**International Journal of Artificial Intelligence, Data Science, and Machine Learning**
Grace Horizon Publication | Volume 3, Issue 3, 102-110, 2022
ISSN: 3050-9262 | https://doi.org/10.63282/3050-9262.IJAIDSML-V3I3P111

*Original Article*

# Using ML Models to Detect Unusual Database Activity or Performance Degradation

Nagireddy Karri[1], Sandeep Kumar Jangam[2], Partha Sarathi Reddy Pedda Muntala[3]
[1]Senior IT Administrator Database, Sherwin-Williams, USA.
[2]Lead Consultant, Infosys Limited, USA.
[3] Software Developer at Cisco Systems, Inc, USA.

***Abstract*** *- Database performance and activity has turned into a major concern due to the demanding growth of data and increased complexity of the existing database systems. Conventional monitoring is normally applied on the available thresholds and the potential ability to suspect the relationship by human eye, which cannot act as proper sensors to indicate the occurrence of subtle anomalies or predict the degree of performance degradation. The present paper considers the potential of machine learning (ML) models in identifying the signs of abnormal activity in the database and performance issues. The patterns are the indicators of suspicious activity that are recognized by the ML algorithms on both the historical database data metrics and transactional logs. The paper taking a comparison of various supervised /unsupervised approaches to learning includes Random Forest, Support Thanks Machines (SVM), K-Means clustering, and Auto encoders by their accuracy, precision, recall, and time efficiency. The results of the experimental evidence confirm that the ML-driven models are more efficient in comparison with the conventional monitoring frameworks in detecting deviations, such as query delays, resource bottlenecks and the unforeseen access pattern. The paper further proposes a blend structure of identifying the aberrations that would be an integration of the predictive performance analytics and anomaly detection to enable the preemptive action. The results demonstrate that the advantage of applying ML to the process of monitoring databases is that it can increase the detection rate, reduce the instances when false positives are detected, which is best utilized to control the resources and enhance the reliability of the system. The task adds to the growing body of research on intelligent database management and provides a glimpse of the actual implementation of the systems based on the use of MLs as a means of monitoring.*

***Keywords*** *- Machine Learning, Database Monitoring, Anomaly Detection, Performance Degradation, Predictive Analytics, Autoencoders, Random Forest, Support Vector Machines.*

## 1. Introduction

### *1.1. Background*

In the present day, enterprises heavily rely on database systems to store, manipulate, and access large amounts of formatted and unformatted information created by a wide range of applications, including e-commerce network, financial frameworks, health information, and cloud service providers. [1-3] As a trend with high workload database structures (including distributed system), high throughput, mixed work load, etc., it is currently of overriding importance to ensure effective performance in addition to coherence of the system. The root-cause behind the abnormal actions of the database can be traced to be caused due to an array of factors including a failure in hardware, sub-optimal or inefficient queries, configuration and system failures, or other more sinister causes such as an attack by cyber-attackers and attempted intrusion.

These aberrations can be extremely harmful and could encompass the long time-out, degradation of performance of the application utilized, loss of finances and dissatisfaction by the users. Older system monitors are primarily based on the set levels of key indicators which are fixed, such as the CPU usage, memory usage, disk inputs and query response time. As much as these techniques of specifically setting thresholds may come in handy in the discovery of any seeming deviation, it may not be effective enough in spotting any slight anomalies that could erode into an indicator of a developing performance issue. Moreover, severe thresholds can cause an enormous number of false results or false alarms that will work to the overload of the database administrator and can delay the entry of the timely intervention. The evolving size, non-homogeneity and dynamism of the modern enterprise databases are driving more advanced, dynamically adaptable approaches of observation, capable of detecting awareness and non-familiarity abnormalities in real-time. These problems emphasize the relevance of advanced techniques, such as machine learning ones, capable of acquiring normal behavior patterns on their own, identifying abnormalities, and helping to proactively organize databases to guarantee their high performance, reliability, and security in a more complex tasks.

