Engineering and Technology Journal e-ISSN: 2456-3358

Volume 10 Issue 03 March-2025, Page No.- 4326-4349

DOI: 10.47191/etj/v10i03.39, I.F. – 8.482

Machine Learning Approaches for Enhancing Query Optimization in Large Databases

Warveen merza eido[1], Hajar Maseeh Yasin [2]

[1,2]Akre University for Applied Sciences Technical College of Informatics Department of Information Technology Duhok, Iraq

ABSTRACT: More effective query optimization strategies in large-scale databases are required due to the growing volume and complexity of data in contemporary applications. Performance inefficiencies result from traditional query optimization techniques, such as rule-based and cost-based approaches, which frequently find it difficult to manage dynamic and complicated workloads. By utilizing deep learning, reinforcement learning, and predictive analytics to enhance query execution plans, indexing, and workload management, machine learning (ML) has become a game-changing method for improving query optimization. With its many advantages—including workload-aware indexing, adaptive tuning, and real-time performance improvements—ML-driven optimization approaches are especially well-suited for distributed and cloud-based database setups. However, challenges remain, such as the need for more explainable AI-powered optimizers, security vulnerabilities, and the high computational costs of training machine learning models. To ensure reliable and efficient database management, future research should focus on creating hybrid optimization frameworks, strengthening security measures, and making machine learning-based decision-making more explainable. By addressing these challenges, machine learning-powered query optimization could open the door to smarter, more flexible, and scalable database systems.

KEYWORDS: Machine Learning-based Query Optimization, Database Performance Enhancement, Reinforcement Learning in Databases, Big Data and Cloud-based Optimization, AI-driven Indexing Strategies, Deep Learning for Query Execution, Autonomous Database Management

1.0. INTRODUCTION

There is a great need for effective query optimization strategies in large-scale databases due to the quick expansion of data in contemporary applications. Performance bottlenecks frequently result from traditional query optimizers' inability to effectively handle complicated and dynamic workloads, such as rule-based and cost-based methods (Saleh & Zebari, 2025a). Machine learning (ML), which uses deep learning, reinforcement learning, and predictive analytics to improve query execution plans, indexing strategies, and workload predictions, has become a ground-breaking method to improve query optimization as data-driven applications grow (W. merza Eido & Yasin, 2025). These developments enable databases to adjust themselves according to past performance, which eventually shortens query execution times and enhances resource usage (Tato & Yasin, 2025). The capacity of machine learning to simulate intricate interactions between queries and execution parameters—something that conventional optimization approaches find difficult to accomplish—is one of the main benefits of machine learning in query optimization (Saleh & Yasin, 2025). For example, by continually learning from past executions to dynamically improve database setups, reinforcement learning allows adaptive optimization of query execution plans (W. merza Eido & Zeebaree, 2025). Additionally, supervised learning models can automatically recommend effective indexing algorithms and classify queries according to their complexity, thereby increasing system throughput and lowering computing overhead (Saleh & Zebari, 2025b). The need for effective query optimization strategies in large-scale databases has increased due to the exponential rise of data. Complex workloads are frequently too much for traditional methods, which results in ineffective execution and resource use (Akhtar & Farzana, 2024). In addition, learning database systems offer adaptive indexing techniques that improve scalability and efficiency in real-time settings (Marcus et al., 2021). Databases may now dynamically improve query execution plans based on past performance and workload patterns thanks to machine learning, which has become a potent solution (Thirupurasundari et al., 2023). Significant gains in query response times and indexing strategies have been shown with methods like reinforcement learning and predictive modeling (Ding et al., 2019). explainable AI methods, lowering the processing cost of ML model training, and putting strong security measures in place to protect AI-driven optimization procedures (W. merza Eido & Yasin, 2025). In order to protect AI-driven optimization processes, future research should concentrate on creating more explainable AI methodologies, lowering the computational expense of training ML models, and putting strong security measures in place (Islam, 2024). Overcoming these obstacles will enable ML-powered database optimization to advance further, opening the door for intelligent, self-optimizing database management systems that can manage workloads with ever-increasing complexity (M.M.F. Fahima et al., 2024). The capacity of machine learning to simulate intricate interactions between queries and execution parameters—something that conventional optimization approaches find difficult to accomplish—is one of the main benefits of machine learning in query optimization (Alamu, n.d.). By continually learning from past executions, reinforcement learning, for example, allows adaptive tweaking of query execution plans to dynamically improve database settings (Hamza Akhtar et al., 2025). Additionally, supervised learning models can automatically recommend effective indexing algorithms and classify queries according to their complexity, thereby increasing system throughput and lowering computational overhead (Dou et al., 2023).

2. RESEARCH METHODOLOGY

This section outlines the methodology followed in conducting the literature review and analysis presented in this paper. The research follows a structured approach to selecting, reviewing, and synthesizing existing literature on the impact of web technology, cloud computing, digital marketing, and machine learning in transforming enterprise systems.

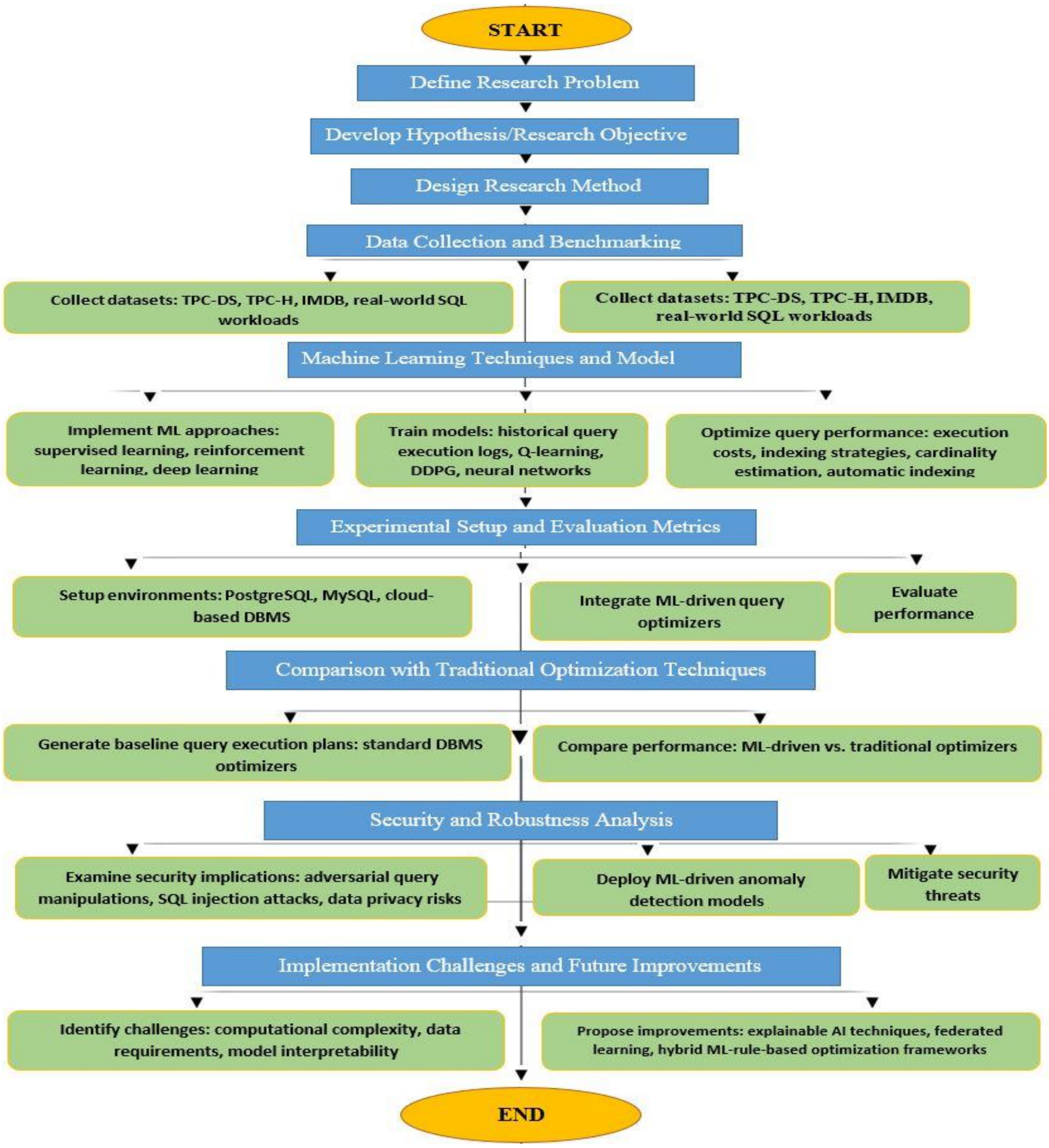

Figure1: General Flowchart of the Methodology

The purpose of this research methodology is to evaluate how well machine learning (ML) techniques can improve query optimization in big databases. To do this, a variety of datasets were used to assess the performance of ML-driven query optimizers, including TPC-DS, TPC-H, IMDB, and real-world SQL workloads. These datasets offer a wide variety of queries with different degrees of difficulty, guaranteeing a thorough examination of the effects of machine learning-based methods.

2.2. Machine Learning Techniques and Model Selection

A number of machine learning techniques, including as deep learning models, reinforcement learning, and supervised learning, were used to maximize query performance. In order to forecast execution costs and suggest effective indexing techniques, supervised learning models were trained using past query execution logs. In order to automatically improve query execution plans according to workload patterns, reinforcement learning methods like Q-learning and Deep Deterministic Policy Gradient (DDPG) were utilized. In order to increase query execution performance, deep learning methods—such as neural networks—were also used for autonomous indexing and cardinality estimation.

2.3. Experimental Setup and Evaluation Metrics

The experimental setup consisted of **PostgreSQL, MySQL, and cloud-based DBMS environments** where ML-driven query optimizers were integrated. Performance evaluations were conducted based on the following key metrics:

- **Query Execution Time** – The time taken to execute queries before and after applying ML-based optimization.
- **Throughput Improvement** – The number of queries processed per second in optimized vs. non-optimized environments.
- **Indexing Efficiency** – The impact of ML-driven indexing on query retrieval performance.
- **Computational Overhead** – The additional processing cost introduced by ML integration in query optimization.

2.4. Comparison with Traditional Optimization Techniques

Comparisons with conventional cost-based and rule-based optimization strategies were conducted in order to evaluate the efficacy of ML-based approaches. Standard DBMS optimizers were used to create baseline query execution plans, and ML-driven optimizers were used to compare how well they performed.

2.5. Security and Robustness Analysis

Additionally, the study looked at the security implications of ML-based query optimization, specifically in relation to data privacy issues, SQL injection attacks, and hostile query manipulations. To identify and reduce security risks in query execution, ML-driven anomaly detection models were implemented.

2.6. Implementation Challenges and Future Improvements

Computational complexity, data needs for machine learning training, and model interpretability are among the difficulties this study uncovered. In order to improve flexibility and security in database administration, future developments will concentrate on creating explainable AI methods, federated learning for privacy-preserving optimization, and hybrid ML-rule-based optimization frameworks.

3. THEORETICAL FRAMEWORK

3.1. Machine Learning-based Query Optimization

Through the development of predictive models that enhance execution strategies and automate performance adjustment, machine learning has completely transformed query optimization. While ML-driven approaches dynamically modify indexing, join ordering, and query scheduling depending on workload trends, traditional rule-based optimizers frequently fall short in the face of complex, changing workloads (Saleh & Zebari, 2025a). By learning from past executions and modifying database setups, reinforcement learning models provide ongoing query performance improvement (W. merza Eido & Yasin, 2025). Furthermore, cloud contexts, where dispersed data processing calls for flexible and scalable solutions, have seen the effective use of ML-based optimizers (Tato & Yasin, 2025).

3.2. Database Performance Enhancement

Enhancing database performance is essential to effectively executing queries. Although machine learning has improved these tactics, traditional methods like indexing, caching, and partitioning are still important for cutting down on query execution time (Alamu, n.d.). ML models can forecast query workloads and optimize resource allocation by examining historical query data, which lowers latency and increases throughput (Hamza Akhtar et al., 2025). Workload-aware optimizations and automated query rewriting guarantee that performance is high even when database conditions vary (Dou et al., 2023).

