(RESEARCH ARTICLE)

Check for updates

Database management systems for artificial intelligence: Comparative analysis of postgre SQL and MongoDB

Yijie Weng [1, *] and Jianhao Wu [2]

[1] University of Maryland College Park, MD, USA.
[2] Cornell University, NY, USA.

Abstract

The rapid evolution of artificial intelligence (AI) has amplified the need for efficient database management systems (DBMS) to handle the growing volume, variety, and velocity of data. PostgreSQL, a robust relational database, and MongoDB, a leading NoSQL solution, are two widely adopted DBMSs in AI applications, each offering unique advantages. This paper provides a comprehensive comparative analysis of PostgreSQL and MongoDB, focusing on their suitability for AI use cases. Key evaluation criteria include data modeling, query complexity, scalability, ACID compliance, indexing, and integration with AI frameworks. PostgreSQL excels in scenarios requiring strict data consistency, complex querying, and structured data, making it ideal for financial modeling, scientific research, and feature engineering. Conversely, MongoDB's schema-less design, horizontal scalability, and native support for semi-structured data align with real-time analytics, IoT, and evolving AI datasets. The study highlights that the choice between the two databases depends on specific project requirements and proposes hybrid approaches to leverage their complementary strengths. This analysis aims to guide AI practitioners in making informed database decisions to optimize performance, scalability, and flexibility in AI systems.

1. Introduction

The rapid advancement of artificial intelligence (AI) has led to an exponential increase in data generation and consumption, necessitating robust database management systems (DBMS) to efficiently handle, process, and analyze vast amounts of information. Among the myriad of DBMS options available, PostgreSQL and MongoDB have emerged as prominent choices for AI applications, each offering unique strengths and capabilities (Özsu & Valduriez, 1999).

PostgreSQL, a powerful relational DBMS, is renowned for its adherence to ACID (Atomicity, Consistency, Isolation, Durability) principles, extensive feature set, and ability to handle complex queries (Obe & Hsu, 2017). Its strong support for structured data and advanced indexing techniques makes it particularly suitable for AI projects requiring strict data integrity and sophisticated analytical operations (Sharma et al., 2012).

Conversely, MongoDB, a leading NoSQL database, offers a flexible document-based data model that excels in managing semi-structured and unstructured data commonly encountered in AI applications (Chodorow, 2019). Its horizontal scalability and ability to handle large volumes of diverse data types align well with the dynamic nature of AI datasets and real-time analytics requirements (Győrödi et al., 2015).

---

* Corresponding author: Yijie Weng

The choice between PostgreSQL and MongoDB for AI projects often depends on specific use cases, data characteristics, and performance requirements. While PostgreSQL's relational model provides strong consistency and complex query capabilities, MongoDB's schema-less design offers greater flexibility and scalability for evolving data structures (Nayak et al., 2013). This comparative analysis aims to explore the strengths, limitations, and optimal use cases of PostgreSQL and MongoDB in the context of AI applications. By examining factors such as data modeling, query performance, scalability, and integration with AI frameworks, this study seeks to provide insights that can guide database selection decisions for AI researchers and practitioners.

2. Literature Review

Győrödi et al. (2015) compared MongoDB, a NoSQL database, with MySQL, a relational database, focusing on insertion, selection, and data aggregation performance. The study found MongoDB excels in insertion speed, particularly with large datasets, while MySQL performs better in complex queries and joins. MongoDB's flexibility with unstructured data and horizontal scalability make it advantageous for big data applications. This study offers empirical evidence for comparing NoSQL and relational databases, providing a framework applicable to evaluating PostgreSQL and MongoDB for AI applications.

Nayak et al. (2013) categorize NoSQL databases into key-value stores, document databases, column-oriented, and graph databases, explaining core concepts and highlighting scalability and flexibility advantages. They contrast NoSQL with relational databases, noting NoSQL's performance benefits with unstructured data. This foundational overview is valuable for understanding MongoDB's role within NoSQL and its comparison with relational systems like PostgreSQL. By exploring the fundamental differences, this article helps researchers decide which database system suits specific AI tasks like data storage and retrieval.

Sharma et al. (2018) analyze SQL and NoSQL databases, covering their architecture, data models, and evolution. They highlight SQL's strengths in consistency, complex queries, and ACID compliance, contrasting with NoSQL's scalability and suitability for unstructured data. The paper also compares various NoSQL types, including MongoDB, discussing ideal use cases. This balanced perspective on SQL and NoSQL provides context for evaluating PostgreSQL and MongoDB, offering insights into their respective strengths for tasks like machine learning, data preprocessing, and real-time analytics.

