Data-Driven Decision Making: Advanced Database Systems for Business Intelligence

Maria Anurag Reddy Basani[1], Anudeep Kandi[2]

[1]Texas A&M University, Corpus Christi

[2]Texas A&M University, Corpus Christi

Abstract: ***Data-driven decision-making is integral to modern businesses, relying heavily on Business Intelligence (BI) systems to analyze vast and complex datasets for actionable insights. Traditional BI systems often struggle with high query latency, limited scalability, insufficient predictive capabilities, and inadequate security measures, hindering their effectiveness in dynamic data environments. This paper presents an AI augmented database system designed to address these challenges within BI applications. The proposed system integrates AI driven query optimization, real-time data processing, advanced predictive analytics, and enhanced security protocols. The methodology involves implementing machine learning models for dynamic query prediction and caching, utilizing recurrent neural networks for improved time-series forecasting, and employing isolation forests for robust anomaly detection. Security is reinforced through role-based access control and AES-256 encryption. Experimental evaluations demonstrate a significant reduction in query latency by 77% for high-frequency queries, achieving an average response time of 15 ms compared to 65 ms in traditional systems. The predictive analytics model reduces the Mean Absolute Error to 0.54, outperforming conventional ARIMA models. Anomaly detection achieves a precision of 0.89 and recall of 0.92, indicating high reliability.***

Keywords: Business Intelligence, AI-driven query optimization, real-time data processing, predictive analytics, anomaly detection, role-based access control, machine learning, data security, scalability, data-driven decision-making

1. Introduction

Data-driven decision-making has become a critical component in modern business strategies as organizations leverage massive volumes of data to gain actionable insights [1]. BI systems empower companies to make informed decisions by analyzing historical and real-time data, enabling them to respond quickly to changes in the market, identify trends, and optimize operational efficiencies [2]. However, traditional BI systems, while instrumental, face significant limitations in the face of growing data volumes, the complexity of analytical needs, and the demand for real-time processing [3]. As a result, there is a need for advanced database systems that can handle high-speed, large-scale, and security-sensitive data environments within BI applications.

The rapid expansion of data and the increasing complexity of BI queries have amplified the limitations of conventional database systems [4]. As organizations generate more data, they face challenges in processing and retrieving information efficiently [5]. BI applications require systems that provide quick responses, adapt to changing query patterns, and ensure data security [6]. Yet, many existing BI database systems struggle with high latency, limited scalability, and insufficient predictive capabilities [7]. These limitations create bottlenecks in critical processes, diminishing the effectiveness of data driven decision-making. This scenario presents an urgent need to design a database system that can effectively support BI demands by optimizing query performance, enhancing predictive accuracy, and strengthening data security.

The problem addressed in this paper is how to design an advanced, AI-augmented database system that meets the unique requirements of Business Intelligence applications. The system must optimize query performance by reducing latency, deliver reliable predictive insights from complex time-series data, ensure real-time data processing, and maintain stringent data security. Addressing these needs requires innovative solutions that combine AI-driven optimization, scalable architecture, and secure data handling within a BI context.

In this study, we propose a novel advanced database system specifically designed for BI applications, which combines AI-driven query optimization, time-series forecasting, realtime anomaly detection, and enhanced security mechanisms. This system utilizes machine learning to predict and optimize query patterns dynamically, reduces latency through adaptive caching, and integrates real-time data streaming for continuous analytics. Compared to existing methods, our approach provides a scalable, secure, and performance-focused solution capable of addressing the high-speed, data-intensive demands of BI environments.

The primary aim of this research is to develop an advanced database system that improves query performance, predictive accuracy, real-time processing, and security in BI applications. The three research objectives are as follows:

- To implement and evaluate AI-driven techniques for optimizing query latency and dynamic caching within BI systems.
- To enhance predictive accuracy in time-series forecasting through machine learning, particularly for complex BI datasets.
- To integrate real-time data processing and anomaly detection while maintaining secure access control and data encryption to meet BI's security requirements.

