

Milvus 使用手册

版本	修订内容	修订时间	版本状态
V1.0.0	Milvus 产品使用手册	20251205	正式版

一、产品概述

1.1 产品介绍

Milvus 作为一款开源的高性能向量数据库，具有出色的向量检索性能、高扩展性和易用性，在人工智能、机器学习、计算机视觉、自然语言处理等领域展现出了巨大的应用潜力。目标市场可以根据应用领域细分为自然语言处理、计算机视觉、推荐系统等多个市场。其客户群体主要包括互联网、电商、金融、医疗、安防等行业的企业，以及科研机构 and 高校等。

1.2 产品核心能力

Milvus 的核心能力是作为一个高性能、高扩展性的云原生向量数据库，专门为处理大规模非结构化数据和支持人工智能应用而设计。

它的能力主要围绕高效的向量相似性搜索展开，具体体现在以下几个方面：

1. 高效的向量搜索与检索

(1) 大规模相似性搜索 (ANN)：能够从数十亿甚至千亿级的高维向量数据集中，快速、准确地找到与查询向量最相似的 K 个向量 (K-近邻搜索)。

(2) 异构计算支持 (Knowhere): Milvus 的核心向量执行引擎 Knowhere 能够集成并优化各种流行的向量相似性搜索库 (如 Faiss, Hnswlib, Annoy)，并能控制在 CPU 或 GPU 上执行索引构建和搜索请求，以充分利用硬件能力。

(3) 多样化的搜索模式: 除了基础的 ANN 搜索，还支持过滤搜索 (基于标量字段的条件过滤)、范围搜索和混合搜索 (结合向量和标量数据)，极大地满足了复杂的业务检索需求。

2. 高性能与底层优化

(1) C++ 搜索引擎: 使用 C++ 语言实现核心搜索引擎组件，以实现高性能、底层优化和高效的资源管理。

(2) 硬件感知优化: 集成了汇编级向量操作、多线程并行化和调度等优化，最大限度地利用硬件性能。

(3) 面向列的存储 (Column-Oriented): 采用面向列的数据库系统, 在执行查询时只读取涉及的特定字段, 大大减少了数据访问量, 并有利于向量化操作。

3. 灵活的功能与应用支撑

(1) 多模态检索: 结合 AI 推理和 Embedding 模型, 高效索引和检索图片、视频、文本、声纹等多种非结构化数据。

(2) 数据导入与管理: 支持高通量数据导入, 并提供数据分片、分区键、可调一致性模型等管理功能。

1.3 产品优势

为客户提供一键式的 Milvus 应用部署体验, 省去繁琐的部署过程快速赋能人工智能、机器学习、计算机视觉、自然语言处理等领域的应用。

二、使用指南

2.1 Milvus 使用

前端界面: <http://your-eip:9091/webui>

后端调用: <http://your-eip:19530>

2.2 创建数据库

Python ▾

```
1 # 创建数据库
2 from pymilvus import MilvusClient
3
4 client = MilvusClient(
5     uri="http://localhost:19530",
6     token="root:Milvus"
7 )
8
9 client.create_database(
10     db_name="my_database_1"
11 )
```

2.3 查看数据库

Python ▾

```
1 # 查看所有数据库
2 client.list_databases()
3
4 # 输出
5 # ['default', 'my_database_1', 'my_database_2']
6
7 # 查看数据库详情
8 client.describe_database(
9     db_name="default"
10 )
11
12 # 输出
13 # {"name": "default"}
```

2.4 使用数据库

Python ▾

```
1 # 使用数据库
2 client.use_database(
3     db_name="my_database_2"
4 )
```

2.5 删除数据库

Python ▾

```
1 # 删除数据库
2 client.drop_database(
3     db_name="my_database_2"
4 )
```

常见问题 Q&A

Q1: Milvus 是什么？

A: Milvus 是一个开源的、云原生的 向量数据库（Vector Database）。它专门设计用于存储、索引和管理大规模的向量数据，并支持高并发的相似性搜索（K-近邻搜索）。

Q2: Milvus 和传统数据库有什么区别？

A: 传统数据库（如 MySQL、PostgreSQL）主要处理结构化数据（表格、数值），侧重于精确匹配。Milvus 处理非结构化数据（图片、文本、语音转换成的向量），侧重于相似性搜索。

Q3: 向量数据库能解决什么问题？

A: 解决了非结构化数据的检索问题，例如：图片搜索（以图搜图）、智能问答、推荐系统、语音识别和基因序列比对等。

Q4: Milvus 的查询速度快吗？

A: 非常快。它通过使用各种高效的 向量索引算法（如 HNSW, IVFFlat, Annoy）以及底层 C++ 引擎优化，实现了大规模数据集上的毫秒级相似性搜索。

Q5: 向量索引是如何工作的？

A: 向量索引算法通过牺牲一定的准确性（Approximate Nearest Neighbor, ANN）来换取巨大的性能提升。它们将高维空间划分或压缩，从而避免暴力地比较每一个向量。

Q6: 如何将数据导入 Milvus？

A: 需要先使用 AI 模型（如 BERT, ResNet 等）将原始数据（文本、图片等）转换成固定长度的向量（称为 Embedding）。然后通过 Milvus SDK (Python, Java, Go) 将这些向量数据写入 Milvus Collection 中。

版权与声明

© [2025] [天翼云科技有限公司]。保留所有权利。

本文档内容仅供参考。