPT模型交互的关键部分。此函数根据用户输入的动物类型动态生成提示。

function generatePrompt(animal) {

const capitalizedAnimal = animal[0].toUpperCase() + animal.slice(1).toLowerCase();

return `Suggest three names for an animal that is a superhero.

Animal: Cat

Names: Captain Sharpclaw, Agent Fluffball, The Incredible Feline

Animal: Dog

Names: Ruff the Protector, Wonder Canine, Sir Barks-a-Lot

Animal: ${capitalizedAnimal}

Names:`;

}

此外，代码中还有一个部分专门用于与OpenAI API进行交互，发送请求并获取响应。这部分使用了completions端点，并设置了特定的参数，如温度为0.6。

const completion = await openai.createCompletion({

model: "text-davinci-003",

prompt: generatePrompt(req.body.animal),

temperature: 0.6,

});

通过深入了解这些核心代码部分，开发者可以更好地理解如何为LLM应用定制OpenAI功能。

OpenAI 的模型型号解析

当开发者决定在LLM应用中集成OpenAI时，了解其提供的不同模型是至关重要的。每种模型都有其独特的功能和应用场景。以下是OpenAI的各种模型的详细介绍：

**1. GPT-4：**这是OpenAI的最新模型系列，它在理解和生成自然语言或代码方面相较于GPT-3.5有所改进。对于希望获得高质量文本或代码输出的LLM应用开发者，GPT-4是一个理想的选择。

**2. GPT-3.5：**作为GPT-3的改进版本，这一系列模型继续在自然语言处理和代码生成方面展现出卓越的性能。

**3. DALL·E：**这是一个独特的模型，可以根据自然语言提示生成和编辑图像。对于需要图像生成功能的LLM应用，DALL·E无疑是一个强大的工具。

**4. Whisper：**Whisper模型专门用于将音频转化为文本。对于需要语音识别功能的LLM应用，Whisper是一个不可或缺的资源。

**5. Embeddings：**这是一组模型，可以将文本转化为数值形式，为进一步的文本分析和处理提供了有力的支持。

**6. Moderation：**这是一个经过微调的模型，能够检测文本是否可能包含敏感或不安全的内容，从而保证LLM应用的内容安全。

**OpenAI 的先锋模型探析**

当开发者决定在LLM应用中采用OpenAI技术时，了解OpenAI的模型更新策略和模型版本是至关重要的。这能确保开发者始终能够获得最先进、最高效的自然语言处理技术。

**1. 持续更新的模型：**  
随着gpt-3.5-turbo的发布，OpenAI开始实施一种持续的模型更新策略。例如，模型名称为gpt-3.5-turbo、gpt-4和gpt-4-32k的模型会指向最新的版本。为了确认具体使用的模型版本，开发者可以查看发送ChatCompletion请求后的响应对象。这将明确显示所用的模型版本，如gpt-3.5-turbo-0613。