### *1.2. Importance of ML Models to Detect Unusual Database Activity*

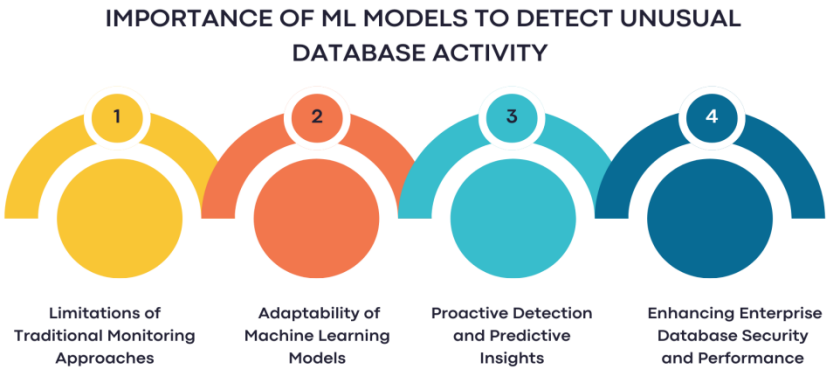


**Fig 1: Importance of ML Models to Detect Unusual Database Activity**

- **Limitations of Traditional Monitoring Approaches:** Traditional database monitoring systems tend to monitor using a set of pre-determined values of the key performance indicators such as CPU utilization, memory utilization, disk I/O and query response period. Though it could be true that these techniques can identify certain blatant deviations or deviations that are difficult to identify, they might fall short to identifying some more subtle deviations that do not appear to be out of control once some preset limits have been taken. This inadequacy can result in latent declaration of potential lapses in performance or system malfunction. There are also fixed thresholds and this is normally accompanied by false positives that drown administrators in irrelevant alerts and devalue operations. Through dynamic and complex database environments, the traditional methods could no longer be applied to gather complex pattern of an aberrant conduct that could arm an emergent risk.
- **Adaptability of Machine Learning Models:** Alternative methodologies may be replaced by machine learning (ML) to provide adaptable and data-grounded monitoring. Unlike threshold methods, there are no known patterns of normal database behavior that the ML algorithms cannot recognize on the basis of past data and refresh their models on the new available data. This enables the detection of a familiar and unfamiliar anomalies like finer deviations that may be the onset of a significant performance downfall. Although supervised ML models such as the Random Forest and Support Vector Machines can be identified to detect known consequences of anomaly satisfactorily in case of known labels in a dataset, unsupervised models such as Autoencoders and K-Means clustering can be used to identify novel anomalies in an unknown dataset since this is quite suitable in dynamic and changing contexts of the database.
- **Proactive Detection and Predictive Insights:** With the assistance of the ML models, the organizations will be able to switch to active database management rather than reactive monitoring. There is instant detection of abnormal activity with the assistance of ML, which may be used to help with predictive analytics to determine potential performance challenges. This enables database administrators take proactive expectations of corrective measures that can either be query optimization, resources reallocation, or infrastructure tracing before problems can get out of control. In such a manner, the resource-efficient operation of the CII-based monitoring can ensure the saving of time, the enhancement of system reliability and efficiency, as well as employ the resources in an efficient and cost-effective manner.
- **Enhancing Enterprise Database Security and Performance:** Besides tracking the performance, another possible use of ML models is that they assist in detecting suspicious access behaviour or cyber threats to the database. Use of an anomaly in the user query or transaction patterns will help in eliminating security menace without jeopardizing the integrity of the performance. Total, the application of ML-based monitoring systems will furnish companies with intelligent, dynamic, and effective systems to ensure the most optimal database running and ultra-safe in the continuously developed data environments.

