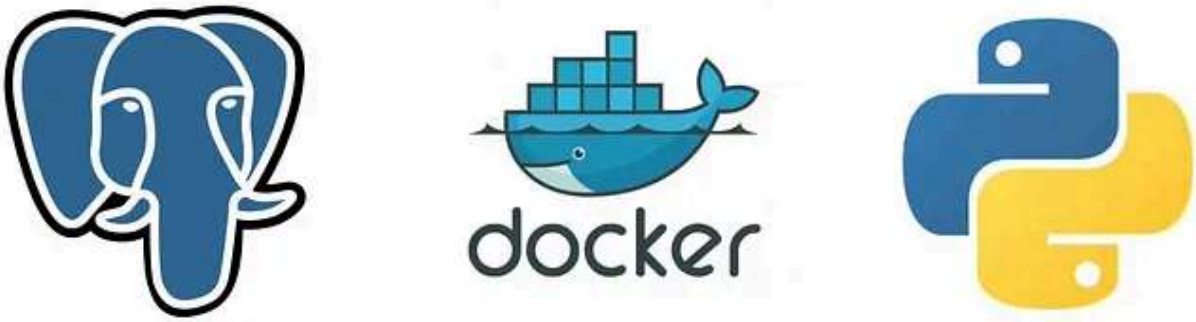
Setting Up Postg… pgvector in Dock… Guide

Adarsh Ajay · Follow

3 min read · Nov 8, 2024

Vector embeddings have become increasingly important in modern applications, from semantic search to AI-powered recommendations. In this guide, I'll walk you through setting up PostgreSQL with pgvector in Docker, enabling you to store and query vector data efficiently.

PostgreSQL + Pgvector



What is pgvector?

pgvector is a powerful PostgreSQL extension that adds vector similarity search capabilities to your database. It's particularly useful for applications involving machine learning, natural language processing, and image similarity search.

Prerequisites

- Docker installed on your machine
- Basic familiarity with PostgreSQL
- Terminal or command prompt access

Step 1: Pulling the Docker Image

We'll use the `ankane/pgvector` image, which comes with PostgreSQL and pgvector pre-installed. Open your terminal and run:

```
docker pull ankane/pgvector
```

Step 2: Running the PostgreSQL Container

Now, let's start a PostgreSQL container with pgvector enabled:

```
docker run -e POSTGRES_USER=myuser \
           -e POSTGRES_PASSWORD=mypassword \
           -e POSTGRES_DB=mydatabase \
           --name my_postgres \
           -p 5432:5432 \
           -d ankane/pgvector
```

Let's break down these parameters:

- `-e POSTGRES_USER=myuser` : Creates a database user
- `-e POSTGRES_PASSWORD=mypassword` : Sets the user's password
- `-e POSTGRES_DB=mydatabase` : Creates a new database
- `--name my_postgres` : Names your container
- `-p 5432:5432` : Maps the container's PostgreSQL port to your host
- `-d` : Runs the container in detached mode

Step 3: Connecting to Your Database

You can connect to your newly created database using the `psql` command-line tool

```
psql -h localhost -U myuser -d mydatabase -p 5432
```

You'll be prompted for the password you set earlier.

Step 4: Enabling pgvector

Once connected to your database, enable the pgvector extension by running:

```
CREATE EXTENSION vector;
```


Step 5: Verifying the Installation

To ensure pgvector is properly installed, run:

```
SELECT * FROM pg_extension;
```

You should see `vector` listed among the installed extensions.

Using pgvector with Python

To interact with your vector-enabled database from Python, first install the PostgreSQL adapter:

```
pip install psycopg2-binary
```

Here's a simple example of how to use it:

```
import psycopg2
import numpy as np

# Connect to your database
conn = psycopg2.connect(
    dbname="mydatabase",
    user="myuser",
    password="mypassword",
    host="localhost"
)
# Create a table with a vector column
with conn.cursor() as cur:
    cur.execute("""
        CREATE TABLE items (
            id serial PRIMARY KEY,
            embedding vector(3)
        );
    """)
    conn.commit()
```

Common Issues and Troubleshooting

1. **Connection refused errors:** Make sure the Docker container is running:

```
docker ps | grep my_postgres
```

2 . Permission denied: Verify your username and password are correct

3 . Extension creation fails: Ensure you're connected as a user with sufficient privileges

Next Steps

Now that you have PostgreSQL with pgvector running in Docker, you can:

- Start storing vector embeddings from your machine learning models
- Implement semantic search in your applications
- Build AI-powered recommendation systems
- Experiment with similarity searches

Conclusion

Setting up PostgreSQL with pgvector in Docker provides a powerful foundation for building vector-based applications. This containerized setup ensures consistency across different environments and makes it easy to get started with vector similarity search in your projects.

Remember to secure your database properly before deploying to production, and consider your specific use case when configuring PostgreSQL parameters for optimal performance.

Happy coding! If you found this guide helpful, feel free to share it with others who might benefit from it.