3. PostgreSQL and MongoDB Explanation

PostgreSQL is an advanced, open-source relational database management system (RDBMS) that has gained significant popularity in the field of artificial intelligence due to its robust features and reliability. Originally developed at the University of California, Berkeley, PostgreSQL has evolved into a powerful database solution that adheres to SQL standards while offering numerous extensions and advanced capabilities. PostgreSQL offers a comprehensive set of features that make it well-suited for AI applications. Its ACID compliance ensures data integrity, while support for advanced data types and indexing techniques enhances its ability to handle complex AI-related data structures and queries. The implementation of Multi-Version Concurrency Control allows for efficient simultaneous database access, and its extensibility enables customization for specific AI needs. PostgreSQL's full-text search capabilities further support natural language processing tasks. In the context of AI, these features translate to advantages such as strong data integrity, support for complex queries and analysis, reliable transactional support, and scalability for handling large datasets, making it a powerful choice for many AI workloads that require structured data management and sophisticated analytical capabilities.

MongoDB is a popular, open-source NoSQL database that has gained traction in the AI community due to its flexibility and scalability. Developed by MongoDB Inc., it is classified as a document-oriented database, which stores data in flexible, JSON-like documents called BSON (Binary JSON). MongoDB offers a set of features that are particularly advantageous for AI applications dealing with diverse and rapidly evolving data. Its flexible, schema-less document model allows for easy adaptation to changing data requirements, which is crucial in the dynamic field of AI. MongoDB's design for horizontal scalability through sharding makes it capable of handling the massive datasets often encountered in AI and machine learning projects. The database's high-performance read and write operations, coupled with its support for various indexing types and a powerful aggregation framework, make it well-suited for real-time AI systems and big data processing. Features like GridFS for efficient large file storage further enhance its utility for AI applications dealing with multimedia data. In the AI context, MongoDB's flexibility, scalability, performance, JSON-like structure, and geospatial capabilities make it an excellent choice for applications involving diverse data types, real-time analytics, and location-based analysis, particularly in scenarios where data structures are less rigid and may evolve over time.

Both PostgreSQL and MongoDB have their strengths in the context of AI applications. The choice between them often depends on the specific requirements of the project, such as data structure, scalability needs, consistency requirements, and the nature of the AI algorithms being employed.

4. PostgreSQL and MongoDB Comparison

4.1. Data Model and Schema Flexibility

4.1.1. PostgreSQL:

- Uses a rigid, predefined schema based on the relational model.
- Enforces data integrity through constraints, foreign keys, and data type validation.
- Well-suited for structured data with clear relationships.

4.1.2. MongoDB:

- Employs a flexible, schema-less document model.
- Allows for dynamic schema changes without downtime.
- Ideal for semi-structured or unstructured data common in AI applications.
- PostgreSQL's structured approach is beneficial for AI projects requiring strict data consistency and complex relationships. However, MongoDB's flexibility can be advantageous for projects with evolving data requirements or diverse data types, which is often the case in AI research and development.

4.2. Query Language and Complexity

4.2.1. - PostgreSQL:

- Uses standard SQL with extensive support for complex queries, joins, and subqueries.
- Offers powerful query optimization capabilities.
- Supports window functions and common table expressions (CTEs) for advanced analytics.

4.2.2. - MongoDB:

- Uses a custom query language based on method chaining and JSON-like syntax.
- Provides an aggregation framework for complex data transformations and analysis.
- Limited support for joins compared to relational databases.

PostgreSQL's SQL support makes it superior for complex analytical queries often required in AI data preprocessing and analysis. MongoDB's query language, while less expressive for complex operations, offers simplicity and aligns well with document-based data structures commonly used in AI applications.

4.3. Scalability and Performance

4.3.1. - PostgreSQL:

- Primarily designed for vertical scaling (adding more resources to a single server).
- Offers excellent performance for complex queries and transactions.
- Can handle large datasets but may face challenges with extremely high write loads.