### *1.3. Detect Unusual Database Activity or Performance Degradation*

It is important to identify abnormal database operations or decreasing performance to ensure smooth continuity, effectiveness, and safety of the contemporary enterprise database systems. [4,5] Unusual activity may be in various forms such as a sudden exclusion of execution time on a query, dynamic nature on increase in a CPU or memory, unusual transaction rate or anomalous disk I/O behavior. These anomalies could be caused by a number of factors, including hardware failures, badly written queries, bugs in software, configuration failures or an unintentional attack. These irregularities may develop to serious performance problems, system failures, or inconsistency of data, which may interfere with business activities and result in financial or reputational losses, unless these irregularities are detected in time. These subtle or emerging anomalies may not be detected using the traditional methods of database monitoring which are based on known thresholds of the key performance metrics. Too strict thresholds can fail to notice slow emerging problems and too sensitive thresholds can result in high numbers of alerts overwhelming the database administrators and decreasing productivity.

The solution to these issues is that modern technologies use machine learning (ML) models capable of extrapolating complex trends in database measurements and identify the abnormalities in the anticipated behavior. The supervised models including random Forest and Support Vector Machines (SVM) are effective in defining known anomalies under the situation where historical labeled data is available. Untrained models, e.g., K-Means clustering and Autoencoders, in turn, are just as good at the detection of anomalies that were never observed previously, particularly in time-series workloads. With the data of operating, these models are capable of continuously learning and therefore capable of distinguishing normal variation as compared to actual abnormal activity and hence capable of taking the initiative to intervene before the situation goes amok. Abnormal behavior identification does not only enhance performance of the system but also improves the level of security based on databases through the detection of unwanted access or suspicious pattern of query execution. All in all, when equipped with smart anomaly detection services, the enterprise databases will become stable, responsive, and stable in the presence of complex and high load operations that can provide the business organization with not only a stable operation but also long-term gains in the management of the vital data resources.

## 2. Literature Survey

### *2.1. Traditional Database Monitoring Techniques*

The traditional means of checking on the database generally rely on the set bounds of the key parameters of performance, such as the intensity of the CPU load, the response time of a query, [6-9] disk I/O and the use of memory. The database administrators come up with the alerts that should be triggered when these metrics go beyond the set values. This space has been popular with some of the tools that include Oracle Enterprise Manager, MySQL Enterprise Monitor and SQL server proiler that provide real time dashboards and alerting systems. Even though these techniques are effective in pointing to blatant violations like contention of resources or sudden spike in query response time, they typically cannot point to less obvious or gradually evolving violations. Furthermore, threshold based monitoring will tend to generate false positives which are likely to be numerous and drown administrators and result in reduction of efficiency of an operation. This reactive strategy is constrained in that it is not dynamically adjusted to the varying workloads or identifies complex relationships among multiple system measures.

### *2.2. Machine Learning for Anomaly Detection*

Machine learning (ML) models have become effective methods that can be used to identify anomalies in various other fields, such as databases. Random Forests and Support Vector Machines (SVM) are some examples of supervised learning models that need labelled datasets (predefined normal and anomalous behaviors). These models can be very accurate and make predictive information when trained on representative historical information. But in the dynamic database environments it may be difficult to have labeled data. To overcome this shortcoming, unsupervised methods of learning, such as K-Means clustering and Autoencoders, do not rely on pre-labels to identify anomalies in the data stream. These models are specifically applicable when we want to determine patterns or irregularities of behavior in logs of systems, database queries and network traffic that had not been previously observed. Surveys have always demonstrated that ML-based methods are more successful than conventional threshold-based methods to detect subtle, transforming anomalies and can more easily adapt to evolving business conditions.

### *2.3. Comparative Studies in Database Anomaly Detection*

Comparative studies in database anomaly detection have found that the ensemble and deep learning techniques tend to improve over the single-model ones. Random Forest and Gradient Boosting are ensemble methods, which employ several classifiers to increase their predictive accuracy and strength, especially when working with noisy data. Autoencoders and Long Short-Term Memory (LSTM) networks are deep learning models which are better at discovering temporal dependencies of time-series data, and can be used to track time-varying metrics of database performance trends. Autoencoders will, as an example, be able to reconstruct typical patterns and point out the abnormalities, whereas LSTMs will be able to identify the abnormalities in sequences of query executions or resource use over time. In Table 1, the results of some of these studies are summed up, which reveals the discrepancy in accuracy, precision, recall, and applicability. On the whole, all of these studies can indicate that machine learning techniques could be of great benefit compared to the traditional methods of monitoring, which could be especially useful in complex and dynamic environments.