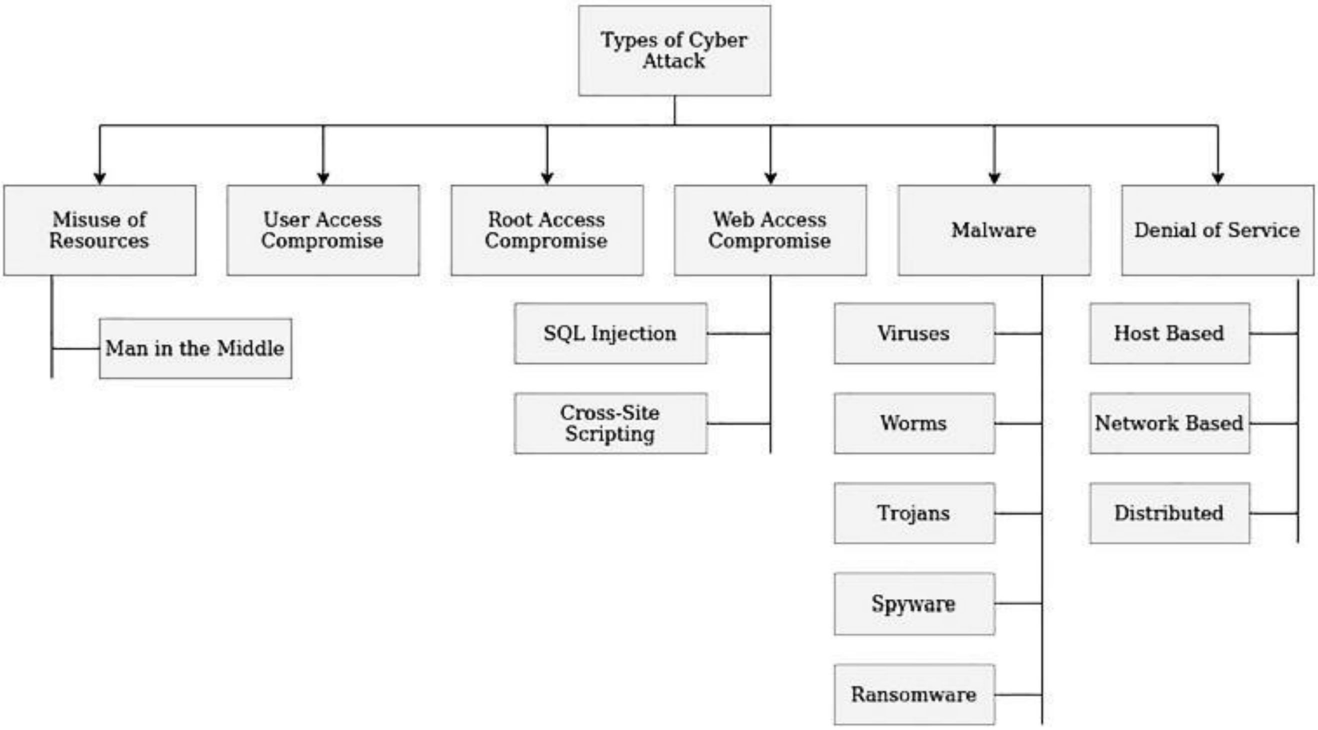

Figure 3: Attacks on databases using contemporary cyber security

3.3. Reinforcement Learning in Databases

An AI technique called reinforcement learning (RL) uses models that constantly learn from past performance to optimize in real time. By dynamically modifying indexing, caching, and scheduling policies, RL-based query optimizers in databases enhance execution methods (M.M.F. Fahima et al., 2024). By making data-driven judgments customized for each unique query workload, these methods beat conventional static optimizers (Islam, 2024). In cloud contexts, where real-time adaptation is required to accommodate changing query loads, RL-powered optimizers are very helpful (Guo et al., 2025).

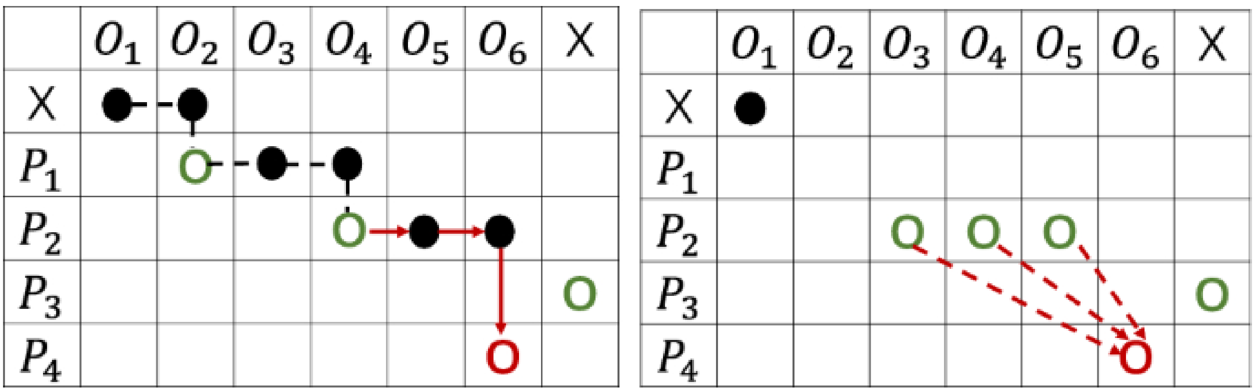

Figure 3 : In-database query optimization on SQL with ML predicates

3.4. Big Data and Cloud-based Optimization

Cloud infrastructures and big data systems provide particular difficulties for query optimization. To optimize efficiency, cloud-based query processing necessitates clever workload balancing due to the remote storage and computing resources involved (Saleh & Zebari, 2025a). By anticipating resource requirements and adjusting processing strategies appropriately, AI-driven methods provide adaptive query execution (Tato & Yasin, 2025). Workload-aware scheduling and federated learning are two strategies that provide excellent availability and scalability while lowering expenses (Dou et al., 2023).

3.5. AI-driven Indexing Strategies

One of the most important aspects of database speed is indexing. By examining query patterns and workload parameters, AI-driven indexing automates the process of choosing and maintaining indexes (Hamza Akhtar et al., 2025). Through dynamic index structure adjustments based on real-time performance data, machine learning-based indexing algorithms have been demonstrated to drastically reduce query execution times (Alamu, n.d.). Furthermore, as database usage patterns change, self-learning indexing systems continuously improve index configurations to guarantee peak performance (Islam, 2024).

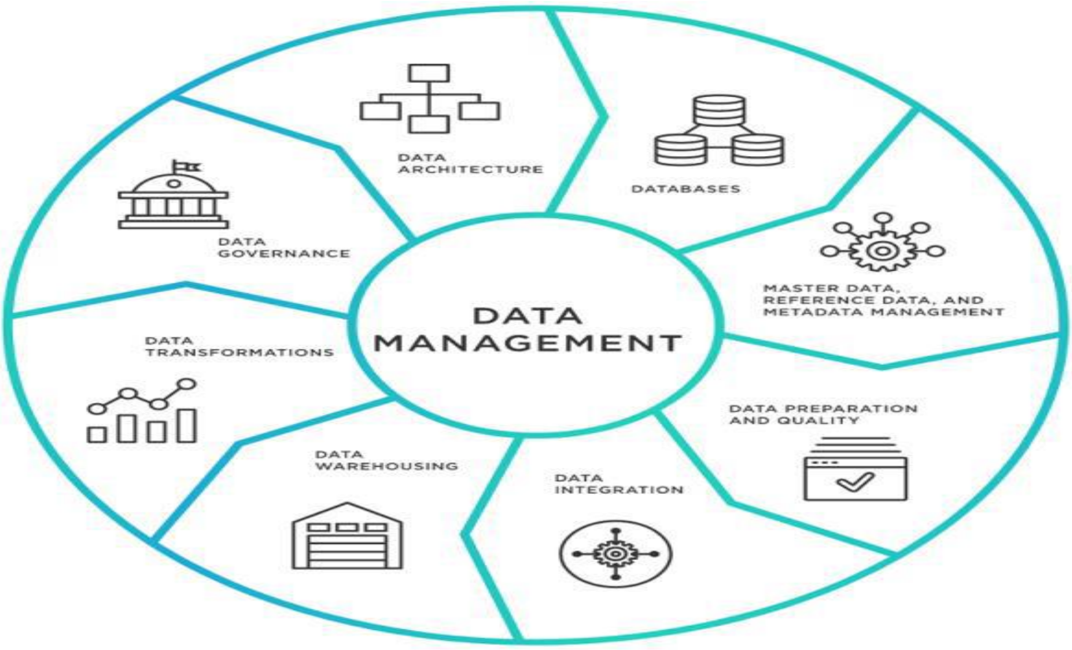

Figure 4: Combining Big Data Frameworks with SQL

3.6. Deep Learning for Query Execution

By discovering intricate patterns in data access and retrieval, deep learning models have demonstrated potential in improving query execution (Guo et al., 2025). Cardinality estimation is a crucial stage in query planning that neural networks can improve, resulting in more precise execution methods (Saleh & Zebari, 2025a). Additionally, transformer-based models have been used to increase the comprehension and execution efficiency of SQL queries, especially in natural language processing (W. M. Eido & Ibrahim, 2025).

3.7. Autonomous Database Management

Without human assistance, autonomous databases use artificial intelligence to optimize, repair, and secure themselves (Saleh & Yasin, 2025). To anticipate and stop performance deterioration, these systems use artificial intelligence (AI) methods including anomaly detection and reinforcement learning (Tato & Yasin, 2025). AI integration in database administration guarantees real-time workload adaptability, resulting in increased security, scalability, and efficiency (Dou et al., 2023).

3.8. Cost-based vs. Machine Learning-based Query Optimization

Conventional cost-based query optimizers select execution plans using predetermined heuristics and statistical estimations, but these methods frequently fall short when dealing with complex queries and dynamic workloads (Saleh & Zebari, 2025a). In contrast, machine learning-based query optimization makes use of past query execution data to forecast and suggest the best query plans (W. merza Eido & Yasin, 2025). Databases can adaptively modify execution techniques thanks to deep neural networks and reinforcement learning, which lowers computing overhead and boosts efficiency (Tato & Yasin, 2025). In cloud-based and distributed database contexts where workload changes need real-time decision-making, these ML-driven models perform noticeably better than conventional optimizers (Guo et al., 2025).

3.9. Neural Networks for Cardinality Estimation

A critical element in query planning is cardinality estimation, which establishes how many rows are processed at various execution phases. Outdated statistics or skewed data distributions frequently cause traditional estimating techniques to be inaccurate (Dou et al., 2023). Cardinality estimate has benefited from the use of deep learning models, such as convolutional and recurrent neural networks, which increase accuracy and decrease execution plan mistakes (Islam, 2024). Compared to conventional histogram-based techniques, these models are able to produce more accurate predictions because they are able to learn intricate data correlations (Saleh & Zebari, 2025a).

3.10. Query Execution Plan Optimization Using Machine Learning

How a database processes and retrieves requested data is determined by query execution plans. Plans are chosen by cost-based optimizers based on projected resource consumption; however, inaccurate cost prediction might result in less-than-ideal performance (W. merza Eido & Zeebaree, 2025). According to (Tato & Yasin, 2025), machine learning-based execution plan optimizers dynamically choose the optimal execution path for a given query by analyzing past performance data and making adjustments in real-time. Through ongoing learning from query workload patterns, reinforcement learning techniques have been very successful in enhancing execution plan selection (Guo et al., 2025).

3.11. AI-Driven Workload Management

In order to guarantee peak performance, database workload management entails distributing query execution among several resources. Machine learning models are used in AI-driven workload management methods to forecast workload trends and proactively assign resources in accordance with them (Hamza Akhtar et al., 2025). In order to ensure that databases manage peak loads effectively and without needless resource consumption, reinforcement learning models dynamically modify CPU, memory, and storage use (Alamu, n.d.). These methods are particularly helpful in cloud contexts where effective resource allocation is required due to pay-per-use pricing models (Dou et al., 2023).