**2. 静态模型版本：**  
尽管OpenAI不断推出更新的模型，但它还提供静态模型版本供开发者使用。即使推出了新版本，这些静态版本至少还可以继续使用三个月。

**3. 为模型改进提供贡献：**  
随着模型更新的加快，OpenAI鼓励社区为不同的用例贡献评估，以帮助改进模型。对此感兴趣的开发者可以查看OpenAI Evals存储库，参与模型的持续完善。

**4. 临时快照模型：**  
以下列出的模型是暂时的版本快照。一旦有了更新版本，OpenAI将宣布它们的停用日期。如果开发者希望始终使用最新的模型版本，只需使用标准的模型名称，如gpt-4或gpt-3.5-turbo。

**gpt-3.5-turbo-0301**：预计停用日期为2024年6月13日，替代模型为gpt-3.5-turbo-0613。

**gpt-4-0314**：预计停用日期为2024年6月13日，替代模型为gpt-4-0613。

**gpt-4-32k-0314**：预计停用日期为2024年6月13日，替代模型为gpt-4-32k-0613。

更多关于模型停用的信息，开发者可以参考OpenAI的官方停用页面。

在构建LLM应用时，确保跟随OpenAI的模型更新步伐是至关重要的，这将确保应用在自然语言处理领域保持前沿。

# Claude 2 模型介绍

对于那些寻求构建高效LLM应用的开发者，了解最新的模型和技术是至关重要的。Anthropic发布的Claude 2模型，一款被广大开发者高度期待的模型。

Claude 2是什么？

Claude 2是Anthropic推出的新模型。这款模型在多方面都实现了显著的进步，包括编码、数学和推理能力。事实上，Claude 2在Bar考试的多项选择部分得分为76.5%，这比Claude 1.3的73.0%有所提高。在GRE的阅读和写作部分，Claude 2的得分超过了90%的应试者。

Claude 2的三大特点：

1. 处理大量数据的能力：Claude 2模型允许用户在每次提示中输入多达100K的标记，这意味着它可以处理从技术文档到整本书的大量数据。此外，Claude 2现在也可以一次性编写从备忘录到信件到几千个标记的长文档。

2. 代码能力的增强：Claude 2的模型在编码方面进行了明显的优化，Sourcegraph是一个代码AI平台，他们的编码助理Cody利用Claude 2改进的推理能力为用户查询提供更准确的答案，同时也可以提供高达100K的上下文窗口。此外，Claude 2接受了更多的最新数据培训，这意味着它拥有了新框架和库的知识供Cody参考。

3. 安全性的增强：Anthropic对Claude 2进行了一系列的安全性优化。首先，Claude 2在内部红队评估中的表现比Claude 1.3好出了2倍，这意味着它在回应可能有害的提示时更能产生无害的回应。此外，为了提高模型的输出安全性，Anthropic使用了多种安全技术，并进行了广泛的红队测试。

对于追求最前沿技术的LLM应用开发者，Claude 2无疑为他们提供了一个值得关注的选择。

四大主要使用场景

随着技术的持续进步，为开发者带来了更多创新的机会，其中最为引人注目的是LLM应用的快速发展。其中，Claude 2模型为LLM应用带来了新的灵感和实践机会。Claude 2在LLM应用开发中的四大主要使用场景。

1. 处理海量文本

无论开发者面临的是文档、电子邮件、常见问题解答、聊天记录还是其他内容，Claude 2都能提供卓越的支持。此模型可以编辑、重写、总结、分类、提取结构化数据，并根据内容进行问答等操作。这为LLM应用开发者提供了一个强大的工具，助力他们更高效地处理和分析文本数据。

2. 自然对话交流。Claude 2可以在对话中扮演各种角色。只需为其提供角色详情和常见问题的解答，它就能与用户进行自然、相关的双向对话。这为开发者在LLM应用中实现流畅的用户互动提供了可能。

3. 获得答案。Claude 2拥有广泛的通用知识，这些知识来源于其庞大的训练语料库，包括技术、科学和文化知识。除了常见的自然语言，Claude 2还能理解和生成多种编程语言。这为开发者提供了在LLM应用中集成知识库或编程助手的机会。

4. 自动化工作流。Claude 2能够处理各种基本指令和逻辑场景，包括按需格式化输出、执行if-then语句以及在单一提示中进行一系列逻辑评估。这使得开发者能够在LLM应用中实现复杂的自动化任务和工作流。

用户关心的10个常见问题

在LLM应用的开发过程中，开发者可能对Anthropic的Claude 2模型有很多疑问。为了更好地帮助初学者和开发者了解Claude 2，以下列出了关于该模型的10个常见问题及其解答。