4.3.2. - MongoDB:

- Built for horizontal scaling through sharding.
- Excels in write-heavy workloads and can handle massive datasets across distributed systems.
- May sacrifice some consistency for improved performance and scalability.
- MongoDB's horizontal scalability makes it more suitable for AI applications dealing with big data or requiring high write throughput. PostgreSQL, while scalable to a certain extent, is better suited for AI projects that prioritize data consistency and complex query performance over extreme scalability.

Figure 1 PostgreSQL and MongoDB Read Write Time Comparison by Dataset Size

4.4. ACID Compliance and Data Consistency

4.4.1. - PostgreSQL

- Fully ACID compliant, ensuring strong data consistency.
- Supports multi-version concurrency control (MVCC) for handling simultaneous transactions.
- Ideal for applications requiring strict data integrity and transactional reliability.

4.4.2. - MongoDB

- Offers tunable consistency levels, from eventual consistency to strong consistency.
- Provides atomic operations at the document level.
- May sacrifice some consistency for improved performance and scalability.
- PostgreSQL's strong ACID compliance makes it preferable for AI applications that require absolute data consistency, such as financial modeling or critical decision-making systems. MongoDB's flexible consistency model can be advantageous for AI applications that prioritize availability and partition tolerance over strict consistency, such as real-time recommendation systems or social network analysis.

4.5. Indexing and Query Optimization

4.5.1. - PostgreSQL

- Supports various index types, including B-tree, Hash, GiST, and GIN.
- Offers a sophisticated query planner and optimizer.
- Allows for partial and expression indexes.

4.5.2. - MongoDB

- Provides multiple index types, including single field, compound, multi-key, and text indexes.
- Supports geospatial indexing for location-based queries.
- Offers query optimization using covered queries and index intersection.

Both databases offer robust indexing capabilities, but their strengths differ. PostgreSQL's advanced indexing and query optimization are particularly beneficial for complex analytical queries in AI applications. MongoDB's indexing features, especially geospatial indexing, can be advantageous for AI projects involving location-based data or text analysis.
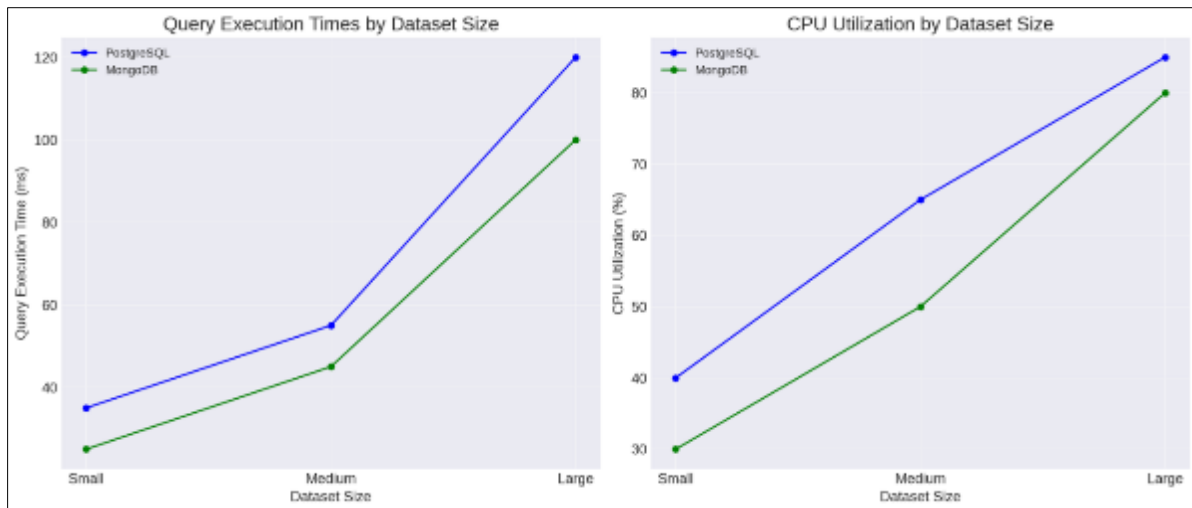
Figure 2 PostgreSQL and MongoDB Query Execution Times and CPU Utilization by Dataset Size

4.6. Support for Unstructured Data and AI-specific Features

4.6.1. - PostgreSQL:

- Offers extensions for handling JSON and other semi-structured data types.
- Supports full-text search capabilities.
- Can integrate with AI libraries through extensions and procedural languages.