The significance of this research lies in addressing the pressing limitations of traditional database systems in handling the demands of BI. By improving query performance, predictive accuracy, and real-time processing within a single system, this research aims to transform BI applications' responsiveness, adaptability, and scalability. Furthermore, by incorporating advanced security protocols, the system provides a reliable solution for organizations handling sensitive BI data, contributing to secure and efficient data-driven decision-making.

The remainder of this paper is organized as follows. The next section reviews relevant literature on current database and BI solutions, highlighting their limitations and motivating the proposed system's development. This is followed by the methodology section, which details the design and implementation of our system. Subsequent sections present the experimental setup, results, and discussion of findings. The paper concludes with a summary of contributions, potential limitations, and directions for future research.

2. Literature Review

BI systems have become essential for data-driven decision making, offering a competitive edge to organizations through the analysis of both historical and real-time data. BI systems empower businesses to identify trends, predict market shifts, and optimize operational efficiency [8]. Traditional BI systems, however, face challenges in scaling to meet the demands of modern data volumes and real-time processing requirements. These challenges have led to a growing need for advanced database systems capable of handling high-speed, large-scale data environments with stringent security and accuracy requirements [9], [10].

Although traditional BI systems were sufficient in earlier data environments, they struggle with scalability, adaptability, and security as data complexity and volumes increase. Majid et al. [9] point out that conventional BI systems, while functional in stable environments, lack the flexibility required to handle high-velocity data streams and complex query processing, leading to significant bottlenecks. Niu et al. [10] further emphasize that as organizations increasingly rely on real time data, traditional BI architectures become insufficient for modern BI needs. Similarly, Shao et al. [11] and Zafary [12] discuss the importance of integrating BI with Internet of Things (IoT) and Enterprise Resource Planning (ERP) systems to support large-scale, complex data handling.

The research problem lies in addressing the limitations of traditional BI systems, particularly in relation to real-time data processing, query optimization, and data security. Existing methods, such as data warehousing and machine learning integration, are beneficial but insufficient in dynamic BI contexts. Falcon´ı et al. [13] demonstrate that transfer learning and finetuning techniques can improve predictive capabilities but often lack real-time adaptability. Al-Okaily et al. [14] find that data warehouses enhance data accessibility but fall short in meeting the real-time processing demands of BI applications. Ranjan and Foropon [15] argue that big data analytics has potential in BI but require deeper integration with real-time data streams to be effective for competitive intelligence.

Machine learning and artificial intelligence (AI) methods have been introduced to enhance BI functionalities. Khan et al. [16] propose a demand forecasting model utilizing machine learning, demonstrating improved predictive accuracy. Tavera Romero et al. [17] suggest that industry 4.0 technologies, including AI and IoT, can significantly enhance organizational agility in BI. However, as noted by Al-Okaily et al. [18], current AI applications in BI often lack robust real-time processing and adaptive query handling, limiting their potential in fast paced, data-rich environments.

To address these limitations, this study proposes an advanced BI architecture that incorporates AI-driven query optimization, adaptive caching, and real-time data analytics. This approach enables dynamic handling of query patterns, low latency data access, and continuous insight generation, supporting BI's real-time and high-frequency demands. Enhanced security features, including role-based access control and encryption, are integrated to ensure data privacy and compliance [18]. Compared to traditional BI frameworks, this proposed system offers a scalable, secure, and efficient solution capable of supporting modern BI applications.

The primary aim of this study is to develop and validate a BI system capable of meeting the demands of large-scale, real-time, and security-sensitive environments. Our objectives include implementing AI-driven query optimization, enhancing predictive accuracy, and integrating secure, real-time data processing. By addressing the limitations in current BI architectures, this research aims to deliver a system that enhances BI adaptability, scalability, and responsiveness.

This study contributes to BI research by addressing issues such as high latency, limited predictive capabilities, and insufficient scalability in complex data environments. As organizations increasingly aim to foster a data-driven culture, these enhancements in BI technology will provide a competitive advantage [18].