1. Claude有哪些版本可供选择？

目前提供两个版本的Claude：

Claude：是最强大的模型，擅长从复杂对话和创意内容生成到详细指导的各种任务。

Claude Instant：较快、较便宜但仍然非常能干的模型，可以处理包括休闲对话、文本分析、总结和文档问题回答等任务。

2. Claude的价格是多少？

https://www-files.anthropic.com/production/images/model\_pricing\_july2023.pdf。

3. Claude支持哪些语言？

Claude主要以英语为训练基础，但在其他常见语言中也表现出色。此外，Claude还对常见的编程语言有深入的了解。

4. Claude可以访问互联网吗？

不可以。Claude被设计为独立的，不会搜索互联网来响应。但可以为Claude提供互联网上的文本，并要求其对该内容执行任务。

5. 什么是宪法训练？

宪法训练是一个训练模型遵循所需行为“宪法”的过程。Anthropic的核心模型经过宪法训练，目的是变得有助于、诚实和无害。

6. "HHH"是什么意思？

"HHH"代表Helpful（有帮助）、Honest（诚实）和Harmless（无害）。这是构建与人们利益一致的AI系统（如Claude）的三个组件。

7. 如何进一步自定义Claude的行为？

可以通过提示广泛地修改Claude的行为。提示可以用来解释所需的角色、任务、背景知识以及所需响应的几个示例。

8. Claude模型可以进行微调吗？

在大多数情况下，我们相信精心设计的提示可以在没有微调的费用或延迟的情况下为您提供所需的结果。但一些大型企业用户可能会从微调模型中受益。

9. Claude的上下文窗口有多长？

输入和输出的综合上下文窗口约为100,000个标记，这大约相当于70,000个单词，具体取决于内容类型。

10. Claude可以进行嵌入吗？

目前还不行。我们发现开源的SBERT嵌入对于大多数用途已经足够好。

# Cohere 模型介绍

随着LLM应用的广泛应用，开发者对于高效、高性能的语言模型的需求日益增强。在这一背景下，Cohere应运而生，为开发者提供了一个先进的语言处理API。

Cohere的核心能力

Cohere不仅训练了大型的语言模型，并通过一个简洁的API为开发者提供服务，它还允许用户根据自己的需求训练定制的大型模型。这意味着开发者无需为收集大量的文本数据、选择合适的神经网络架构、分布式训练或模型部署而感到困扰，Cohere为开发者处理了所有这些复杂性。

Cohere的模型类型

Cohere为开发者提供了两大类的模型：

生成模型：通过generate端点，开发者可以访问这类模型。这些模型的代表包括像GPT2和GPT3这样的模型。

表示模型：通过embed端点，开发者不仅可以访问这类模型，还可以获取输入文本的嵌入向量。BERT是这类模型的代表。

对于希望在LLM应用开发中实现前沿语言处理功能的初学者和开发者，Cohere提供了一个高效且功能强大的解决方案。无论是需要生成内容，还是需要理解和表示语言，Cohere都为开发者提供了一站式的解决方案。

三大LLM应用案例

在现代LLM应用开发中，语言模型的功能越来越强大，开发者可利用这些功能来解决实际问题。Cohere作为一款领先的大型语言模型，为开发者提供了多种实用功能。以下将深入探讨Cohere的三大应用案例，助力开发者更好地理解其在LLM应用中的潜在价值。

1. 文本摘要与改写。随着文本生成技术的进步，大型语言模型如Cohere已经能够生成近乎人类水平的文本。其中，文本摘要与改写成为了开发者们的热门应用点。开发者可以通过Cohere为输入文本生成有意义的摘要或改写，仅需在提示中提供任务描述。此外，Cohere为文本摘要提供了Co.summarize端点，为开发者进一步简化了任务。

2. 文本分类。文本分类是语言处理中最常见的用例之一。利用Cohere的语言模型，开发者可以构建高效的分类器来自动化语言任务，从而节省大量时间和精力。Cohere不仅提供了简单的Classify端点进行分类，还允许开发者在embed端点之上构建更高级的分类器。

3. 语义相似性判断。在客服领域，经常会有大量重复的问题需要回答。Cohere的语言模型能够判断文本的相似性，从而确定一个新问题是否与FAQ部分已经回答的问题相似。通过计算两个嵌入的余弦相似性，开发者可以得到一个相似性得分，然后根据这个得分采取相应的行动，例如显示与最相似问题的答案。