### *2.4. Research Gaps*

Although machine learning has been effective at detecting anomalies, there are still some gaps in the research. The other challenge is that there are very few labeled datasets restricting the performance of supervised models. The current dynamism and changeability of the workloads on modern databases render the use of static models less credible and reliable since the trend of normal and anomalous occurrences may shift over time. Additionally, most solutions at work focus on reactive identification but not the proactive control, which opens the possibility of combining predictive analytics and the identification of anomalies. Such integration would give early warning and automatic corrective action before degeneration of performance occurs. The available literature is also typically on one model or one measure which does not take into account the entire interactions amongst the various parts of the system. Closing these lapses can lead to smoother, smarter, and elaborate ways of monitoring database performance within the realistic world conditions.

# 3. Methodology

## *3.1. Data Collection*

Various and popular database management systems (DBMS) to represent the diverse database environments like MySQL, PostgreSQL and Oracle were systematically used to collect available data, to create a variety of database environments. [10-12] The primary objective was to capture performance based metrics, which would now provide an insight on the regular operations and also any kind of anomalies. Measures of the important metrics included were CPU, memory usage, query run time, and number of transactions, disk I/O measures. The usage of the CPU was traced to determine the periods that the computer would need additional computing-power because this may be a sign of an intensive query or a current process running in the background. The information on memory consumption provided the understanding of the use of buffers and caches, which can be significant to the analysis of performance bottlenecks. The number of transactions and time taken by the queries were recorded to the database workload patterns to monitor the deviation that can possibly indicate query inefficientness or an abnormal user pattern. The disk I/O statistics was also taken since the disk read/write performance has been known to be a bottleneck in the database responsiveness particularly when executing systems process large volumes of data. The homogeneity of the data collection procedure was achieved by averaging the measures of a minute. This granularity made it possible to build a high-resolution time-series data, which would indicate the short and long-term performance of the database. These were aggregated by cleaning raw logs and performance counter by both the DBMS into one schema where the database could be consistent and able to compare them with the other systems. In addition, time all the data points right and handling missing or incompleting data were also taken into account, which is crucial to the integrity of time-series analyses. It is accomplished through subsequent consolidation of a pool of performance metrics of different platforms and standardization of the process of their convergence, the received dataset is an effective constituent of next anomaly detection and prediction. This is due to the fact that it is not only simple to detect the peculiar mannerisms of the workings of the database but it also can contribute to the models which could be applied to extrapolate within an unevenly matched database environment.