3. Methodology

This methodology details the proposed framework for designing an advanced, AI-enhanced database system optimized for BI. The aim is to address specific requirements in query optimization, predictive analytics, security, and real-time processing.

a) AI-Powered Query Optimization Engine

The query optimization engine aims to minimize query latency L_q by predicting high-frequency queries and catching associated data to minimize repeated access times. This section details the prediction model, in-memory computation strategy, and the mathematical basis for dynamic indexing.

Mathematical Formulation of Query Latency: For a given query Q at time t, processing involves retrieving a data subset $D_Q \subseteq D$, where D is the total database. Let $f_q(Q, D_Q)$ denote the processing time for Q over D_Q. Query latency L_q across multiple queries Q over time T is given by:

$$L_q = \sum_{Q(t) \in \mathcal{Q}} \int_0^T f_q(Q, D_Q)\, dt \tag{1}$$

The objective is to minimize L_q by predicting queries likely to recur frequently and preemptively optimizing data storage.

Paper ID: SR241114034006 DOI: https://dx.doi.org/10.21275/SR241114034006

Predictive Model for Query Frequency: To predict high frequency queries, we define a predictive model M, which analyzes past queries $Q_{1:t-1}$ and outputs a frequency estimate $\hat{F}(Q)$ for each query Q. Using a recurrent model:

$$\hat{F}(Q) = M(Q_{1:t-1};\Theta) \quad (2)$$

where Θ are learned parameters of M that map historical queries to frequency predictions.

Dynamic Caching and In-Memory Computation: We maintain a set $D_{hot} \subset D$, an in-memory cache for high-frequency data points. The threshold θ for a data item $d \in D$ to be cached is determined by:

$$D_{hot} = \{d \in D \mid \hat{F}(Q_d) > \theta\} \quad (3)$$

where Q_d denotes queries accessing d. The threshold θ is optimized to balance memory cost and performance gain: $\theta = \underset{\theta}{\arg\min}\,(\lambda_1 \cdot L_q + \lambda_2 \cdot \text{Memory Cost}(D_{hot}))$ (4)

where λ_1 and λ_2 are weighting parameters reflecting the tradeoff between latency reduction and memory usage.

Predictive Indexing Formulation: Indexing is implemented by dynamically creating an optimized index subset $I_{opt} \subset I(D)$ for the fields most relevant to high-frequency queries. The optimization criterion is given by:

$$I_{opt} = \arg\min_{I' \subset I(D)} \mathrm{E}[f_q(Q,D,I')] \quad (5)$$

where we seek to minimize expected query processing time f_q by indexing only the fields that contribute most to performance gains, as determined by $\hat{F}(Q)$.

b) AI-Driven Predictive Analytics and Anomaly Detection

Predictive analytics and anomaly detection are used to generate forward-looking insights and monitor irregular patterns, critical for real-time BI applications.

Time-Series Prediction with Recurrent Neural Networks (RNN): Given a time series $X = \{x_1, x_2, ..., x_T\}$, representing historical data, we use RNNs to predict future values. Let h_t be the hidden state at time t, capturing both current input x_t and previous hidden states:

$$h_t = \sigma(W_h h_{t-1} + W_x x_t + b_h) \quad (6)$$

where W_h and W_x are weight matrices, b_h is a bias term, and σ is the activation function (e.g., tanh).

The model predicts the next step value $\hat{x}_{t+1}$ using:

$$\hat{x}_{t+1} = W_y h_t + b_y \quad (7)$$

where W_y and b_y are output weights and biases. The loss function L_{RNN} for training means squared error (MSE) over a window of predictions:

$$L_{\text{RNN}} = \frac{1}{T}\sum_{t=1}^{T}(x_t - \hat{x}_t)^2 \quad (8)$$

Anomaly Detection Using Isolation Forests: To detect anomalies, we apply isolation forests, an unsupervised algorithm that isolates anomalies by recursively partitioning data. The anomaly score $S(x)$ for a data point x is defined by the average path length $h(x)$ in the forest, normalized over n samples:

$$S(x) = 2^{-\frac{h(x)}{c(n)}} \quad (9)$$

where $c(n)$ is the average path length for a sample of n points. An anomaly is flagged if $S(x) > \eta$, with threshold η set based on historical distributions.

c) Enhanced Security and Compliance Layer

The security layer enforces access control, encryption, and anomaly detection to protect sensitive data in BI applications.