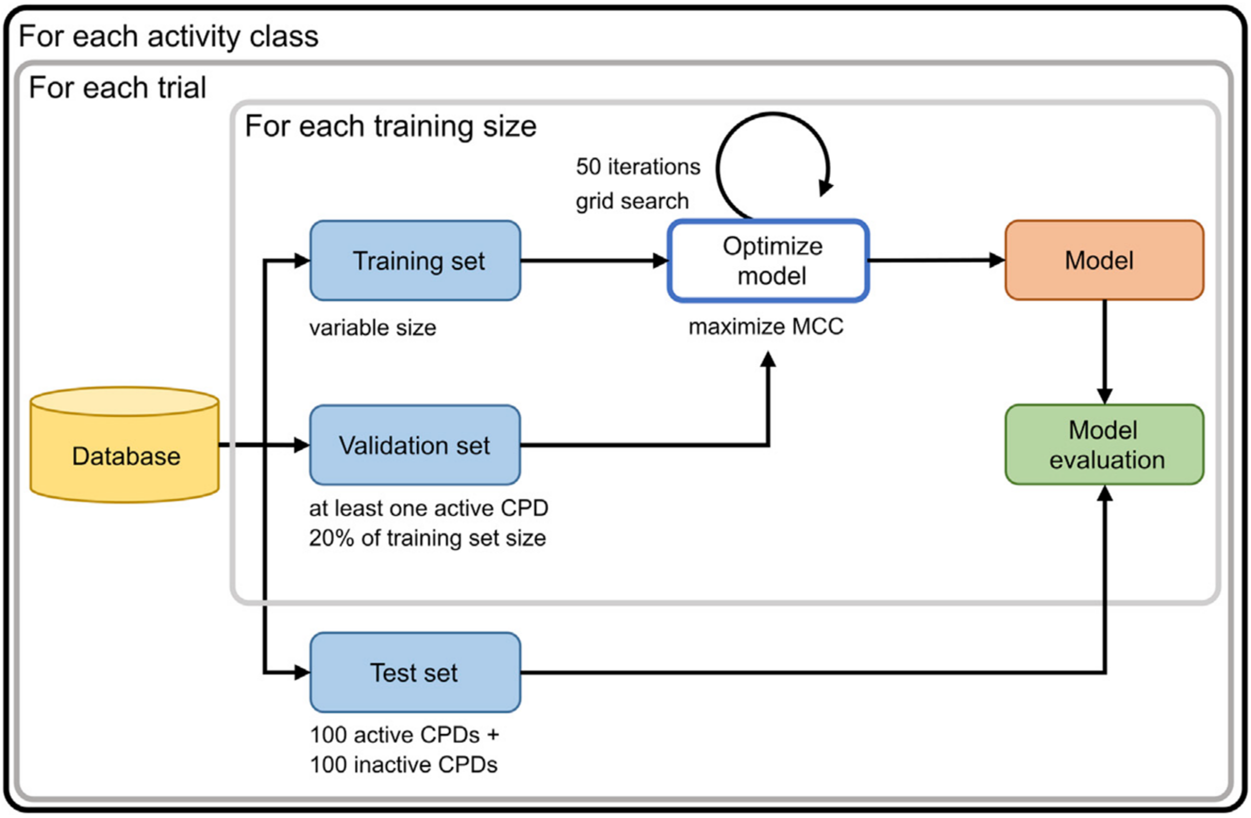

Figure 1. Molecular classification calculation protocol. Eight separate trials using various seeds were conducted for every activity class. A test data set including 100 active and 100 inert substances was selected at random for every experiment. Both training and validation data sets were compiled for every training set size.

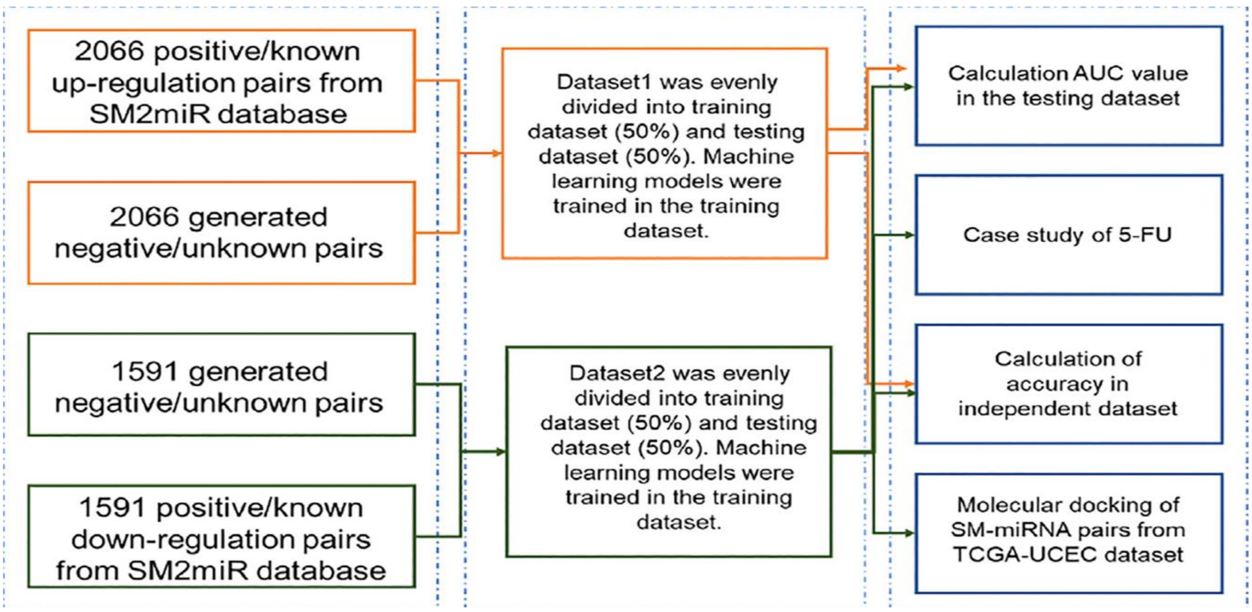

Figure 2. Model development flowchart for predicting SM-miRNA regulation. Models to forecast the upregulated pairings of small molecules and miRNAs were built using data set1. Likewise, models to forecast down-regulation pairs were built using data set2.

3.12. Adaptive Query Caching and Prefetching

By storing frequently accessed data and forecasting future queries based on workload patterns, query caching and prefetching enhance database performance (Saleh & Zebari, 2025a). While ML-based techniques examine query history to find high-value caching opportunities, traditional caching solutions employ preset rules to decide which requests to keep (Eido & Yasin, 2025). Particularly in remote and cloud-based database systems, AI-driven caching and prefetching increase response times and eliminate unnecessary calculations (Tato & Yasin, 2025).

3.13. Security Challenges in AI-Powered Query Optimization

Security issues including data poisoning, adversarial assaults, and unauthorized query execution arise when AI is incorporated into query optimization (Islam, 2024). If ML models are trained on skewed or hacked datasets, they can be manipulated, resulting in harmful or ineffective query execution strategies (Guo et al., 2025). Furthermore, there is rising worry about explainability in AI-driven query optimization since complicated deep learning models frequently lack transparency, which makes it challenging to identify and reduce potential security vulnerabilities (Saleh & Yasin, 2025). To guarantee safe and reliable query optimization in databases, research into explainable AI methods and adversarial-resistant AI models is crucial (Dou et al., 2023).

3.14. Future Trends in Machine Learning for Databases

Automation, effectiveness, and scalability are the main goals of machine learning in database administration going forward. Database performance will be redefined by developments in self-learning databases that may automatically optimize queries, identify anomalies, and modify indexing algorithms (W. merza Eido & Yasin, 2025). Another encouraging trend in query processing is the use of generative AI models, like transformers and large language models (LLMs), which enable automated question rewriting and more natural query interaction (Tato & Yasin, 2025). Furthermore, to enable secure query optimization without disclosing sensitive data, federated learning and privacy-preserving AI approaches are being investigated (Guo et al., 2025).

4. LITERATURE REVIEW

(Abbasi et al., 2024) investigated how PostgreSQL's machine learning (ML) integration may improve workload management and query optimization. The paper suggested a system for dynamically modifying database setups and predicting query execution times that combines supervised and unsupervised learning approaches. With a 74% increase in throughput and a 42% decrease in query execution times, their examination using the TPC-DS benchmark showed notable gains. The direct integration of machine learning models into the database management system, which ensures real-time optimizations without human involvement, was a significant contribution of this research. Notwithstanding these advantages, the authors pointed out difficulties with tuning conflicts and the processing expense of ML integration, suggesting future lines of inquiry to address these problems.

(Ahmadi, n.d.) examined the ways in which machine learning methods enhance cloud data warehousing performance. The study highlighted how ML-driven automation improves workload management, adaptive resource allocation, and indexing, which raises data warehousing efficiency. The trade-off between automation and security was one of the most significant issues found, since a greater dependence on machine learning creates new risks and vulnerabilities. The report highlights future innovations that will further improve data warehouse administration, such as Explainable AI, Automated ML, and Federated Learning. According to the study's findings, incorporating machine learning (ML) into cloud data warehouses has many advantages, but it also presents some security and privacy issues that need to be carefully considered.

(Aken et al., 2017) presented OtterTune, an ML-based tool for autonomous database tuning that uses historical tuning data to choose the best configurations for DBMSs. The study showed how difficult it is to manually change hundreds of database parameters and showed how machine learning (ML) can greatly enhance database performance through automated tweaking. By predicting optimal configurations using supervised and unsupervised learning approaches, OtterTune can achieve latency reductions of 58–94% and cut down on the amount of time needed for database tuning. The analysis of the Vector, Postgres, and MySQL databases demonstrated that ML-driven tuning can improve database performance more effectively than human specialists. The authors stressed the significance of applying knowledge gained from prior tuning attempts to fresh database implementations in order to enable ML models to generalize across various workloads.

(Akhtar & Farzana, 2024) offered a thorough method for optimizing queries in remote databases that makes use of Explainable AI (XAI), Reinforcement Learning (RL), and Autonomous AI. The study showed how AI-driven systems may dramatically improve query performance by managing resources and adjusting configurations on their own without human assistance. Adaptive query optimization, in which the system continuously learns from execution feedback to improve query plans, was made possible by the use of reinforcement learning. By making the decision-making process transparent, XAI made sure that optimization tactics were understandable and in line with corporate goals. The authors came to the conclusion that integrating various AI methods produces a database system that optimizes itself and can effectively manage changing workloads.

(Hamza Akhtar et al., 2025) emphasized the necessity of multi-layered protection strategies while concentrating on ML-driven database security advances. In order to counteract these risks, the paper suggested ML-based anomaly detection systems and identified important vulnerabilities in contemporary databases, such as SQL injection assaults and access control flaws. To build a strong security architecture, their strategy included access control, sophisticated threat detection, and cryptographic protocols. The study emphasized how crucial database security is becoming in hybrid cloud settings, where conventional security strategies can't keep up with changing threats. According to the authors, database management systems can greatly improve protection against cyberattacks while preserving system efficiency by including AI-based security frameworks.

(Alamu, n.d.) examined how deep learning methods may be used into data management, highlighting how they can improve query optimization and data architecture. The study demonstrated how deep neural networks, convolutional neural networks, and recurrent neural networks enhance indexing, automate feature extraction, and use predictive modeling to optimize query execution. The study also looked at query planning powered by reinforcement learning, which dynamically adjusts to changes in workload to improve compute efficiency and retrieval speed. The study also covered how deep learning helps with anomaly detection, data integrity, and avoiding anomalies in contexts with vast amounts of data. Although deep learning greatly improves database performance, the author came to the conclusion that more research is necessary to address difficulties like high computational overhead and interpretability.

(Wisam Altaher & Hasan Hussein, n.d.) compared typical query optimization techniques in database management systems with those based on machine learning. ML-based optimization greatly increases query execution efficiency, according to the study, which evaluated the efficacy, flexibility, and mathematical modeling of various strategies. The results showed that while cost-based optimization is still useful in some static settings, ML-driven approaches are more flexible when it comes to real-time data processing and dynamic workload conditions. Models for supervised learning and reinforcement learning were emphasized as viable strategies for cutting down on execution time and maximizing resource utilization. The author underlined that even with their benefits, machine learning techniques need careful integration with current database infrastructures and a large amount of data for training.