4.6.2. - MongoDB:

- Native support for JSON-like documents and nested data structures.
- Provides a flexible model for storing and querying unstructured data.
- Offers built-in support for geospatial queries and text search.
- MongoDB's native support for semi-structured and unstructured data makes it particularly well-suited for AI applications dealing with diverse data types, such as text analysis, image processing, or IoT data. PostgreSQL, while capable of handling such data through extensions, excels in scenarios where the data has a more defined structure and relationships.

4.7. Integration with AI Frameworks and Tools

4.7.1. - PostgreSQL

- Can integrate with popular AI frameworks through database connectors and APIs.
- Supports in-database machine learning through extensions like MADlib.
- Works well with data analysis tools like Pandas and Jupyter Notebooks.

4.7.2. - MongoDB

- Offers native drivers for many programming languages used in AI development.
- Integrates easily with document-oriented AI frameworks and NoSQL-based tools.
- Provides connectors for big data processing frameworks like Apache Spark.
- Both databases can integrate well with AI frameworks and tools. PostgreSQL's strength lies in its compatibility with traditional data analysis tools and support for in-database machine learning. MongoDB's advantage is its natural fit with document-oriented AI workflows and big data processing frameworks.

5. Conclusion

The comparison between PostgreSQL and MongoDB in the context of AI applications reveals that both databases have their strengths and are suited for different aspects of AI development and deployment. The choice between them depends largely on the specific requirements of the AI project at hand. PostgreSQL excels in scenarios that require:

- Strong data consistency and ACID compliance
- Complex querying and data analysis
- Structured data with clear relationships
- Advanced indexing and query optimization for analytical workloads
- Integration with traditional data analysis tools and in-database machine learning

These characteristics make PostgreSQL particularly suitable for AI applications in fields such as financial modeling, scientific research, and any domain where data integrity and complex analytical queries are paramount. Its ability to handle sophisticated data relationships and support for advanced SQL features can greatly benefit feature engineering and data preprocessing tasks in machine learning pipelines. On the other hand, MongoDB shines in scenarios that prioritize:

- Flexibility in data modeling and schema design
- Horizontal scalability for handling big data
- High-performance read and write operations
- Native support for semi-structured and unstructured data
- Geospatial data processing and real-time analytics

These features make MongoDB an excellent choice for AI applications dealing with diverse and evolving data types, such as social media analytics, IoT data processing, content management systems, and real-time recommendation engines. Its scalability and performance characteristics are particularly beneficial for AI systems that need to handle large volumes of data or require real-time processing.

It's important to note that the boundaries between relational and NoSQL databases are becoming increasingly blurred, with PostgreSQL adding support for JSON and other NoSQL-like features, and MongoDB improving its support for complex queries and consistency models. This convergence suggests that future AI applications may benefit from hybrid approaches or multi-model databases that combine the strengths of both paradigms.

In conclusion, the selection between PostgreSQL and MongoDB for AI applications should be based on a careful analysis of the project's specific requirements, including:

- Data structure and consistency needs
- Scalability and performance demands
- Query complexity and analytical requirements
- Integration with existing AI frameworks and tools
- Long-term flexibility and maintainability considerations

By weighing these factors, developers and data scientists can make an informed decision that best supports their AI initiatives. In some cases, a hybrid approach using both databases for different aspects of the AI system may provide the optimal solution, leveraging the strengths of each database where they are most beneficial. As AI continues to evolve, database management systems will likely adapt further to meet the unique challenges posed by AI applications. Both PostgreSQL and MongoDB are actively developing features to better support AI workloads, and staying informed about these advancements will be crucial for making the best database choices in future AI projects

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] Özsu, M. T., & Valduriez, P. (1999). Principles of distributed database systems (Vol. 2). Englewood Cliffs: Prentice Hall.

[2] Obe, R. O., & Hsu, L. S. (2017). PostgreSQL: up and running: a practical guide to the advanced open source database. " O'Reilly Media, Inc.".

[3] Weng, Y., & Wu, J. (2024). Fortifying the Global Data Fortress: A Multidimensional Examination of Cyber Security Indexes and Data Protection Measures across 193 Nations. International Journal of Frontiers in Engineering Technology, 6(2), 13-28.

[4] Sharma, V., & Dave, M. (2012). Sql and nosql databases. International Journal of Advanced Research in Computer Science and Software Engineering, 2(8).

[5] Bradshaw, S., Brazil, E., & Chodorow, K. (2019). MongoDB: the definitive guide: powerful and scalable data storage. O'Reilly Media.