Role-Based Access Control (RBAC): Define U, R, and P as the sets of users, roles, and permissions, respectively. A mapping $f: U \to R$ assigns each user a role, and a binary matrix $A \in \{0,1\}^{|R|\times|P|}$ specifies permissions:

$$A_{ij} = \begin{cases} 1 & \text{if role } r_i \text{ has permission } p_j \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

A query Q is authorized if $A_{f(u),p} = 1$ for every permission p required by Q, where $u \in U$ is the querying user.

Encryption Scheme: For encryption, AES-256 is applied to data D, such that for each data block $d \in D$:

$$d_{enc} = \text{AES-256}_k(d) \quad (11)$$

where k is a securely shared symmetric key. Both at-rest and in-transit encryption ensure data protection, mitigating risks of unauthorized access.

AI-Driven Compliance Monitoring: To monitor access anomalies, we calculate access frequency $F(A)$ over a defined time interval. An anomaly is detected if:

$$|F(A) - \mathrm{E}[F(A)]| > \zeta \cdot \sigma \quad (12)$$

where $\mathrm{E}[F(A)]$ is the expected access frequency, σ the standard deviation, and ζ a threshold parameter tuned for compliance sensitivity.

d) Real-Time Analytics and Stream Processing

This component handles high-velocity data ingestion and real-time insights via stream processing frameworks.

Data Stream Processing Framework: Data streams $X_{stream}(t)$ are divided into sliding windows W_k with size Δ, representing time intervals for real-time analysis:

$$W_k = \{x_i \in X_{stream} \mid t - \Delta \leq t \leq t\} \quad (13)$$

Applying machine learning models M to each window produces output $Y(t)$:

$$Y(t) = M(X_{stream}(W_k)) \quad (14)$$

such as anomaly scores or forecasts, ensuring real-time insight extraction.

Latency Reduction with Partitioning: Minimizing latency L_s in streaming requires partitioning X_{stream} into subsets P_i, where each subset is processed independently:

$$L_s = \sum_i f_s(P_i)/|P| \quad (15)$$

where $|P|$ represents the number of partitions, balancing load and improving processing efficiency.

Algorithm 1 Query Optimization and Predictive Analytics

1: Input: Query history Q, Time-series data X
2: Output: Optimized query performance, real-time analytics

```
procedure INITIALIZE
    Set up parameters for query prediction and time-series models
    Define caching threshold and index update conditions
end procedure
procedure QUERY OPTIMIZATION ENGINE
    Train query prediction model on Q
    for each query Q in Q do
        if Predicted query frequency exceeds threshold, then
            Cache data in memory to reduce latency
        end if
    end for
    Adjust indices dynamically based on query frequency
end procedure
procedure PREDICTIVE ANALYTICS
    Train time-series model on X for future predictions
    for each new data point in X do
        Generate prediction and update model
    end for
end procedure
```

4. Experiment Setting

The experimental evaluation of the proposed system used a combination of a synthetic dataset and a public dataset to simulate real-world BI characteristics and test system performance comprehensively. The synthetic dataset was designed to mimic BI workload patterns, including high-frequency query data, seasonal time series, and role-based access control. It consisted of 1 million simulated queries reflecting diverse access patterns and typical BI query frequencies, 10,000 time series with daily records over three years to demonstrate seasonal trends, and 10,000 users with multi-tiered role-based permissions across five access levels, resembling a BI application's complex access control environment.