(Ashlam et al., 2022) centered on using machine learning to identify SQL injection (SQLi) attacks, which pose a serious risk to database systems' cybersecurity. The paper presented a novel method for classifying possible SQLi attacks by first extracting characteristics from SQL queries using CountVectorizer and then utilizing supervised learning models. The study created the PALOSDM model, which improved accuracy from 94% to 99% while drastically lowering false positive and false negative rates. Given the expanding complexity of cyberattacks, the author emphasized the growing significance of automated threat detection in database security. The study came to the conclusion that although machine learning models offer excellent detection accuracy, practical application needs to strike a compromise between computing efficiency and threat adaptation.

(Azhir et al., 2019) examined cloud-based query optimization techniques in a methodical manner, classifying them into search-, machine learning-, schema-, and security-based strategies. Because distributed data storage results in a large number of possible query execution plans, the study emphasized the scalability issues that cloud databases confront. When compared to conventional static optimization strategies, machine learning-based optimization techniques were found to be more effective in managing dynamic workload fluctuations and enhancing execution efficiency. The study also looked at rule-based and cost-based optimization techniques, pointing out that static methods frequently don't work in cloud systems that are extremely dynamic. Future studies on hybrid optimization approaches, which integrate ML models with conventional cost-based tactics for improved performance, were suggested by the author.

(Chandramouli et al., n.d.) investigated query optimization methods for multimedia databases, incorporating user relevance input and query refining for contextualized picture retrieval. To improve picture search accuracy, the study suggested a methodology that makes use of particle swarm optimization, k-means clustering, and hypernym identification based on Wikipedia. According to the author, retrieval accuracy is much increased and user search ambiguity is decreased when queries are refined based on contextual awareness. Furthermore, it was demonstrated that supervised classification-based relevance feedback systems may further hone search results according to user preferences. According to the study's findings, adding query optimization strategies to multimedia databases improves user experience but necessitates more developments in machine learning-driven semantic comprehension.

(Choi & Lim, 2020) investigated the use of machine learning techniques to optimize targeted advertising by classifying strategies into two categories: content-centric and user-centric. The study highlighted the importance of ML-based techniques in improving audience targeting and cutting down on inefficiencies by identifying 23 distinct ML-based online advertising tactics. The discovery of an unexplored field—algorithm-based detection of click fraud—to guarantee the integrity of digital advertising initiatives was one of the major contributions. The study showed that using machine learning (ML) for behavioral targeting and user profiling greatly increased ad engagement rates and marketers' return on investment. The author came to the conclusion that although ML-driven advertising strategies have many benefits, issues like algorithmic biases and data privacy issues need more research.

(Thirupurasundari et al., 2023) proposed an adaptive ML-based query optimizer after examining the function of machine learning in maximizing query performance in big data systems. In order to forecast the best query techniques, the study presented a two-component methodology in which machine learning models were trained on past query execution data. In comparison to conventional rule-based optimization methods, experimental results showed that the ML-based optimizer enhanced query performance by as much as 60%. The study also demonstrated the approach's flexibility and efficacy in managing different data distributions and query patterns. With the possibility for additional improvement through deep learning and reinforcement learning approaches, the authors stressed that the incorporation of machine learning into big data query processing constitutes a fundamental shift in database administration.

(Ding et al., 2019) examined how using ML-driven classifiers in place of conventional optimizer cost estimations might improve index suggestion in database management systems. According to the study, automated indexing is inefficient because traditional indexing methods frequently underestimate the expenses associated with query execution. The suggested ML model greatly increased accuracy in suggesting ideal index configurations by rephrasing the indexing decision process as a classification problem as opposed to a cost-estimation assignment. According to experimental evaluations, this method significantly eliminated performance regressions brought on by erroneous optimizer estimations by reducing indexing mistakes by up to five times. The authors came to the conclusion that database speed and reliability can be greatly improved by including ML-driven classifiers into automated indexing systems.

(Dong et al., 2024) introduced MLETune, a cutting-edge machine learning method for database knob tuning that makes use of multiple large language models (LLMs) to facilitate deep reinforcement learning. The study emphasized the difficulty of effectively adjusting database parameters since manual tuning is laborious and complicated because database knobs are high-dimensional. MLETune greatly sped up the tuning process by using a genetic algorithm for preliminary exploration and reinforcement learning for fine-grained optimization. According to experimental data, MLETune performs better than current tuning techniques, increasing system performance by up to 26% and increasing efficiency by six times. The authors came to the conclusion that combining reinforcement learning with LLMs presents a viable approach to automatic database performance optimization.

(Dou et al., 2023) reviewed different machine learning and deep learning approaches tailored for tiny datasets, with an emphasis on machine learning approaches for tackling small data difficulties in molecular science. With an emphasis on their uses in molecular chemistry and biological sciences, the study divided techniques into fundamental machine learning algorithms, artificial neural networks, convolutional neural networks, transfer learning, and reinforcement learning. The authors stressed that tiny datasets present particular difficulties such imbalance, noise, and data diversity, necessitating the use of specific machine learning approaches for efficient analysis. The study also looked into physical model-based data augmentation to enhance model generalization and hybrid approaches that combine deep learning and conventional machine learning. The authors came to the conclusion that although advances in machine learning present encouraging answers to problems involving limited data, more study is required to improve the efficiency and interpretability of the models.

(Du et al., 2020) examined four cutting-edge machine learning techniques for managing geographical data: support vector machines (SVM), semi-supervised learning, ensemble learning, and deep learning. The study demonstrated how SVM-based kernel learning has outperformed conventional statistical models in tasks involving regression and categorization of geographical data. The problem of tiny training datasets was addressed by semi-supervised and active learning techniques, which enable models to learn from a small number of labeled samples while increasing accuracy. The study also highlighted how ensemble learning can improve model generalization by integrating several classifiers, which makes it an effective tool for prediction and spatial interpolation. In conclusion, the importance of deep learning algorithms in the categorization of high-resolution remote sensing images was explored, showcasing their exceptional performance in intricate spatial data analysis.

(During, n.d.) investigated improved query optimization in distributed databases with an emphasis on enhancing the efficiency and reliability of the supply chain. The study looked at how machine learning models and other clever algorithms might reduce query response times and enhance resource usage in distributed database systems. Data segmentation, indexing, and query rewriting were among the main optimization strategies covered; these strategies are essential for minimizing latency and increasing throughput. In order to dynamically optimize query execution plans and guarantee adaptable answers to shifting workloads, predictive analytics was found to be a key component. According to the study's findings, incorporating redundancy techniques and fault-tolerant systems improves database resilience and permits continuous access to vital supply chain data.

(M.M.F. Fahima et al., 2024) examined how machine learning can improve query processing and database administration, highlighting its contribution to the modernization of conventional database systems. The study examined many machine learning methods for workload management, indexing, and query optimization, including deep learning, reinforcement learning, and natural language processing (NLP). It demonstrated how machine learning models can adjust dynamically to shifting data conditions, improving system efficiency and query execution speed. The study also highlighted the value of machine learning (ML) in anomaly detection, maintaining database integrity, and enhancing security protocols. The authors came to the conclusion that database systems powered by machine learning perform noticeably better than conventional ones, providing increased effectiveness and flexibility to meet contemporary data difficulties.

(Gadde, 2022) examined how artificial intelligence (AI) might improve database optimization techniques by being incorporated into SQL query processing. Important issues like algorithm selection, data quality, and the requirement for efficient integration frameworks for AI-enhanced SQL systems were noted in the study. In order to increase execution times and resource use, it suggested a hybrid strategy that blends machine learning algorithms with conventional rule-based query optimization. Results from experiments showed that AI-driven SQL processing could automate indexing and drastically cut down on query execution times, resulting in more effective database administration. Although AI offers enormous promise for optimizing SQL queries, the study came to the conclusion that more investigation is required to resolve integration issues and improve AI-driven database management techniques.

(Guo et al., 2025) introduced SQL+ML, a revolutionary method for optimizing in-database queries by incorporating machine learning (ML) predicates into SQL. The computational overhead of processing ML-based queries was one of the main issues that the study addressed in order to optimize ML predicates within database management systems. By producing SQL predicates based on ML predicates, the study's Smart framework improves SQL+ML query processing, lowering the quantity of pointless tuples processed and greatly increasing performance. According to experimental results, Smart performed up to three orders of magnitude better on benchmark datasets than current query optimization strategies. The authors came to the conclusion that, by removing the requirement for expensive data transfers, optimizing ML predicates directly within database systems offers a more effective and secure substitute for conventional ML processing techniques.

(Hasan et al., 2020) investigated deep learning models to solve the problem of precisely forecasting query result sizes for selectivity estimation in multi-attribute queries. The study suggested two new methods: one that uses supervised learning to predict selectivity values and another that uses neural density estimation to capture joint probability distributions. Deep learning considerably outperformed conventional histogram and sampling-based estimation methods, according to experimental assessments, especially for queries with multiple predicates and low selectivity. Practical issues like featurization and integrating query workload dynamics into deep learning frameworks were also covered in the study. The authors came to the conclusion that although deep learning provides increased efficiency and accuracy, more research on flexibility and interpretability is necessary before it can be used practically in real-world database management systems.

(Aoun, n.d.) investigated query efficiency in several big data settings and suggested sophisticated query processing methods to maximize the retrieval of information. The study emphasized the necessity for adaptive optimization techniques by highlighting the effects of user behavior, indexing schemes, and data diversity on query efficiency. To improve query speed across many data sources, a thorough framework combining distributed systems, machine learning, and parallel computing was presented. According to empirical data, these methods greatly increase reaction times and relevance ranking, which is advantageous for academic researchers, corporations, and policymakers. The authors came to the conclusion that improving query efficiency in diverse settings necessitates a multifaceted strategy that includes both technological and user-centered design components.

(Islam, 2024) examined the effects of artificial intelligence (AI) and machine learning (ML) on big data analytics and SQL databases, highlighting significant developments and trends. The paper examined previous research and demonstrated how ML-driven query optimization improves efficiency in dynamic data contexts and cuts down on processing times. In order to improve accuracy and enable automated workflows, a significant emphasis was focused on integrating machine learning models into SQL databases for real-time predictive analytics. The efficiency of AI-based security measures in proactive threat detection and data protection was also examined. According to the study's findings, data management is being revolutionized by the incorporation of ML and AI into SQL databases, which calls for more research on hybrid database architectures for efficiency and scalability.

(Jambigi et al., 2024) proposed a machine learning architecture to improve regression testing in SAP HANA and investigated SQL-based root cause analysis for database workload problems. The study tackled the problem of false positive errors in workload replays, which are brought on by infrastructure variability, concurrency problems, and data privacy restrictions. A large language model (LLM) was used to generate succinct failure summaries from SQL logs, improving interpretability and aiding in failure classification. According to experimental assessments, the F1-Macro score improved by 4.77%, increasing the approach's dependability in actual database testing settings. The authors came to the conclusion that database managers and software developers can gain from the substantial improvement in error analysis that can be achieved by incorporating ML and LLMs into database workload replays.

(Kaur et al., 2018) focused on the effects of big data and machine learning on predictive analytics and data-driven disease diagnostics in safe healthcare systems. Big data storage, security measures, machine learning-based predictive analytics, and encrypted patient data access restrictions were all integrated into the study's innovative four-layer healthcare information system. Data privacy was ensured by implementing a number of security measures, such as endpoint validation, granular access control, and masking encryption. According to the study, integrating machine learning with safe big data frameworks improves healthcare decision-making, resulting in more precise diagnoses and effective patient care. In order to balance efficiency and data privacy, the authors came to the conclusion that ML-driven analytics must be incorporated into future healthcare systems together with robust security measures.

(Marcus et al., 2021) presented SageDB, a learnt database system that optimizes query execution and storage structures by incorporating machine learning (ML) models into all database components. The study showed that databases may greatly enhance performance by using trained models to customize execution plans according on workload and data distribution. In order to reduce query execution time and storage overhead, the authors emphasized that ML-driven indexing and query optimization can take the role of conventional rule-based optimizers. The idea of autonomous code synthesis, which allows databases to dynamically adjust to changing workloads without operator intervention, was a significant contribution of this study. The study came to the conclusion that although learning database systems offer significant efficiency advantages, more research is necessary to address issues with generalization, model training costs, and integration into current systems.