Pgvector | Artificial Intelligence | Postgres | Programming | Python Programming

Written by Adarsh Ajay

5 Followers · 4 Following

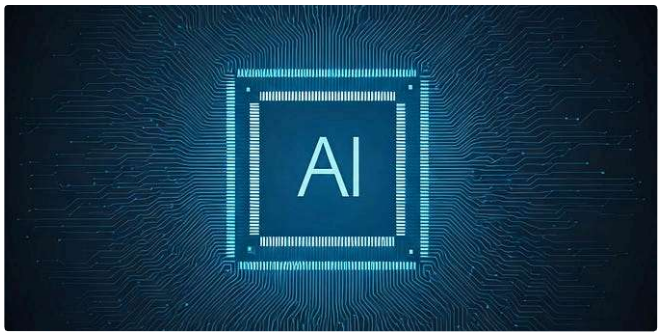
Follow

No responses yet

What are your thoughts?

Respond

More from Adarsh Ajay



Adarsh Ajay

How to Fine-Tune LLaMA 3.1 Locally: A Step-by-Step Guide

Fine-tuning large language models like Meta's LLaMA 3.1 is a powerful way to adapt these...

Sep 24, 2024

Adarsh Ajay

Unleashing the Power of Google Gemini with Python: A Step-by-...

Generative AI models have taken the tech world by storm, and Google's Gemini series...

Dec 17, 2024 50

Adarsh Ajay

Using PyMuPDF as Data Feeder in LLM / RAG Applications

In the rapidly evolving landscape of Large Language Models (LLMs) and Retrieval-...

Dec 2, 2024 50

Adarsh Ajay

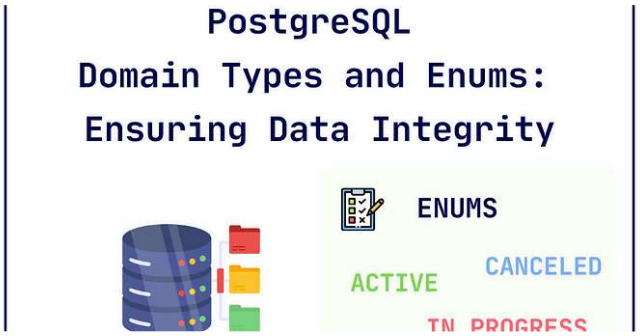
Fine-Tuning Mistral 7B: A Step-by-Step Guide

Fine-tuning a large language model like Mistral 7B can greatly enhance its...

Sep 24, 2024

See all from Adarsh Ajay

Recommended from Medium



Java Jedi

PostgreSQL Domain Types and Enums: Ensuring Data Integrity

Welcome back to the third post in our series on advanced PostgreSQL features. In the...

Jan 7 14

In DevOps.dev by Chris Chin

Making PostgreSQL Run SQL Faster

Optimizing PostgreSQL's performance for SQL queries is crucial for database...

Jan 8 69

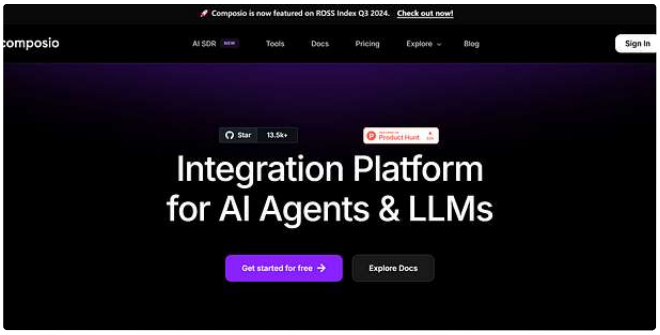
Lists

General Coding Knowledge
20 stories · 1874 saves

ChatGPT
21 stories · 942 saves

AI Regulation
6 stories · 670 saves

Coding & Development
11 stories · 975 saves



Let's Code Future

In Level Up Coding by Jacob Bennett

15 Modern Open-Source Tools for Supercharge Your Next Project

13 open-source tools to supercharge your next big project this year!

5d ago 194

The 5 paid subscriptions I actually use in 2025 as a Staff Software...

Tools I use that are cheaper than Netflix

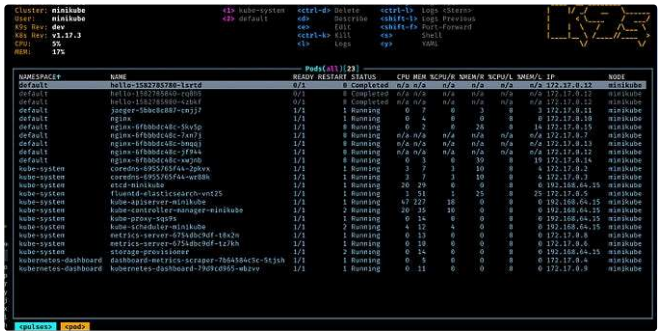
Jan 8 3.8K 92

Varun Singh

Python 3.14 Released — Top 5 Features You Must Know

Faster Annotations & Mind-Blowing Updates You NEED to Know!

Dec 31, 2024 823 4



In ITNEXT by Alex Pliutau

Essential CLI/TUI Tools for Developers

An opinionated list of CLI/TUI applications for developer productivity.

Jan 7 470 7

See more recommendations