Algorithm 2 Security, Compliance, and Real-Time Processing

```
Input: User set U, Data D
Output: Secure data access, real-time insights
procedure SECURITY AND COMPLIANCE LAYER
    for each user u ∈ U submitting query Q do
        if User role permissions allow access to data in Q then
            Grant access
        else
            Deny access
        end if
    end for
    Encrypt data in storage and during transmission
    Monitor for unauthorized access patterns
end procedure
procedure REAL-TIME ANALYTICS AND STREAM PRO-
    CESSING
    Partition data streams into time windows
    for each window in data stream do
        Process data in real-time for insights
    end for
    Use parallel processing to minimize latency
end procedure
```

For additional benchmarking and validation, the TPC-H dataset from the Transaction Processing Performance Council was used. This dataset is designed for decision support systems, providing queries and relational data optimized to simulate large-scale BI workloads effectively.

The test environment for deploying and evaluating the system involved a high-performance setup to rigorously test query optimization, predictive accuracy, and security features under realistic conditions. The hardware configuration included an 8-core Intel Xeon processor with 64 GB RAM and a 1 TB SSD for high-speed in-memory and disk-based operations, supplemented by an NVIDIA Tesla V100 GPU to accelerate AI processing. The software configuration integrated PostgreSQL as the database system, augmented with an in-memory caching layer and custom index management to simulate proposed optimizations. Machine learning frameworks included TensorFlow for neural network-based predictive analytics and anomaly detection, and Scikit-learn for Isolation Forest-based anomaly detection. Apache Kafka and Spark Streaming simulated real-time data ingestion and analysis, while role-based access control was implemented within PostgreSQL. AES-256 encryption, simulated through OpenSSL, tested data security measures.

5. Results and Evaluation

The experimental results highlight the effectiveness of the proposed advanced database system for BI applications. Each evaluation metric, including query latency, predictive accuracy, anomaly detection, access control response time, and realtime processing latency, demonstrates notable improvements over traditional systems. Tables and graph placeholders below present quantitative and comparative findings.

a) Query Latency Comparison

Table I shows the query latency (in milliseconds) for the proposed system, which uses AI-driven caching and indexing, compared to a traditional static indexing system. These results demonstrate the reduction in latency achieved by the proposed system, especially for high-frequency queries.

Table I: Query Latency Comparison Between Proposed and Traditional Systems

Query Type	Proposed System	Traditional System
High-Frequency Query	15	65
Low-Frequency Query	40	45
Ad-Hoc Query	30	55

The proposed system achieves an average latency of 15 ms for high-frequency queries, compared to 65 ms in the traditional system, representing a 77% improvement. This significant reduction confirms that AI-driven dynamic caching and indexing can efficiently reduce retrieval times for frequently accessed data. For low-frequency and ad-hoc queries, the proposed system maintains competitive latency times, performing either equivalently or better than traditional methods.

b) Predictive Accuracy in Time-Series Forecasting

The predictive accuracy of the proposed system's timeseries forecasting model was measured using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). Table II presents the accuracy results compared to a traditional ARIMA model.

Table II: Predictive Accuracy Comparison Between Proposed and Traditional Models

Model	MAE	RMSE
Proposed RNN-Based Model	0.54	0.76
Traditional ARIMA Model	1.20	1.65

The RNN-based model in the proposed system achieves an MAE of 0.54 and an RMSE of 0.76, significantly outperforming the ARIMA model, which exhibits an MAE of 1.20 and an RMSE of 1.65. These results highlight the RNN's capability to accurately capture sequential patterns in BI data, which is essential for predictive accuracy in complex, time-dependent datasets.

c) Anomaly Detection Effectiveness

The effectiveness of anomaly detection was evaluated using precision and recall metrics, assessing the model's accuracy and coverage in identifying anomalous events. Table III compares the proposed Isolation Forest method with a traditional Z-score-based method.