(J. Zhang et al., 2019) Introducing QTune, a query-aware database tuning system that optimizes database setups using deep reinforcement learning (DRL). The study introduced an automated system that learns from query workloads and dynamically modifies database parameters in order to overcome the drawbacks of conventional knob-tuning techniques. Comparing QTune to state-of-the-art tuning techniques, experimental results showed that it greatly improves query performance, achieving higher throughput and reduced latency. The authors presented a brand-new Double-State Deep Deterministic Policy Gradient (DS-DDPG) model that effectively strikes a compromise between tweaking at the query, workload, and cluster levels. Although there are still issues with real-world deployment, the study found that DRL-based database tuning is a viable way to maximize database performance with little human involvement.

(X. Li et al., 2022) examined the use of deep learning (DL) in large data analysis for smart cities, with a particular emphasis on digital twins (DTs) and the Internet of Things (IoT). In order to improve IoT-based data processing, the study presented a distributed parallel technique for convolutional neural networks (CNNs), which lowers data transmission latency and increases predictive accuracy. By facilitating real-time data analysis and automated decision-making, the authors showed how DL combined with big data analytics greatly improves the effectiveness of smart city government. One important discovery was that multi-hop transmission technology enhances the performance of IoT data transfer, guaranteeing safe and effective data exchange in smart city settings. The study came to the conclusion that DL-driven IoT analytics are essential for promoting the growth of smart cities and laying the groundwork for intelligent urban infrastructure in the future.

(BRIDGING DATA MANAGEMENT AND MACHINE LEARNING: CASE STUDIES ON INDEX, QUERY OPTIMIZATION, AND DATA ACQUISITION, n.d.) presented case studies on index optimization, query optimization, and data collection as they examined the relationship between machine learning and data management. According to the study, ML-based query optimizers perform better than conventional cost-based optimizers because they can learn from historical query execution patterns and adjust dynamically to changes in workload. The study presented novel methods for ML-driven indexing that preserve effective query retrieval times while lowering storage overhead. The creation of ML-based data acquisition techniques that increase model training effectiveness while lowering data collection expenses was a major contribution. Although issues with model interpretability and deployment complexity need to be resolved, the study found that incorporating machine learning (ML) into database administration results in more effective and flexible solutions.

(M.-L. Li, 2023) offered an AI-powered framework with an emphasis on indexing and query optimization for automatic database performance tweaking. In order to minimize manual labor and increase query execution efficiency, the study suggested a machine learning-based system that continuously learns from workload patterns to optimize indexing algorithms. In contrast to conventional heuristic-based techniques, the study demonstrated notable gains in database performance by introducing reinforcement learning and deep neural networks for adaptive query optimization. According to experimental findings, AI-driven tuning improves scalability in high-load scenarios, optimizes resource usage, and speeds up query response times. According to the study's findings, AI-driven database tuning is a significant development in database administration; however, further work is required to improve flexibility and lower computational cost.

(Ma & Triantafillou, 2019) presented DBEst, an approximate query processing (AQP) engine powered by machine learning that improves query efficiency in big data analytics. In contrast to conventional AQP systems, the study suggested approximating analytical inquiries using regression models and probability density estimators, which would greatly cut down on response times and processing expenses. According to the experimental results, DBEst fared better than cutting-edge AQP engines like BlinkDB and VerdictDB, reaching orders of magnitude faster query execution times without sacrificing accuracy. The capacity of ML models to generalize from small data samples, guaranteeing low memory overhead while providing quick and accurate analytics, was a significant contribution of this study. Although additional research is required to extend their applicability to more complicated question types, the study found that ML-based AQP engines are a good substitute for conventional query execution techniques.

(Marcus et al., 2018, 2021) presented Neo, a query optimizer based on deep learning that uses neural networks to automatically discover effective execution strategies. A reinforcement learning-based alternative that can continuously improve itself was suggested by the study, which also emphasized the drawbacks of conventional query optimizers, which depend on manually constructed rules and laborious manual tuning. According to experimental tests, Neo's performance was on par with or even better than that of commercial query optimizers like as those made by Microsoft and Oracle for specific workloads. Neo stood out for its capacity to learn from previous execution patterns and adjust dynamically to shifts in workload and data distributions. The authors came to the conclusion that, despite ongoing difficulties with explainability and interoperability with current systems, trained query optimizers offer a promising path for database administration.

(Marcus et al., 2021) presented Bao, an AI-powered query optimizer that overcomes obstacles like high training costs and lack of adaptability to make learnt query optimization feasible. Unlike earlier learning-based optimizers, Bao did not completely replace the conventional query optimizer; instead, it used Thompson sampling in conjunction with reinforcement learning to provide per-query optimization tips. The study showed that, especially in cloud database setups, Bao greatly decreased query execution time and increased cost-efficiency. Bao's capacity to learn and adjust to changing workloads was a major advantage, guaranteeing that query strategies would always be the best. The authors came to the conclusion that while AI-assisted query optimization provides a scalable solution for contemporary database systems, more effort is needed to improve its decision-making procedure and compatibility with current optimizers.

(Meduri et al., 2021) investigated the possibility of using machine learning models—more especially, Q-learning and recurrent neural networks (RNNs)—to forecast the subsequent SQL query in a running user session. In order to facilitate speculative query execution and prefetching for improved database interactivity, the study suggested a novel method that used previous query patterns to predict future questions. Experimental comparisons with conventional query suggestion techniques showed that temporal predictors, such as RNNs and reinforcement learning models, performed more accurately and efficiently than collaborative filtering techniques. The creation of schema-aware SQL embeddings, which enhanced query representation and prediction accuracy, was one of the study's major contributions. The authors came to the conclusion that, despite ongoing issues with scalability and real-time learning, query prediction models can greatly increase database responsiveness.

(Milicevic & Babovic, 2024) categorized methods into query optimization, index architecture, and parameter tweaking after conducting a thorough analysis of deep learning applications in database query execution. The study highlighted significant developments in automatic indexing, deep learning-based cardinality estimation, and query plan enumeration and their effects on increasing the efficiency of query execution. The function of learnt index structures, which use neural networks to replace conventional B-tree and hash-based indexing systems and provide quicker lookup speeds and less storage space, was a key area of study. The paper also covered the difficulties of integrating deep learning models into actual database systems, such as integration complexity, processing overhead, and model interpretability. Although deep learning has the potential to completely transform database query execution, the authors came to the conclusion that major implementation challenges must be removed before deep learning can be widely adopted.

(Mitchell & Nelson, 2021) explored the ways in which machine learning improves query processing in big data analytics, with an emphasis on themes such natural language processing, predictive analytics, and query optimization. The study stressed that standard query processing approaches struggle to address the increased complexity of large data settings, necessitating adaptive machine learning techniques. The examination of automated data discovery techniques, which employ machine learning to find hidden patterns in datasets without the need for human intervention, was a significant contribution. The authors showed how adding machine learning to query processing greatly increases system scalability, overall efficiency, and data retrieval speed. According to the study's findings, big data analytics will be further improved by ongoing developments in machine learning algorithms, which will make searches more flexible and sensitive to changing data conditions.

(Oluwafemi Oloruntoba, 2025) examined AI-powered self-management of databases with a focus on intelligent indexing, self-tuning, and predictive query optimization. According to the study, self-tuning mechanisms driven by AI dynamically modify system parameters to maximize performance, eliminating the need for manual database administration. A key AI-driven feature that uses deep learning algorithms to foresee performance problems and improve query execution strategies is predictive query optimization. Intelligent indexing was also established by the study, in which database indexes are automatically maintained and adjusted by machine learning models to maximize the effectiveness of data retrieval. The study found that while security and regulatory issues need to be resolved, AI-driven autonomous databases greatly improve operational efficiency and reliability.

(Panwar, 2024) studied advanced stored procedures with an emphasis on query efficiency and speed enhancements as a way to maximize big data processing in SQL Server systems. The study showed that by precompiling intricate SQL queries, stored procedures improve scalability, maintainability, and execution speed. Reducing query execution time through parallel processing and containerization, which improved database burden distribution, was one of the main benefits found. Additionally, the study examined indexing strategies in stored procedures, emphasizing how they might minimize query bottlenecks and improve data retrieval. Although its implementation necessitates careful optimization tactics, the author came to the conclusion that advanced stored procedures provide a cost-effective alternative for managing large-scale data processing.

(Patel et al., 2020) outlined how AI-driven models speed up lead identification and molecular screening in the context of drug discovery review. In order to predict chemical and biological interactions in drug development, the study looked at a number of machine learning (ML) techniques, such as support vector machines, random forests, and deep learning. One significant breakthrough was the use of machine learning into high-throughput screening, which enhanced the precision and effectiveness of drug candidate discovery. In order to enable more accurate drug-target predictions, the scientists also investigated how machine learning (ML) models improve structure-activity relationship (SAR) studies. Although machine learning greatly improves the efficiency of drug development, the study found that issues like data quality and algorithm interpretability need more investigation.

(Polkowski et al., 2021) examined query optimization with computational and evolutionary methods in virtual database environments. The inefficiencies of static query execution plans were examined in the study, and dynamic optimization strategies that adjust to changes in workload were suggested. One significant advancement was the application of heuristic methods to optimize query execution plans in remote virtual environments, which improved response times and decreased processing costs. The authors also discussed how to prioritize complicated heterogeneous queries, highlighting how global learning coefficients help optimize optimization techniques. The study found that while adaptive query optimization techniques boost database performance, further work is required to increase scalability and security in cloud-based settings.

(Rachakatla & Machireddy, n.d.) examined how machine learning may improve data integration and query optimization in data warehousing. The study emphasized the shortcomings of conventional data integration techniques, which find it difficult to effectively combine disparate data sources into a single repository. The study showed how using machine learning may significantly enhance data translation, anomaly detection, and schema matching while lowering manual labor and integration process errors. The study also looked at how machine learning affected query optimization, with models using deep learning and reinforcement learning dynamically modifying execution plans to improve performance. Although issues like computational cost and model interpretability necessitate more study, the authors came to the conclusion that machine learning offers a revolutionary answer for data warehousing.

(Rahman et al., 2024) emphasized multi-level indexing, query rewriting, and dynamic execution plans while concentrating on sophisticated query optimization strategies in SQL databases for real-time big data analytics. The study found that the size and complexity of contemporary big data settings cannot be handled by conventional SQL query optimization techniques, such as heuristic-based approaches. Multi-level indexing and machine learning-driven query rewriting were used to significantly increase resource utilization by cutting query execution times by about 40%. The study also demonstrated how these methods can be tailored to various query complexity levels, which makes them appropriate for dynamic data contexts. The authors came to the conclusion that managing massive, real-time data processing can be effectively accomplished by using machine learning into SQL query optimization.

(Ramu, 2023) studied several methods for improving database speed, with an emphasis on resource usage and effective query execution. To increase response times and system performance, the study examined a variety of optimization strategies, such as query rewriting, indexing, caching, and parallel processing. The study showed how database administrators can optimize execution plans to reduce computational overhead and improve overall efficiency by utilizing these techniques. The study also covered how database schema design affects performance, emphasizing recommended practices such data segmentation, normalization, and denormalization. The authors came to the conclusion that efficient database optimization techniques guarantee scalability and responsiveness in data-intensive applications while also greatly increasing system efficiency.

(Singh et al., 2025) evaluated several quantum machine learning algorithms for classifying big data, paying particular attention to how effective and applicable they were for large datasets. A new quantum-enhanced support vector machine (QeSVM) was presented in the study, and it outperformed conventional machine learning models in managing challenging categorization problems. Results from experiments demonstrated that quantum algorithms performed faster and more accurately, especially in high-dimensional data environments. In order to improve model performance, the study also investigated hybrid quantum-classical methodologies, which mix quantum computers with classical optimization techniques. Although there are still issues with scalability and quantum hardware restrictions, the authors came to the conclusion that quantum computing has the potential to completely transform big data classification.

(Sundaram et al., 2023) suggested a firefly technique for near-duplicate picture recognition that is combined with highly connected deep learning networks. The study emphasized how crucial it is to spot nearly identical photos for use in copyright enforcement, content management, and search engine optimization. The suggested approach greatly increased the accuracy of picture similarity recognition by fusing DenseNet for feature extraction with the firefly algorithm for hyperparameter tweaking. The framework performed better than conventional image detection models, according to experimental results, especially in challenging visual pattern recognition tasks. The authors came to the conclusion that improving near-duplicate picture identification performance can be achieved by the integration of evolutionary techniques with deep learning.