# PaLM 2模型介绍

PaLM 2代表了Google在机器学习和负责任的AI领域不断创新的成果，是继PaLM后的下一代大型语言模型。作为LLM应用的开发者，理解PaLM 2的基础构造和核心优势对于充分利用其功能至关重要。

PaLM 2模型的特点

高级推理任务：PaLM 2在编码、数学、分类和问题回答、翻译和多语言能力以及自然语言生成等高级推理任务上都表现出色。

超越先前模型：相较于之前的最先进的LLM应用，包括PaLM，PaLM 2在各种任务上都有更好的表现。

构建方法：PaLM 2之所以能够实现这些任务，归功于其构建方式，结合了计算最优缩放、改进的数据集混合和模型架构的改进。

负责任的AI：PaLM 2基于Google负责任地构建和部署AI的方法，经过了严格的潜在危害和偏见、能力和下游用途的评估。

以PaLM 2为基础的产品

除了作为一个独立的大型语言模型，PaLM 2还为其他最先进的模型提供支持，如Med-PaLM 2和Sec-PaLM。此外，它正在为Google的一些生成性AI功能和工具提供动力，如Bard和PaLM API，这为LLM应用开发者提供了更广泛的实际应用场景。

三大核心功能

**1. 推理**。PaLM 2在复杂任务的分解以及对人类语言细微差异的理解上，相比之前的LLM，如PaLM，表现得更为出色。它能够非常精准地解读谜语和习语，这需要对词语的模糊和比喻意义有深入的理解，而不仅仅是字面意义。

**2. 多语言翻译。**与PaLM相比，PaLM 2在更大规模的多语言文本上进行了预训练。这使得它在多语言任务上具有显著的优势。通过大量的多语言文本预训练，PaLM 2为开发者在LLM应用中实现高效的多语言处理提供了坚实的基础。

**3. 编码**。PaLM 2的另一个亮点是它在大量的网页、源代码和其他数据集上进行的预训练。这意味着它不仅擅长流行的编程语言，如Python和JavaScript，而且还能够生成Prolog、Fortran和Verilog等专用编程语言的代码。结合其语言处理能力，可以帮助团队跨语言进行合作。

# Pinecone**向量数据库**介绍

Pinecone简介

对于初步接触LLM应用开发的开发者来说，选择一个高性能的向量搜索工具是关键的初步决策。Pinecone为此提供了一个完美的解决方案。

Pinecone是一个云原生的向量数据库，专门为高性能向量搜索应用程序设计。借助其托管服务和简化的API接口，开发者可以无缝地集成其功能，而无需过多关注底层基础架构的细节。

接下来，我们将详细探讨Pinecone的主要特性，这些特性使其在LLM应用开发领域中脱颖而出：

**高速查询性能**：Pinecone确保即使在数十亿条目中也能保持超低的查询延迟，满足实时应用的需求。

**实时索引更新**：随着数据的添加、修改或删除，索引可以实时更新，确保数据的即时性和准确性。

**过滤功能**：Pinecone允许开发者结合元数据过滤器进行向量搜索，这有助于获得更加相关和快速的查询结果。

**无缝托管服务**：Pinecone的完全托管特性使得开发者可以更加专注于LLM应用的开发和优化，而不是数据库的维护和管理。

Pinecone**的主要应用场景**

对于那些正在研究LLM应用开发的开发者，了解如何在实际应用中利用向量数据库如Pinecone是至关重要的。Pinecone由于其高效性和灵活性，已被广泛应用于多种场景。以下我们将深入探讨Pinecone的主要使用场景：