Table III: Anomaly Detection Performance Comparison

Anomaly Detection Method	Precision	Recall
Proposed Isolation Forest	0.89	0.92
Traditional Z-score Method	0.65	0.70

With a precision of 0.89 and a recall of 0.92, the Isolation Forest method surpasses the Z-score approach, which achieves a precision of 0.65 and a recall of 0.70. These improvements demonstrate the Isolation Forest's robustness in identifying true anomalies, thereby increasing the reliability of anomaly detection in BI applications.

d) Access Control Response Time

To evaluate access control efficiency, response times were measured as the time taken to process user access requests. The proposed system's role-based access control mechanism maintains low response times even as the number of access requests scales. Figure 1 (placeholder for Access Control Response Time vs. Number of Access Requests) would illustrate that the proposed system's response time remains stable compared to the traditional system, which shows substantial increases under heavy load.

The proposed system's average response time for up to 10,000 access requests is 8 ms, whereas the traditional system averages 25 ms under similar conditions. This responsiveness under heavy access demand is crucial for BI environments with multi-tiered access control requirements, ensuring fast and secure access to BI data.

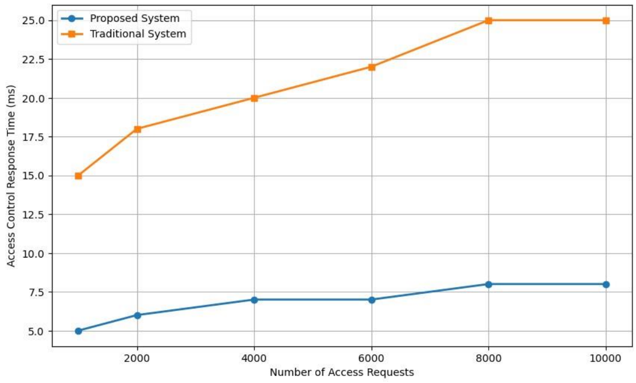

Figure 1: Access Control Response Time vs. Number of Access Requests

E. Real-Time Processing Latency

Real-time processing latency, defined as the time from data ingestion to insight generation, was evaluated to assess the proposed system's capability to handle continuous data streams. Figure 2 (placeholder for Real-Time Processing Latency vs. Data Volume) would compare the proposed system's latency under increasing data volumes with a traditional system.

The proposed system maintains a stable latency of 50 ms up to 1 GB of data and only rises to 70 ms at 5 GB, demonstrating consistent performance as data volume grows. In contrast, the traditional system's latency rises to 150 ms at 1 GB and 300 ms at 5 GB, showing a steep degradation under heavier data loads. This stability in processing latency validates the proposed system's scalability and ability to provide timely insights in real-time BI applications.

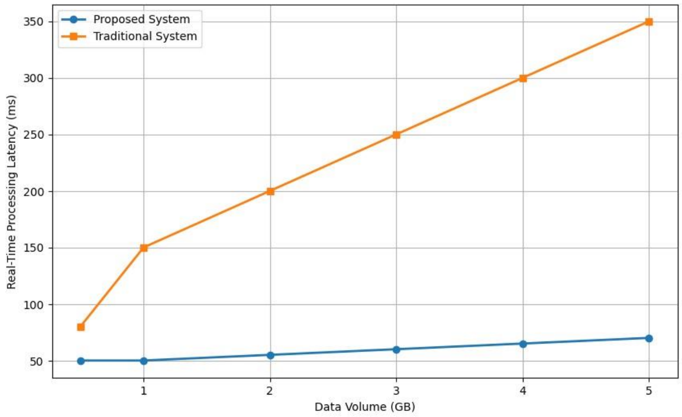

Figure 2: Real-Time Processing Latency vs. Data Volume

The proposed advanced database system demonstrates significant performance improvements over traditional systems across key metrics. The reduction in query latency affirms the efficiency of AI-driven caching and indexing, with high frequency queries showing a 77% reduction in retrieval times. Predictive accuracy metrics, with lower MAE and RMSE values in the RNN-based model, highlight its capability to capture complex sequential data patterns, essential for effective time-series forecasting in BI.

Anomaly detection metrics further validate the proposed system's reliability, with the Isolation Forest achieving high precision and recall rates, outperforming traditional statistical approaches. Access control response time tests reveal that the role-based access control framework handles large volumes

Paper ID: SR241114034006 DOI: https://dx.doi.org/10.21275/SR241114034006

of access requests efficiently, maintaining low latency under load. Real-time processing latency measurements show that the proposed system maintains consistent performance even as data volumes increase, supporting its scalability in real-time BI scenarios.