(Hayath et al., 2023) offered a thorough analysis of SQL optimization strategies meant to enhance query performance in big databases. The study investigated a number of techniques to improve SQL query execution times, such as caching, partitioning, query rewriting, and indexing. The authors stressed that query optimization is essential to database management systems (DBMS), especially when it comes to effectively managing intricate and sizable queries. One of the main conclusions was that by dynamically modifying execution plans in response to workload patterns, machine learning techniques might further enhance SQL optimization. According to the study's findings, query performance can be significantly enhanced by implementing AI-driven optimization approaches, even while conventional optimization techniques are still effective.

(D. Wang et al., 2014) suggested a retrieval-based face annotation system that uses local coordinate coding with weak label regularization to improve the accuracy of facial recognition. Retrieving similar facial photos and improving noisy labeling from online image databases were two issues that the study addressed. To enhance face annotation results, the suggested method combines a sparse reconstruction technique with content-based image retrieval approaches. According to experimental data, the framework improved retrieval precision and decreased labeling errors, which greatly improved annotation performance. The authors came to the conclusion that their method offers a scalable and efficient way to handle face annotation tasks in huge databases of web facial images.

(Z. Wang et al., 2021) examined how big data and machine learning tools are used to optimize heat pipe performance, emphasizing how they might improve energy efficiency. The intricacy of current simulation models and the difficulties of generating precise performance forecasts were two of the major issues in heat pipe technology that the study discovered. The authors suggested a machine learning system trained on large data to optimize operational circumstances and structural parameters for heat pipe applications. According to their findings, machine learning greatly lowers computing expenses while increasing heat transfer modeling's predictive accuracy. According to the study's findings, incorporating machine learning into heat pipe technology presents a viable path toward enhancing scalability and performance.

(Xie et al., 2021) created a machine learning-based method for big data analytics-based demand forecasting for Chinese cruise tourism. In order to increase prediction accuracy, the study implemented a least squares support vector regression model that was tuned using a gravitational search technique. In contrast to conventional statistical models, the authors showed that predicting performance was improved by combining Baidu search query data with economic indicators. According to empirical findings, the suggested methodology greatly decreased forecast errors and enhanced decision-making for investments in the cruise industry. The study came to the conclusion that reliable demand forecasting in the tourism industry requires the use of big data and cutting-edge machine learning techniques.

(J. Zhang et al., 2019) presented CDBTune, a deep reinforcement learning-based end-to-end automatic cloud database tuning system for DBMS configuration optimization. The work used a deep deterministic policy gradient approach to solve the difficulties of fine-tuning high-dimensional setup parameters in cloud databases. According to experimental results, CDBTune fared better at optimizing query performance and lowering latency than both conventional database managers and cutting-edge tuning tools. The study's introduction of a reward-feedback mechanism, which allowed for real-time learning and workload modification, was a major innovation. The authors came to the conclusion that deep reinforcement learning provides an effective and scalable way to automatically tune databases in cloud systems.

(A. Zhang et al., 2022) examined how machine learning applications have changed in the healthcare industry, focusing on the transition from model development to model implementation. According to the study, despite the fact that machine learning has significantly improved medical imaging, diagnostics, and predictive analytics, data limits and integration problems still make practical application difficult. The authors looked at how federated learning and deep generative models might help address privacy issues and data shortages, allowing for wider implementation in healthcare settings. They also talked about how transformer models have emerged to handle massive clinical datasets and increase the effectiveness of healthcare decision-making. According to the study's findings, the emphasis should move from model optimization to enhancing data quality, accessibility, and integration within healthcare systems as machine learning advances from development to implementation.

(Zhou et al., 2022) categorized research into two main areas: databases for AI deployment (DB4AI) and artificial intelligence for database optimization (AI4DB), offering a thorough overview of the nexus between these two fields. In order to improve efficiency in large-scale data management, the study looked at how AI-driven methods like deep learning and reinforcement learning complement conventional database chores like query optimization, indexing, and automatic tuning. On the other hand, by enhancing data governance, model training, and inference speed, the authors investigated how database technology can support AI applications. The study's identification of research issues, such as hybrid AI-DB models, co-optimization techniques, and the incorporation of AI-driven declarative query languages, was a major contribution. According to the study's findings, the combination of AI with database systems signifies a revolutionary change that calls for more investigation into automation, scalability, and interpretability.

5. DISCUSSION AND COMPARISON

Table 1 represents a detailed comparison among the previous works explained in section 3. The table illustrates main metrics that depended for the comparison which are the significant features concluded from these works.

Table 1: Comparison among the reviewed works.

Author name with year	Methods	Datasets	Advantages	Disadvantages	Accuracy	Results
(Hamza Akhtar et al., 2025)	Machine Learning-based Security Models	Hybrid Cloud Security Data	Improved access control and anomaly detection	Improved access control and anomaly detection	Not specified	enhanced database security architecture
(Alamu, n.d.)	Deep Learning for Data Management	Big Data Applications	Automated query optimization and feature extraction	High resource requirements	Not specified	Improved data organization and speed of retrieval
(Wisam Altaher & Hasan Hussein, n.d.)	A Comparison between Conventional Query Optimization and Machine Learning	Database Management Systems	Enhanced flexibility and effectiveness of queries	Integration complexity	Not specified	Enhanced query effectiveness
(Ashlam et al., 2022)	Detecting SQL Injections with Machine Learning	SQL Injection Attack Datasets	A notable decrease in erroneous positives	high overhead in calculation	94%-99%	Improved database security
(Azhir et al., 2019)	Methodical Examination of Cloud Query Optimization	Systems for processing queries in the cloud	Enhanced scalability and efficiency	High requirements for computing and storage	Not specified	thorough evaluation of optimization methods
(Chandramouli et al., n.d.)	Refining Queries for Content-based Search	Wikipedia and Image Databases	More contextual awareness and improved query refinement	restricted reach outside of Wikipedia	Not specified	Improved user query relevance
(Guo et al., 2025)	SQL+ML Optimization	TPC-H, SSB, IMDB	Enhanced in-database ML integration	Complex system integration	97.3% on IMDB, 82.4% on Flight dataset	ML-enhanced SQL query processing
(Rahman et al., 2024)	Query rewriting and	Large SQL Databases	40% faster retrieval, 35%	Requires extensive indexing	Not specified	Enhanced efficiency in

	multi-level indexing		faster execution			query execution
(Akhtar & Farzana, 2024)	Security Models Based on Machine Learning	Hybrid Cloud Security Data	Improved access control and anomaly detection	High computational cost and difficult implementation	Not specified	enhanced database security architecture
(Choi & Lim, 2020)	Machine Learning for Target Advertising	Online Advertising Data	Better fraud detection and audience targeting	Privacy issues and possible prejudice	Not specified	Enhanced ad effectiveness
(Thirupurasundari et al., 2023)	ML-Based Query Optimization	Big Data Systems	Performance improvement and adaptive optimization	High computational cost	Up to 60% improvement	Enhanced query efficiency
(Ding et al., 2019)	AI-Driven Index Optimization	Industry-standard benchmarks	Reduced indexing errors	Dependency on ML accuracy	Up to five times fewer errors in cost estimation	Improved automated indexing
(Dong et al., 2024)	Using Deep Reinforcement Learning to Adjust Knobs	MySQL, PostgreSQL datasets	Expert-guided machine learning for quicker database tweaking	High initial training cost	Up to 26% improvement	Optimized database configurations
(Dou et al., 2023)	ML Methods for Small Data	Molecular Science Data	Improved learning with tiny datasets	Limited generalizability	Not specified	Enhanced molecular study predictive analytics
(Du et al., 2020)	ML for Spatial Data Handling	Remote Sensing Images	Improved classification and interpolation	Large labeled datasets are necessary.	Not specified	Better analysis of geographic data
(During, n.d.)	Improved Query Optimization for Dispersed Databases	Supply Chain Databases	Strong and effective supply networks	Complex implementation	Not specified	Enhanced performance of queries in distributed systems
(M.M.F. Fahima et al., 2024)	ML for Database Management	Various DBMS Workloads	Workload optimization and adaptive indexing	High computational needs	Not specified	Increased database performance
(Gadde, 2022)	AI in SQL Query Processing	SQL Workloads	Improved indexing and automated optimization	Integration challenges	Not specified	Improved query efficiency
(Guo et al., 2025)	SQL+ML Query Optimization	JOB, TPC-H, SSB, Flight datasets	Quicker query execution using machine learning predicates	intricate query optimization procedure	Up to 3x faster processing	Enhanced SQL+ML integration

(Hasan et al., 2020)	Using Deep Learning to Estimate Selectivity	Census, IMDB datasets	Reliable assessment of selectivity for intricate questions	High training time	Not specified	Enhanced accuracy of the query optimizer
(Aoun, n.d.)	Advanced Methods for Processing Queries	Systems for heterogeneous large data	Systems for heterogeneous large data	High computational cost	Not specified	Enhanced retrieval performance
(Islam, 2024)	ML & AI for SQL Optimization	Various SQL database benchmarks	Predictive analytics in real time and enhanced security	Complex integration challenges	Not specified	Enhanced database management
(Jambigi et al., 2024)	ML for Root Cause Analysis in SAP HANA	SAP HANA workload logs	Automated failure classification	Managing new problems is still difficult.	F1-Macro score improved by 4.77%	More precise explanations of SQL failures
(Kaur et al., 2018)	Secure Healthcare Framework based on ML	Healthcare datasets	Improved data security and illness diagnosis	Privacy concerns in implementation	Not specified	Better analytics for healthcare data
(Marcus et al., 2021)	SageDB - Learned Database System	Custom application datasets	Execution of highly specific queries	Complex problems with adaptation and development	Not specified	Effective execution of learnt databases
(G. Li et al., 2018)	DRL for Query-Aware Tuning with QTune	Real database environments	Effective database configuration tuning	High training cost	notable enhancement in performance	Optimized database performance
(M.-L. Li, 2023)	DL for Digital Twins in IoT and Smart Cities	Smart city IoT data	Enhanced data accuracy and energy efficiency	Complexity of implementation	Accuracy 97.8%	Reduced data transmission delay
(Wisam Altaher & Hasan Hussein, n.d.)	Using Deep Learning to Estimate Selectivity	Census, IMDB datasets	Reliable assessment of selectivity for intricate questions	High training time	Not specified	Enhanced accuracy of the query optimizer
(Aoun, n.d.)	Advanced Methods for Processing Queries	Heterogeneous big data systems	Enhanced relevance ranking and query efficiency	High computational cost	Not specified	Enhanced retrieval performance
(X. Li et al., 2022)	ML-Based Data Acquisition, Query Optimization, and Indexing	Datasets for business statistics, web search, and traffic scheduling	Adaptive and lightweight indexing, strong query optimizers	High computational demands	Not specified	Enhanced ML model training and DBMS performance