[6] Győrödi, C., Győrödi, R., Pecherle, G., & Olah, A. (2015, June). A comparative study: MongoDB vs. MySQL. In 2015 13th international conference on engineering of modern electric systems (EMES) (pp. 1-6). IEEE.

[7] Nayak, A., Poriya, A., & Poojary, D. (2013). Type of NOSQL databases and its comparison with relational databases. International Journal of Applied Information Systems, 5(4), 16-19.

[8] Weng, Y., & Wu, J. (2024). Leveraging Artificial Intelligence to Enhance Data Security and Combat Cyber Attacks. Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023, 5(1), 392-399.

[9] Barbierato, E., Gribaudo, M., & Iacono, M. (2014). Performance evaluation of NoSQL big-data applications using multi-formalism models. Future Generation Computer Systems, 37, 345-353.

[10] Han, J., Haihong, E., Le, G., & Du, J. (2011, October). Survey on NoSQL database. In 2011 6th international conference on pervasive computing and applications (pp. 363-366). IEEE.

[11] Weng, Y., & Wu, J. (2024). Big data and machine learning in defence. International Journal of Computer Science and Information Technology, 16(2), 25-35.

[12] Li, Y., & Manoharan, S. (2013, August). A performance comparison of SQL and NoSQL databases. In 2013 IEEE Pacific Rim conference on communications, computers and signal processing (PACRIM) (pp. 15-19). IEEE.

[13] Cattell, R. (2011). Scalable SQL and NoSQL data stores. Acm Sigmod Record, 39(4), 12-27.

[14] Weng, Y., Cao, Y., Li, M., & Yang, X. (2024). The Application of Big Data and AI in Risk Control Models: Safeguarding User Security. International Journal of Frontiers in Engineering Technology, 6(3), 154-164.

[15] Nalla, L. N., & Reddy, V. M. (2022). SQL vs. NoSQL: Choosing the Right Database for Your Ecommerce Platform. International Journal of Advanced Engineering Technologies and Innovations, 1(2), 54-69.

[16] Coronel, C., & Morris, S. (2019). Database systems: design, implementation and management. Cengage learning.

[17] Yan, C., Wang, J., Zou, Y., Weng, Y., Zhao, Y., & Li, Z. (2024, July). Enhancing credit card fraud detection through adaptive model optimization. In 2024 IEEE 7th International Conference on Big Data and Artificial Intelligence (BDAI) (pp. 49-54). IEEE.

[18] Munappy, A., Bosch, J., Olsson, H. H., Arpteg, A., & Brinne, B. (2019, August). Data management challenges for deep learning. In 2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA) (pp. 140-147). IEEE.

[19] Weng, Y., Wu, J., Kelly, T., & Johnson, W. (2024). Comprehensive overview of artificial intelligence applications in modern industries. arXiv preprint arXiv:2409.13059.

[20] Dean, J., Corrado, G., Monga, R., Chen, K., Devin, M., Mao, M., ... & Ng, A. (2012). Large scale distributed deep networks. Advances in neural information processing systems, 25.

[21] Warren, J., & Marz, N. (2015). Big Data: Principles and best practices of scalable realtime data systems. Simon and Schuster.

[22] Chen, Y., Chen, J., Weng, Y., Chang, C., Yu, D., & Lin, G. (2025). AdaMixup: A Dynamic Defense Framework for Membership Inference Attack Mitigation. arXiv preprint arXiv:2501.02182.

[23] Jung, M. G., Youn, S. A., Bae, J., & Choi, Y. L. (2015, November). A study on data input and output performance comparison of mongodb and postgresql in the big data environment. In 2015 8th international conference on database theory and application (DTA) (pp. 14-17). IEEE.

[24] Weng, Y., Fan, Y., Wu, X., Wu, S., & Xu, J. (2024, November). A Multi-Layer Alignment and Adaptive Weighting Framework for Multi-Task Model Fusion. In 2024 International Conference on Intelligent Robotics and Automatic Control (IRAC) (pp. 327-330). IEEE.

[25] Müller, A. C., & Guido, S. (2016). Introduction to machine learning with Python: a guide for data scientists. " O'Reilly Media, Inc.".

[26] VanderPlas, J. (2016). Python data science handbook: Essential tools for working with data. " O'Reilly Media, Inc.".