**语义文本搜索**：开发者可以利用NLP转换器和句子嵌入模型将文本数据转化为向量嵌入。随后，这些向量可以被Pinecone索引和搜索，从而实现高效的语义文本搜索功能。

**生成问答系统**：当接收到用户的查询时，可以从Pinecone检索相关的上下文数据。这些数据随后可以传递给如OpenAI这样的生成模型，产生与真实数据相一致的答案。

**混合搜索**：开发者可以结合语义和关键字搜索，让Pinecone在一个查询中同时执行，从而得到更加相关的搜索结果。

**图像相似度搜索**：首先，将图像数据转换为向量嵌入并使用Pinecone进行索引。当用户提交查询图像时，将其转换为向量并检索在Pinecone中的相似图像，为用户提供相似内容的图像。

**产品推荐系统**：在电子商务领域，基于代表用户的向量，Pinecone可以有效地生成产品推荐，从而为用户提供更个性化的购物体验。

Pinecone核心概念解析

当开发者进入LLM应用开发的领域，理解Pinecone的关键概念将为他们提供明确的方向和坚实的基础。在本部分，我们将深入探讨Pinecone的几个核心概念，从而为开发者铺设通往成功应用的路：

**向量搜索**： 传统搜索方法主要围绕关键字进行，但在向量数据库中，搜索的焦点转向了由ML生成的数据表示 - 向量嵌入。这种搜索方法的目标是找到与查询最相似的项目。

**向量嵌入**： 向量嵌入是表示对象的数字集合，它的特点是能够捕捉对象集合中的语义相似性。这些嵌入是由经过训练的模型生成的。在Pinecone中，开发者可以遇到两种主要的向量嵌入：密集嵌入和稀疏嵌入。为了充分利用Pinecone，开发者需要熟悉如何使用这些向量嵌入。

**向量数据库**： 作为一种特殊的数据库，向量数据库专注于索引和存储向量嵌入，以实现高效的管理和快速的检索。但是，与单纯的向量索引相比，向量数据库如Pinecone提供了更多高级功能。这些功能包括索引管理、数据管理、元数据存储、过滤和水平扩展等。

Pinecone工作流程探究

对于初学者和那些希望使用LLM构建应用的开发者来说，了解Pinecone的工作流程是开发过程中的关键步骤。

**（一）索引的设置**：

**创建索引**：首先，为数据创建一个索引，这是存储和检索向量的关键结构。

**连接索引**：一旦索引创建完毕，开发者需要确保能够与之建立连接。

**数据插入**：接下来，开发者将数据和相应的向量插入创建的索引中。

**（二）索引的使用**：

**查询数据**：在索引中查询特定数据或向量。

**数据过滤**：基于特定条件，开发者可以过滤检索到的结果，确保结果的相关性。

**获取数据**：根据需要，可以检索索引中的特定数据或向量。

**数据更新**：为了保持数据的实时性和准确性，开发者可以插入更多的数据或更新现有的向量。

**（三）索引与数据管理**：

**管理索引**：包括对索引的优化、备份和恢复等操作。

**数据管理**：涉及到数据的删除、修改和备份等任务。

# Milvus向量数据库介绍

Milvus简介

当我们谈论大规模嵌入向量的存储、索引和管理，Milvus向量数据库凭其独特的特性和优势成为了这一领域的明星。自2019年创建以来，Milvus的核心愿景是处理由深度神经网络和其他机器学习（ML）模型产生的大量嵌入向量。

与传统的关系型数据库不同，它们主要处理符合预定义模式的结构化数据，Milvus从其核心被设计为处理从非结构化数据转化而来的嵌入向量。这种设计意味着Milvus能够无缝地处理万亿级的向量索引。

为什么这种能力如此重要？随着互联网、物联网和社交媒体的普及，非结构化数据，如电子邮件、学术论文、传感器数据和社交媒体图片，已成为主流。为了使这些数据对于机器有意义，嵌入技术被用于将它们转换为向量形式。这正是Milvus所擅长的领域。通过存储和索引这些向量，Milvus可以计算两个向量间的相似距离，从而判断原始数据的相似性。