(M.-L. Li, 2023)	AI-Driven Database Performance Tuning	Enterprise database workloads	Optimizing queries and indexing automatically	Complex integration	Not specified	Improved productivity and less manual labor in database tweaking
(Ma & Triantafillou, 2019)	DBEst: Machine Learning Models for Rough Query Processing	TPC-DS, real-life datasets	Quicker reaction times and less memory use	When samples are smaller, errors rise.	High precision with adequate data	Significantly shorter query execution time
(Marcus et al., 2018)	Neo: Learned Query Optimizer	TPC-H, IMDB datasets	constantly picks up knowledge from execution patterns	High initial training cost	Like the most advanced optimizers	Competitive results using for-profit optimizers
(Marcus et al., 2021)	Bao: AI-Powered Tips for Query Optimization	Cloud database workloads	Reinforcement learning combined with adaptive learning	Challenges in practical integration	Up to 30% query speedup	Enhanced performance of cloud-based queries
(Meduri et al., 2021)	Using RNN and Q-Learning to Predict SQL Queries	Real-world SQL logs	anticipates the subsequent query for speculative execution.	restricted ability to generalize beyond patterns	Q-Learning outperforms RNNs	Enhanced response speed and query interaction
(Milicevic & Babovic, 2024)	Methodical Evaluation of DL in the Execution of Queries	Various DBMS datasets	Improved tuning, optimization, and indexing	Implementation challenges	Not specified	thorough assessment of DBMS deep learning
(Mitchell & Nelson, 2021)	ML Trends in Query Processing	Big data environments	NLP integration, predictive analytics, and query optimization	High training overhead	Not specified	Improved handling of massive data queries
(Oluwafemi Oloruntoba, 2025)	AI-Powered Self-Sustained Database Administration	Enterprise IT systems	Intelligent indexing, self-tuning, and predictive query optimization	Security and compliance concerns	Not specified	decreased administrative overhead and increased scalability
(Panwar, 2024)	Advanced SQL Stored Procedures	SQL Server workloads	Using precompiled procedures allows for faster execution.	Implementation complexity	Not specified	Enhanced processing of huge data queries
(Patel et al., 2020)	ML and DL in Drug Discovery	screening at high throughput,	Improved prediction of	High computational cost	Not specified	A more effective method for

		molecular databases	drug candidates			finding new drugs
(Polkowski et al., 2021)	Prioritizing Computational and Evolutionary Queries	Virtual database environments	Improved execution plans	Integration complexity	Not specified	Improved techniques for query optimization
(Rachakatla & Machireddy, n.d.)	ML in Data Warehousing	Data integration systems	Automated transformation and matching of schemas	Computational overhead	Not specified	Enhanced data warehousing efficiency
(Rahman et al., 2024)	Advanced SQL Query Optimization	Large-scale SQL databases	Faster query execution by 40%	Requires extensive indexing	Not specified	Better big data analytics in real time
(Ramu, 2023)	Optimizing Database Performance	Enterprise database workloads	Improved resource utilization	High complexity of implementation	Not specified	Scalability and query execution optimization
(Singh et al., 2025)	Using Quantum Machine Learning to Classify Large Data	Caltech 101 dataset	Quicker and more effective classification	Hardware limitations	98%	Enhanced image classification
(Sundaram et al., 2023)	Deep Learning and the Firefly Algorithm	Near-duplicate datasets for image detection	Enhanced accuracy in similarity detection	High processing power is needed.	Not specified	Improved detection of duplicate images
(Hayath et al., 2023)	SQL Optimization Techniques	Various SQL workloads	Improved query structure and speed of execution	Implementation complexity	Not specified	Enhanced effectiveness of database queries
(D. Wang et al., 2014)	Regularized Local Coordinate Coding with Weak Labels	extensive online libraries of facial images	Better face annotation accuracy	Label noise in datasets	Not specified	Improved face annotation based on retrieval
(Z. Wang et al., 2021)	Optimizing Heat Pipes using Big Data Machine Learning	Data from simulated and experimental heat pipes	Quicker and more precise prediction of heat transfer	Complex data processing	Not specified	Optimized heat pipe performance
(Xie et al., 2021)	An algorithm for gravitational search combined with least squares support vector regression	Economic Indexes and Baidu Search Query Information	Enhanced forecasting performance	Computational complexity	Not specified	Better demand forecasting for cruise tourism

(J. Zhang et al., 2019)	Tuning Cloud Databases with Deep Reinforcement Learning	TPC-H, Sysbench	Optimized database configurations	High training cost	Significantly improved	Improved database efficiency in cloud settings
(A. Zhang et al., 2022)	Federated Learning, Transformer Models, and Deep Generative Models	EHRs for Healthcare and Medical Imaging Information	Improved use of data and protection of privacy	Challenges in model deployment	Not specified	Better performance of ML models in the medical field
(Zhou et al., 2022)	AI4DB and DB4AI Techniques	Enterprise DB Systems	Support for AI deployment and automated optimization	Integration complexity	Not specified	Enhanced effectiveness in AI-powered database administration
(Xie et al., 2021)	An algorithm for gravitational search combined with least squares support vector regression	Economic Indexes and Baidu Search Query Information	Enhanced forecasting performance	Computational complexity	Not specified	Better demand forecasting for cruise tourism
(A. Zhang et al., 2022)	Tuning Cloud Databases with Deep Reinforcement Learning	TPC-H, Sysbench	Optimized database configurations	High training cost	Significantly improved	Improved database efficiency in cloud settings
(Zhou et al., 2022)	AI4DB and DB4AI Techniques	Enterprise DB Systems	Support for AI deployment and automated optimization	Integration complexity	Not specified	Enhanced effectiveness in AI-powered database administration

Database management systems have seen a substantial transformation thanks to the incorporation of machine learning (ML) in query optimization, which has improved their scalability, flexibility, and efficiency. Numerous research have shown that machine learning (ML)-driven techniques, like supervised learning-based query optimizers (Aken et al., 2017) and deep reinforcement learning (Dong et al., 2024), may automate database optimization and increase query execution times by up to 94%. Furthermore, workload-based optimization (Rahman et al., 2024) and dynamic indexing strategies (Ding et al., 2019) have been made possible by ML techniques, which have resulted in significant decreases in query processing latency. High computational costs (G. Li et al., 2018), problems with model interpretability (Zhou et al., 2022), and the difficulty of incorporating ML-driven optimizers into current database systems (Guo et al., 2025) are still obstacles, though. Future studies should concentrate on improved explainability of AI-driven query execution (Akhtar & Farzana, 2024), adaptive security measures to address vulnerabilities in AI-enhanced database frameworks (Hamza Akhtar et al., 2025), and hybrid optimization techniques that combine rule-based strategies with ML models (Azhir et al., 2019). These issues can be resolved, allowing ML-powered database systems to develop further and provide reliable, self-optimizing solutions that can manage dynamic workloads and massive data environments.

6. CHALLENGES AND FUTURE DIRECTIONS

Machine learning (ML)-driven query optimization has made encouraging strides, but a number of obstacles prevent its broad use and efficacy. The interpretability of machine learning models, especially deep learning-based optimizers, is a significant worry. These models frequently operate as "black boxes," making it challenging for database administrators to comprehend and verify their decision-making procedures (W. merza Eido & Yasin, 2025). High computational costs are another major issue because machine learning models need to be updated and retrained frequently to accommodate changing workloads, which increases resource consumption (Saleh & Zebari, 2025a). Additionally, security flaws including data poisoning, unauthorized query manipulations, and adversarial attacks on machine learning models cast doubt on the resilience of AI-driven optimization methods, particularly in cloud-based database environments (Tato & Yasin, 2025). The absence of defined frameworks for incorporating machine learning (ML) into conventional database management systems is another drawback, since various workloads and architectures necessitate tailored solutions, making deployment and upkeep more difficult (Dou et al., 2023).

Future studies should concentrate on hybrid optimization frameworks that combine ML-driven methods with rule-based heuristics to ensure efficiency and interpretability (Azhir et al., 2019). Enhancing transparency and trust will need the development of explainable AI (XAI) models for query optimization, which will enable database administrators to understand and improve ML-based judgments (Akhtar & Farzana, 2024). Additionally, maximizing computational efficiency through the use of distributed training architectures, federated learning, and lightweight machine learning models would aid in lowering resource consumption and enhancing scalability (Guo et al., 2025). To ensure secure and dependable query execution, it will be essential to address security issues using anomaly detection systems, encryption-based machine learning models, and adversarial-resistant AI algorithms (Islam, 2024). The next generation of intelligent databases will be shaped by the incorporation of autonomous AI-driven database systems that can self-learn, self-tune, and self-repair. These systems will offer previously unheard-of performance and flexibility while managing complicated, large-scale workloads (Oluwafemi Oloruntoba, 2025).

7. CONCLUSION

By increasing productivity, flexibility, and scalability, the use of machine learning (ML) into query optimization has fundamentally changed database administration. Databases may now automatically optimize query execution, lower latency, and enhance resource utilization thanks to a variety of machine learning (ML)-driven approaches, including deep learning, reinforcement learning, and predictive analytics (Saleh & Zebari, 2025a). Significant speed improvements have been achieved by implementing machine learning (ML) in indexing, query burden management, and execution plan selection, especially in cloud-based and distributed database contexts (W. merza Eido & Yasin, 2025). However, issues including model interpretability, security flaws, and the high computational cost of training ML models continue to be major obstacles to broad adoption (Tato & Yasin, 2025). To guarantee effective and safe ML-driven query optimization, future research should concentrate on creating hybrid optimization frameworks, explainable AI strategies, and strong security measures (Dou et al., 2023). The future of ML-powered database systems will result in more intelligent, self-adaptive, and autonomous query optimization by tackling these issues, opening the door for next-generation database management systems (Guo et al., 2025).

REFERENCES

1. Abbasi, M., Bernardo, M. V., Váz, P., Silva, J., & Martins, P. (2024). Adaptive and Scalable Database Management with Machine Learning Integration: A PostgreSQL Case Study. Information (Switzerland), 15(9). https://doi.org/10.3390/info15090574
2. Ahmadi, S. (n.d.). Optimizing Data Warehousing Performance through Machine Learning Algorithms in the Cloud. International Journal of Science and Research, 2023(12), 1859–1867. https://doi.org/10.21275/SR231224074241ï
3. Aken, D. Van, Pavlo, A., Gordon, G. J., & Zhang, B. (2017). Automatic database management system tuning through large-scale machine learning. Proceedings of the ACM SIGMOD International Conference on Management of Data, Part F127746, 1009–1024. https://doi.org/10.1145/3035918.3064029
4. Akhtar, S., & Farzana, N. (2024). Optimizing Query Performance in Distributed Databases: A Comprehensive Approach with Autonomous AI, Reinforcement Learning, and Explainable AI. https://doi.org/10.13140/RG.2.2.13274.56008
5. Alamu, R. (n.d.). Deep Learning for Data Management: Enhancing Data Structuring and Query Optimization. https://www.researchgate.net/publication/389316639
6. Aoun, M. (n.d.). Improving query efficiency in heterogeneous big data environments through advanced query processing techniques. https://www.researchgate.net/publication/377334744
7. Ashlam, A. A., Badii, A., & Stahl, F. (2022). A Novel Approach Exploiting Machine Learning to Detect SQLi Attacks. Proceedings of the 2022 5th International Conference on Advanced Systems and Emergent Technologies, IC_ASET 2022, 513–517. https://doi.org/10.1109/IC_ASET53395.2022.9765948
8. Azhir, E., Navimipour, N. J., Hosseinzadeh, M., Sharifi, A., & Darwesh, A. (2019). Query optimization mechanisms in the cloud environments: A systematic study. International Journal of Communication Systems, 32(8). https://doi.org/10.1002/dac.3940
9. BRIDGING DATA MANAGEMENT AND MACHINE LEARNING: CASE STUDIES ON INDEX, QUERY OPTIMIZATION, AND DATA ACQUISITION. (n.d.).

