Resource List

Welcome to the resource list for the course Leveraging *PostgreSQL with Retrieval-Augmented Generation (RAG)*. Below are valuable resources that will enhance your understanding of RAG and its integration with PostgreSQL. These resources include tutorials, articles, and documentation that provide comprehensive insights into RAG's principles, applications, and best practices.

Recommended Resources

1. Nexla: RAG Tutorial and Best Practices

Explore an in-depth tutorial on retrieval-augmented generation, covering its differences from traditional AI models and its applications across various industries. This resource includes best practices and insights into the future potential of RAG.

- Link: https://nexla.com/ai-infrastructure/retrieval-augmented-generation/

2. NVIDIA Blog: What Is Retrieval-Augmented Generation?

This blog post from NVIDIA explains how RAG enhances the accuracy and reliability of generative AI models by linking them to external resources. It discusses the implementation process and practical applications of RAG.

- Link: https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/

3. Pinecone: Understanding RAG

Learn about how RAG architecture provides relevant data to generative AI applications, with a focus on cost-effectiveness and performance improvements.

- Link: https://www.pinecone.io/learn/retrieval-augmented-generation/

4. AWS: What Is RAG?

AWS provides an explanation of how RAG optimizes large language model outputs by referencing external knowledge bases, offering a cost-effective approach to improving AI accuracy without retraining models.

- Link: https://aws.amazon.com/what-is/retrieval-augmented-generation/?nc1=h_ls

5. LangChain Documentation: Building a RAG App

This tutorial guides you through building a simple Q&A application using RAG, highlighting the architecture and additional resources for advanced techniques.

- Link: https://python.langchain.com/docs/tutorials/rag/

6. Vanna.AI Documentation

Explore detailed documentation on how Vanna uses retrieval augmentation to generate accurate SQL queries. This resource provides insights into setting up Vanna and using it with various databases.

- Link: https://vanna.ai/docs/

7. Vanna.AI: GitHub Repository

Access the open-source codebase for Vanna on GitHub. This repository includes examples, installation instructions, and customization options for using Vanna in different environments.

- Link: https://github.com/vanna-ai/vanna

8. Vanna.AI: Vision and Roadmap

Understand the future direction of Vanna through its vision and roadmap documentation. This resource outlines goals for accuracy, interactions, and autonomy in data analysis.

- Link: https://vanna.ai/docs/vision/

9. Tech Bits by Doug Ortiz

Explore a variety of tutorials and insights on PostgreSQL and related technologies. Doug Ortiz's YouTube channel, Tech Bits, offers valuable content for anyone interested in cloud computing, big data, data science, data analytics, DevOps, and databases. Subscribe to stay updated with the latest tips and techniques in database management.

- Link: https://www.youtube.com/@techbits-do/videos

10. PostgresAI-Toolkit: GitHub Repository

Discover PostgresAI-Toolkit, a comprehensive GitHub repository focused on integrating PostgreSQL with artificial intelligence. This toolkit provides resources, tools, and examples to enhance database management and AI capabilities using PostgreSQL. It is ideal for developers and data scientists looking to leverage AI in their database operations.

- Link: https://github.com/illustris-admin/PostgresAI-Toolkit

11. LinkedIn Learning: *Master Meta-Commands in PostgreSQL*

If you manage a relational database in PostgreSQL, you need to know how to utilize the extensive functionality and power of meta-commands. In this course, instructor Doug Ortiz provides a comprehensive overview of the concept of meta-commands and shows you how to start incorporating them into your everyday workflow to manage your database more efficiently and effectively.

Explore ways to streamline your productivity by using meta-commands to manage user access to database objects, analyze query performance, and import, back up, and restore your data as needed to secure your data and prevent future loss. Along the way, test out your new skills in the practice challenges at the end of each section of the course.

- Link: https://www.linkedin.com/learning/master-meta-commands-in-postgresql

12. Other Interesting Data Analytics Questions for RAG Analysis

Which job titles have the highest number of employees at each career level?

```
SELECT career_level, job_title, COUNT(*) AS employee_count
FROM employee_data
GROUP BY career_level, job_title
ORDER BY career_level, employee_count DESC;
```

How does compensation vary across job titles within the same career level?

```
SELECT career_level, job_title, AVG(compensation) AS
       average_compensation
FROM employee_data
GROUP BY career_level, job_title;
```

Are there any noticeable trends in hiring over time, such as peaks in certain months or years?

```
SELECT DATE_TRUNC('month', hire_date) AS hire_month, COUNT(*) AS
       hire_count
FROM employee_data
GROUP BY hire_month
ORDER BY hire_month;
```

How is the workforce distributed across different work locations?

```
SELECT work_location, COUNT(*) AS employee_count
FROM employee_data
GROUP BY work_location;
```

What is the distribution of role types (for example, full-time, part-time) within each department or location?

```
SELECT role_type, COUNT(*) AS employee_count
FROM employee_data
GROUP BY role_type;
```

What percentage of employees identify as veterans, and how does this vary by department or location?

```
SELECT department, COUNT(*) FILTER (WHERE veteran_status = 'Yes')
       AS veteran_count
FROM employee_data
GROUP BY department;
```

How many employees report having a disability, and what roles do they predominantly occupy?

```
SELECT job_title, COUNT(*) FILTER (WHERE disability_status =
      'Yes') AS disability_count
FROM employee_data
GROUP BY job_title;
```

Conclusion

These resources are designed to support your learning journey as you explore the integration of PostgreSQL with retrieval-augmented generation. By leveraging these materials, you can deepen your understanding of how RAG can enhance AI capabilities and improve data management processes.

For further questions or discussions, feel free to reach out through our course's Q&A section or connect on LinkedIn.

Happy learning!