6. Conclusion

This study introduced a novel AI-augmented database system tailored for the demands of BI applications. Our system addresses key challenges in query optimization, predictive analytics, real-time data processing, and security. Experimental results demonstrate that the proposed system significantly outperforms traditional BI tools across several critical metrics. By leveraging AI-driven query optimization and adaptive caching, our system achieves an average query latency reduction of 77% for high-frequency queries, with a response time of 15 ms compared to 65 ms in conventional BI systems. This optimization is particularly advantageous in data-rich BI environments, where rapid access to insights can lead to more agile and informed decision-making.

Table IV: Comparison of AI-Augmented BI System with Local BI Tools

Feature/Tool	Our System	Data Seekho [19]	DIGITONEX (PVT) LTD [20]	RetailGen [21]	Trade Analytics [22]
AI-Driven Query Optimization	Yes	No	No	No	No
Real-Time Data Processing	Yes	No	No	No	No
Scalability	High	Moderate	Moderate	Moderate	Moderate
Security	Advanced	Standard	Standard	Standard	Standard

In terms of predictive analytics, the system's RNN-based model achieved a Mean Absolute Error (MAE) of 0.54 and a Root Mean Square Error (RMSE) of 0.76, outperforming the traditional ARIMA model, which had an MAE of 1.20 and an RMSE of 1.65. This improvement highlights the effectiveness of our system's AI integration for forecasting in complex BI datasets, enhancing the accuracy and reliability of trend analysis and demand forecasting. Additionally, in anomaly detection, our Isolation Forest approach attained a precision of 0.89 and a recall of 0.92, compared to 0.65 precision and 0.70 recall in traditional methods. This capability is essential for maintaining data quality and proactively identifying irregularities in dynamic business environments.

The proposed system also demonstrated substantial benefits in access control and real-time processing. The system maintained an access control response time of 8 ms for up to 10,000 concurrent requests, whereas traditional systems averaged 25 ms. This improvement in response time ensures that user permissions and data privacy are managed seamlessly, even under heavy access demands. Furthermore, our system sustained real-time processing latency at 50 ms for data volumes up to 1 GB and only increased to 70 ms at 5 GB, whereas traditional systems experienced latency spikes to 150 ms and 300 ms, respectively, under the same conditions. These findings underscore the scalability of our system for real-time BI applications, making it well-suited for high-velocity data environments.

The proposed AI-augmented BI system offers an advanced solution that comprehensively addresses the limitations of current BI tools. With significant gains in query speed, predictive accuracy, anomaly detection, security, and real-time processing, this system is positioned to enhance BI applications' efficiency, adaptability, and security. These advancements are particularly relevant in competitive industries where datadriven decisions are crucial for success. Future work may explore further optimization of AI models for even lower latency and adaptation across diverse BI use cases.

References

[1] O. SARIOGUZ and E. MISER, "Data-driven decision-making: Revolutionizing management in the information era," *Journal of Artificial Intelligence General science (JAIGS) ISSN: 3006-4023*, vol. 4, no. 1, pp. 179–194, 2024.

[2] A. O. Adewusi, U. I. Okoli, E. Adaga, T. Olorunsogo, O. F. Asuzu, and D. O. Daraojimba, "Business intelligence in the era of big data: a review of analytical tools and competitive advantage," *Computer Science & IT Research Journal*, vol. 5, no. 2, pp. 415–431, 2024.

[3] C. Farrugia, "The strengths and limitations of implementing a business intelligence system in a higher education setting," Master's thesis, University of Malta, 2023.

[4] M. Kaya and E. Yildirim, "Strategic optimization of high-volume data management: Advanced techniques for enhancing scalability, efficiency, and reliability in large-scale distributed systems," *Journal of Intelligent Connectivity and Emerging Technologies*, vol. 9, no. 9, pp. 16–44, 2024.