10. Chandramouli, K., Kliegr, T., Nemrava, J., Svatek, V., & Izquierdo, E. (n.d.). Query Refinement and User Relevance Feedback for Contextualized Image Retrieval. http://wordnet.princeton.edu
11. Choi, J. A., & Lim, K. (2020). Identifying machine learning techniques for classification of target advertising. In ICT Express (Vol. 6, Issue 3, pp. 175–180). Korean Institute of Communications Information Sciences. https://doi.org/10.1016/j.icte.2020.04.012
12. Ding, B., Das, S., Marcus, R., Wu, W., Chaudhuri, S., & Narasayya, V. R. (2019). AI meets AI: Leveraging query executions to improve index recommendations. Proceedings of the ACM SIGMOD International Conference on Management of Data, 1241–1258. https://doi.org/10.1145/3299869.3324957
13. Dong, W., Liu, W., Xi, R., Hou, M., & Fan, S. (2024). MLETune: Streamlining Database Knob Tuning via Multi-LLMs Experts Guided Deep Reinforcement Learning. Proceedings of the International Conference on Parallel and Distributed Systems - ICPADS, 226–235. https://doi.org/10.1109/ICPADS63350.2024.00038
14. Dou, B., Zhu, Z., Merkurjev, E., Ke, L., Chen, L., Jiang, J., Zhu, Y., Liu, J., Zhang, B., & Wei, G. W. (2023). Machine Learning Methods for Small Data Challenges in Molecular Science. In Chemical Reviews (Vol. 123, Issue 13, pp. 8736–8780). American Chemical Society. https://doi.org/10.1021/acs.chemrev.3c00189
15. Du, P., Bai, X., Tan, K., Xue, Z., Samat, A., Xia, J., Li, E., Su, H., & Liu, W. (2020). Advances of Four Machine Learning Methods for Spatial Data Handling: a Review. In Journal of Geovisualization and Spatial Analysis (Vol. 4, Issue 1). Springer Nature. https://doi.org/10.1007/s41651-020-00048-5
16. During, A. D. (n.d.). Enhanced Query Optimization in Distributed Databases for Resilient and Efficient Supply Chains. https://doi.org/10.13140/RG.2.2.26612.36488
17. Eido, W. M., & Ibrahim, I. M. (2025). Ant Colony Optimization (ACO) for Traveling Salesman Problem: A Review. Asian Journal of Research in Computer Science, 18(2), 20–45. https://doi.org/10.9734/ajrcos/2025/v18i2559
18. Eido, W. merza, & Yasin, H. M. (2025). Pneumonia and COVID-19 Classification and Detection Based on Convolutional Neural Network: A Review. Asian Journal of Research in Computer Science, 18(1), 174–183. https://doi.org/10.9734/ajrcos/2025/v18i1556
19. Eido, W. merza, & Zeebaree, S. R. M. (2025). A Review of Blockchain Technology In E-business: Trust, Transparency, and Security in Digital Marketing through Decentralized Solutions. Asian Journal of Research in Computer Science, 18(3), 411–433. https://doi.org/10.9734/ajrcos/2025/v18i3602
20. Gadde, H. (2022). Integrating AI into SQL Query Processing: Challenges and Opportunities. In International Journal of Advanced Engineering Technologies and Innovations (Vol. 01).
21. Guo, Y., Li, G., Hu, R., & Wang, Y. (2025). In-database query optimization on SQL with ML predicates. VLDB Journal, 34(1). https://doi.org/10.1007/s00778-024-00888-3
22. Hamza Akhtar, M., Ali, A., Ali, S., Nasim, F., Hamza Aziz, M., Khan, H., & Asad Ali Naqvi, S. (2025). Spectrum of Engineering Sciences A Novel Machine Learning Approach for Database Exploitation to Enhance Database Security: A Survey. 3(2).
23. Hasan, S., Thirumuruganathan, S., Augustine, J., Koudas, N., & Das, G. (2020). Deep Learning Models for Selectivity Estimation of Multi-Attribute Queries. Proceedings of the ACM SIGMOD International Conference on Management of Data, 1035–1050. https://doi.org/10.1145/3318464.3389741
24. Hayath, T. M., Usman, K., Shafiulla, M., & Dadapeer. (2023). An Overview of SQL Optimization Techniques for Enhanced Query Performance. 2nd IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics, ICDCECE 2023. https://doi.org/10.1109/ICDCECE57866.2023.10151265
25. Islam, S. (2024). FUTURE TRENDS IN SQL DATABASES AND BIG DATA ANALYTICS: IMPACT OF MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE. International Journal of Science and Engineering, 1(4), 47–62. https://doi.org/10.62304/ijse.v1i04.188
26. Jambigi, N., Hammesfahr, J., Mueller, M., Bach, T., & Felderer, M. (2024). On Enhancing Root Cause Analysis with SQL Summaries for Failures in Database Workload Replays at SAP HANA. Proceedings - 2024 IEEE 35th International Symposium on Software Reliability Engineering Workshops, ISSREW 2024, 85–90. https://doi.org/10.1109/ISSREW63542.2024.00052
27. Kaur, P., Sharma, M., & Mittal, M. (2018). Big Data and Machine Learning Based Secure Healthcare Framework. Procedia Computer Science, 132, 1049–1059. https://doi.org/10.1016/j.procs.2018.05.020
28. Li, G., Zhou, X., Li, S., & Gao, B. (2018). QTune: A QueryAware database tuning system with deep

reinforcement learning. Proceedings of the VLDB Endowment, 12(12), 2118–2130. https://doi.org/10.14778/3352063.3352129

29. Li, M.-L. (2023). AI-Driven Database Performance Tuning: Automated Indexing and Query Optimization. In Advances in Computer Sciences (Vol. 6).
30. Li, X., Liu, H., Wang, W., Zheng, Y., Lv, H., & Lv, Z. (2022). Big data analysis of the Internet of Things in the digital twins of smart city based on deep learning. Future Generation Computer Systems, 128, 167–177. https://doi.org/10.1016/j.future.2021.10.006
31. Ma, Q., & Triantafillou, P. (2019). DBEST: Revisiting approximate query processing engines with machine learning models. Proceedings of the ACM SIGMOD International Conference on Management of Data, 1553–1570. https://doi.org/10.1145/3299869.3324958
32. Marcus, R., Negi, P., Mao, H., Tatbul, N., Alizadeh, M., & Kraska, T. (2021). Bao: Making Learned Query Optimization Practical. Proceedings of the ACM SIGMOD International Conference on Management of Data, 1275–1288. https://doi.org/10.1145/3448016.3452838
33. Marcus, R., Negi, P., Mao, H., Zhang, C., Alizadeh, M., Kraska, T., Papaemmanouil, O., & Tatbul, N. (2018). Neo: A Learned query optimizer. Proceedings of the VLDB Endowment, 12(11), 1705–1718. https://doi.org/10.14778/3342263.3342644
34. Meduri, V. V., Chowdhury, K., & Sarwat, M. (2021). Evaluation of Machine Learning Algorithms in Predicting the Next SQL Query from the Future. ACM Transactions on Database Systems, 46(1). https://doi.org/10.1145/3442338
35. Milicevic, B., & Babovic, Z. (2024). A systematic review of deep learning applications in database query execution. Journal of Big Data, 11(1). https://doi.org/10.1186/s40537-024-01025-1
36. Mitchell, O., & Nelson, S. (2021). Machine Learning for Query Processing in Big Data Analytics: Trends. In Print) International Journal of Engineering and Advanced Technology Studies (Vol. 9, Issue 1).
37. M.M.F. Fahima, A.H. Sahna Sreen, S.L. Fathima Ruksana, D.T.E. Weihena, & M.H.M. Majid. (2024). Machine Learning for Database Management and Query Optimization. Elementaria: Journal of Educational Research, 2(1), 96–108. https://doi.org/10.61166/elm.v2i1.66
38. Oluwafemi Oloruntoba. (2025). AI-Driven autonomous database management: Self-tuning, predictive query optimization, and intelligent indexing in enterprise it environments. World Journal of Advanced Research and Reviews, 25(2), 1558–1580. https://doi.org/10.30574/wjarr.2025.25.2.0534
39. Panwar, V. (2024). Optimizing Big Data Processing in SQL Server through Advanced Utilization of Stored Procedures. International Journal of Management, IT & Engineering, 14, 2. http://www.ijmra.us,
40. Patel, L., Shukla, T., Huang, X., Ussery, D. W., & Wang, S. (2020). Machine Learning Methods in Drug Discovery. Molecules, 25(22). https://doi.org/10.3390/MOLECULES25225277
41. Polkowski, Z., Mishra, J. P., & Mishra, S. K. (2021, July 1). Prioritization of complex heterogeneous queries using evolutionary and computational approach. Proceedings of the 13th International Conference on Electronics, Computers and Artificial Intelligence, ECAI 2021. https://doi.org/10.1109/ECAI52376.2021.9515096
42. Rachakatla, S. K., & Machireddy, J. R. (n.d.). The Role of Machine Learning in Data Warehousing: Enhancing Data Integration and Query Optimization.
43. Rahman, M. M., Islam, S., Kamruzzaman, M., & Joy, Z. H. (2024). ADVANCED QUERY OPTIMIZATION IN SQL DATABASES FOR REAL-TIME BIG DATA ANALYTICS. ACADEMIC JOURNAL ON BUSINESS ADMINISTRATION, INNOVATION & SUSTAINABILITY, 4(3), 1-1–14. https://doi.org/10.69593/ajbais.v4i3.77
44. Ramu, V. B. (2023). Optimizing Database Performance: Strategies for Efficient Query Execution and Resource Utilization. International Journal of Computer Trends and Technology, 71(7), 15–21. https://doi.org/10.14445/22312803/ijctt-v71i7p103
45. Saleh, R. A., & Yasin, H. M. (2025). Advancing Cybersecurity through Machine Learning: Bridging Gaps, Overcoming Challenges, and Enhancing Protection. Asian Journal of Research in Computer Science, 18(2), 206–217. https://doi.org/10.9734/ajrcos/2025/v18i2572
46. Saleh, R. A., & Zebari, I. M. I. (2025a). Enhancing Network Performance: A Comprehensive Analysis of Hybrid Routing Algorithms. Asian Journal of Research in Computer Science, 18(3), 1–16. https://doi.org/10.9734/ajrcos/2025/v18i3573
47. Saleh, R. A., & Zebari, I. M. I. (2025b). Enhancing Network Performance: A Comprehensive Analysis of Hybrid Routing Algorithms. Asian Journal of Research in Computer Science, 18(3), 1–16. https://doi.org/10.9734/ajrcos/2025/v18i3573
48. Singh, B., Indu, S., & Majumdar, S. (2025). Comparison of machine learning algorithms for

classification of Big Data sets. Theoretical Computer Science, 1024. https://doi.org/10.1016/j.tcs.2024.114938

49. Sundaram, S., Somasundaram, K., Jothilakshmi, S., Jayaraman, S., & Dhanalakshmi, P. (2023). Modelling of Firefly Algorithm with Densely Connected Networks for Near-Duplicate Image Detection System. International Conference on Sustainable Communication Networks and Application, ICSCNA 2023 - Proceedings, 66–72. https://doi.org/10.1109/ICSCNA58489.2023.10370117
50. Tato, F. R., & Yasin, H. M. (2025). Detecting Diabetic Retinopathy Using Machine Learning Algorithms: A Review. Asian Journal of Research in Computer Science, 18(2), 118–131. https://doi.org/10.9734/ajrcos/2025/v18i2566
51. Thirupurasundari, D. R., Rajesh Kumar, K., Palani, H. K., Ilangovan, S., & Senthilvel, P. G. (2023). Optimizing Query Performance in Big Data Systems Using Machine Learning Algorithms. 2023 International Conference on Communication, Security and Artificial Intelligence, ICCSAI 2023, 891–895. https://doi.org/10.1109/ICCSAI59793.2023.10421253
52. Wang, D., Hoi, S. C. H., He, Y., Zhu, J., Mei, T., & Luo, J. (2014). Retrieval-based face annotation by weak label regularized local coordinate coding. IEEE Transactions on Pattern Analysis and Machine Intelligence, 36(3), 550–563. https://doi.org/10.1109/TPAMI.2013.145
53. Wang, Z., Zhao, X., Han, Z., Luo, L., Xiang, J., Zheng, S., Liu, G., Yu, M., Cui, Y., Shittu, S., & Hu, M. (2021). Advanced big-data/machine-learning techniques for optimization and performance enhancement of the heat pipe technology – A review and prospective study. Applied Energy, 294. https://doi.org/10.1016/j.apenergy.2021.116969
54. Wisam Altaher, A., & Hasan Hussein, A. (n.d.). Head of information technology department Babelon-Iraq COMPARATIVE ANALYSIS OF MACHINE LEARNING AND TRADITIONAL QUERY OPTIMIZATION METHODS IN DATABASE MANAGEMENT SYSTEMS WITH ADAPTIVE MATHEMATICAL MODELING TEST.
55. Xie, G., Qian, Y., & Wang, S. (2021). Forecasting Chinese cruise tourism demand with big data: An optimized machine learning approach. Tourism Management, 82. https://doi.org/10.1016/j.tourman.2020.104208
56. Zhang, A., Xing, L., Zou, J., & Wu, J. C. (2022). Shifting machine learning for healthcare from development to deployment and from models to data. In Nature Biomedical Engineering (Vol. 6, Issue 12, pp. 1330–1345). Nature Research. https://doi.org/10.1038/s41551-022-00898-y
57. Zhang, J., Liu, Y., Zhou, K., Li, G., Xiao, Z., Cheng, B., Xing, J., Wang, Y., Cheng, T., Liu, L., Ran, M., & Li, Z. (2019). An end-to-end automatic cloud database tuning system using deep reinforcement learning. Proceedings of the ACM SIGMOD International Conference on Management of Data, 415–432. https://doi.org/10.1145/3299869.3300085
58. Zhou, X., Chai, C., Li, G., & Sun, J. (2022). Database Meets Artificial Intelligence: A Survey. IEEE Transactions on Knowledge and Data Engineering, 34(3), 1096–1116. https://doi.org/10.1109/TKDE.2020.2994641