对于希望在LLM应用中使用非结构化数据的开发者，了解并利用Milvus的这些功能将帮助他们更有效地进行数据分析和提取有价值的见解。

窗体顶端

Milvus**的主要应用场景**

在构建和优化LLM应用时，开发者经常面临处理和搜索大量数据的挑战。这正是Milvus展现其强大功能的地方。以下是Milvus在各种应用中的主要应用场景：

图像相似性搜索：Milvus使得从大型数据库中即时返回最相似的图像成为可能，实现了高效的图像搜索功能。

视频相似性搜索：通过将视频的关键帧转换为向量，并利用Milvus进行处理，可以在接近实时的速度下搜索和推荐亿级别的视频。

音频相似性搜索：无论是语音、音乐、音效还是其他类似的声音，Milvus都能在短时间内快速查询大量音频数据。

分子相似性搜索：对于生物技术和化学领域，Milvus能够对特定的分子进行快速的相似性搜索，子结构搜索或超结构搜索。

推荐系统：基于用户的行为和需求，Milvus可以为LLM应用提供信息或产品的精准推荐。

问答系统：为了实现交互式的数字问答机器人，Milvus能够自动、准确地回答用户的问题。

DNA序列分类：在基因研究中，通过与Milvus比较相似的DNA序列，可以在毫秒级别内准确地对一个基因进行分类。

文本搜索引擎：对于需要处理大量文本数据的应用，Milvus能够通过与文本数据库中的关键字进行比较，帮助用户快速找到他们需要的信息。

Milvus核心概念解析

随着数据的爆炸性增长，开发者在构建LLM应用时面临着处理和理解大量非结构化数据的挑战。Milvus的核心概念为开发者提供了一个框架，帮助他们更好地处理这些数据。下面，我们将深入解析这些概念：

非结构化数据：这是指不遵循预定义模型或组织方式的数据。它包括图像、视频、音频和自然语言等信息。事实上，非结构化数据占据了全球数据的约80%。为了使这些数据有意义，必须将它们转换为可以被机器理解的格式，这就是向量的角色。

嵌入向量：嵌入向量是非结构化数据的特征抽象，例如电子邮件、物联网传感器数据、社交媒体照片和蛋白质结构等。在数学上，嵌入向量可以是浮点数或二进制数的数组。通过利用现代嵌入技术，开发者可以将非结构化数据转换为嵌入向量，从而为其LLM应用提供一个坚实的基础。

向量相似度搜索：这是一个搜索过程，将一个向量与数据库中的向量进行比较，目的是找到与查询向量最为相似的向量。为了加速这个搜索过程，通常使用近似最近邻搜索算法。当两个嵌入向量相似时，它们代表的原始数据源也是相似的。

Milvus支持的索引和度量

了解Milvus所支持的索引和度量，为构建高效的LLM应用打下坚实的基础。

索引是数据的组织方式，它定义了如何存储和检索数据。在Milvus中，大部分索引类型使用近似最近邻搜索（ANNS）技术。以下是一些重要的索引类型：

**FLAT**：适合于小规模数据集，提供完全准确和精确的搜索结果。

**IVF\_FLAT**：量化索引，适合于在查询速度和精度之间寻求平衡的场景。

**IVF\_SQ8**：在资源有限的场景中，此量化索引可以显著降低资源消耗。

**IVF\_PQ**：为了获得更高的查询速度，此量化索引可能牺牲一些精度。

**HNSW**：基于图形的索引，适合于高搜索效率需求的场景。

**ANNOY**：基于树形结构的索引，适合于寻求高召回率的场景。

在LLM应用中，度量方法的选择对于向量的分类和聚类性能至关重要。在Milvus中，相似度度量用于确定向量之间的相似性：

对于浮点嵌入，以下两种度量方法常用：

**欧氏距离（L2）**：在计算机视觉领域中常用。

**内积（IP）**：在自然语言处理领域中常用。

而对于二进制嵌入，以下是一些广泛应用的度量方法：

**哈明距离**：在自然语言处理中常用。

**杰卡德距离 & 塔尼莫托距离**：这两种度量方法在分子相似性搜索中都有广泛应用。

**超结构距离 & 亚结构距离**：这两种度量方法用于搜索分子的特定结构相似性。

为了在LLM应用中实现高效的数据检索和管理，开发者需要深入了解并正确选择索引和度量。