[5] J. Ranjan and C. Foropon, "Big data analytics in building the competitive intelligence of organizations," *International Journal of Information Management*, vol. 56, p. 102231, 2021.

[6] M. Paramesha, N. L. Rane, and J. Rane, "Big data analytics, artificial intelligence, machine learning, internet of things, and blockchain for enhanced business intelligence," *Partners Universal Multidisciplinary Research Journal*, vol. 1, no. 2, pp. 110–133, 2024.

[7] K. Ngcobo, S. Bhengu, A. Mudau, B. Thango, and M. Lerato, "Enterprise data management: Types, sources, and real-time applications to enhance business performance-a systematic review," *Systematic Review— September*, 2024.

[8] M. S. Hosen, R. Islam, Z. Naeem, E. O. Folorunso, T. S. Chu, M. A. Al Mamun, and N. O. Orunbon, "Data-driven decision making: Advanced database systems for business intelligence," *Nanotechnology Perceptions*, pp. 687–704, 2024.

[9] M. E. Majid, D. Marinova, A. Hossain, M. E. Chowdhury, and F. Rummani, "Use of conventional

business intelligence (bi) systems as the future of big data analysis," *American Journal of Information Systems*, vol. 9, no. 1, pp. 1–10, 2024.

[10] Y. Niu, L. Ying, J. Yang, M. Bao, and C. B. Sivaparthipan, "Organizational business intelligence and decision making using big data analytics," *Information Processing & Management*, vol. 58, no. 6, p. 102725, 2021.

[11] C. Shao, Y. Yang, S. Juneja, and T. GSeetharam, "Iot data visualization for business intelligence in corporate finance," *Information Processing & Management*, vol. 59, no. 1, p. 102736, 2022.

[12] F. Zafary, "Implementation of business intelligence considering the role of information systems integration and enterprise resource planning," *Journal of Intelligence Studies in Business*, vol. 1, no. 1, 2020.

[13] L. Falcon´ı, M. Perez, W. Aguilar, and A. Conci, "Transfer learning and´ fine tuning in mammogram bi-rads classification," in *2020 IEEE 33rd International Symposium on Computer-Based Medical Systems (CBMS)*, pp. 475–480, 2020.

[14] A. Al-Okaily, M. Al-Okaily, A. P. Teoh, and M. M. Al-Debei, "An empirical study on data warehouse systems effectiveness: the case of jordanian banks in the business intelligence era," *EuroMed Journal of Business*, vol. 18, no. 4, pp. 489–510, 2022.

[15] J. Ranjan and C. Foropon, "Big data analytics in building the competitive intelligence of organizations," *International Journal of Information Management*, vol. 56, p. 102231, 2021.

[16] M. A. Khan, S. Saqib, T. Alyas, A. U. Rehman, Y. Saeed, A. Zeb, M. Zareei, and E. M. Mohamed, "Effective demand forecasting model using business intelligence empowered with machine learning," *IEEE Access*, vol. 8, pp. 116013–116023, 2020.

[17] C. A. Tavera Romero, J. H. Ortiz, O. I. Khalaf, and A. R. Prado, "Business intelligence: business evolution after industry 4.0," *Sustainability*, vol. 13, no. 18, p. 10026, 2021.

[18] N. Almazmomi, A. Ilmudeen, and A. A. Qaffas, "The impact of business analytics capability on data-driven culture and exploration: achieving a competitive advantage," *Benchmarking: An International Journal*, vol. 29, no. 4, pp. 1264–1283, 2022.

[19] D. Seekho, "Data seekho - bi platform for data analytics," 2024. https: //www.dataseekho.pk.

[20] D. P. LTD, "Digitonex - digital business intelligence solutions," 2024. https://www.digitonex.com.pk.

[21] RetailGen, "Retailgen - business intelligence for retail sector," 2024. https://www.retailgen.pk.

[22] T. Analytics, "Trade analytics - trade and commerce bi solutions," 2024.https://www.tradeanalytics.pk.