## *3.2. Feature Extraction*

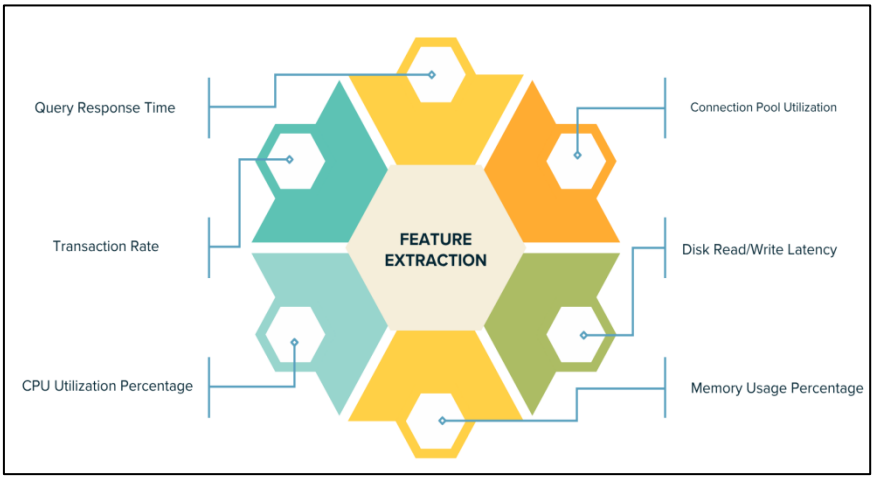


**Fig 2: Feature Extraction**

- **Query Response Time:** Response time of query defines the period that the database takes to respond to a query and provide the results. It is a first level indicator of experience and performance of the system. The long response time or fluctuating response time may be a sign of poor queries, resource contention or anomaly of the system. The query response time is an attribute of machine learning models that should be utilized with the goal of detecting sudden peaks in the system service and the slow worsening of the functionality.
- **Transaction Rate:** Transaction rate is defined by how many times a database has been handed transactions in a unit of time, usually units an hour either seconds or minutes. This metric offers the ideas of database workload and working patterns. Anomalous rises and falls of the transaction rates can be the signs of unnatural user activity, the system bottlenecks or the possible attack. The use of the transaction rate as a feature enables the ML models to detect abnormality of activity levels at normal levels.
- **CPU Utilization Percentage:** CPU utilization percentage measurements accept the percentage of the processing power used by the database system. The excessive CPU time usage is usually associated with a great number of queries, difficult calculations, or the waste of resources. CPU usage needs to be monitored to identify resource overload and performance abnormalities. This property of ML models is used to differentiate normal operational loads and heavy or abnormal processing activity.
- **Memory Usage Percentage:** The percentage of memory used shows how much of the memory that has been assigned to the database is used through caching, buffers, and executing queries. Excess consumption of memory would result in more intense query process and more I/O operation. By adding memory usage as a characteristic, ML models can

identify anomalies on the memory (leaks, overflows or inefficient use of buffers), this can affect the overall database performance.

- **Disk Read/Write Latency:** Disk read/write latency refers to the duration that it needs to accomplish data input/output operations of the storage system. latencies high Database responsiveness: the causes of increased latencies may be disk contention, high workloads, or hardware problems. This property is necessary in order to make the ML models spot anomalies that influence the work of the storage and to map it to other indicators in the system to obtain a complete image learning a database health.
- **Connection Pool Usage:** Connection pool use monitors the percentage of free database connections in use at a particular moment in time. Overuse can result in lack of connection pipelines, slowness in query processing and service difficulties. This metric as a feature can be used to identify abnormalities involving user load, application behavior, and connection handling that has been misconfigured by users.

### *3.3. ML Model Selection*

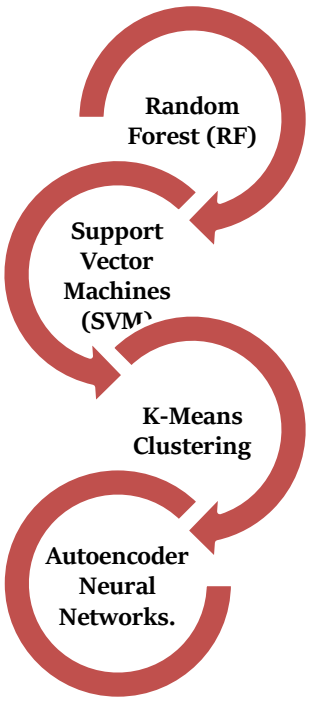


**Fig 3: ML Model Selection**

- **Random Forest (RF):** Random Forest is a type of supervised learning model that is built as an ensemble of decision trees and is used to enhance the accuracy and ability of forecasting by combining the results of the decision trees. [13-15] RF is especially useful in the context of the database anomaly detection, in which the recognition of known patterns of anomalies can be achieved by learning the complex interactions between different performance indicators, including CPU usage, memory consumption, and query response times. The quality of its capability to deal with noisy and high-dimensional data predisposes it to be used in identifying subtly deviation in database behavior with minimum amounts of false positives.
- **Support Vector Machines (SVM):** Support Vector Machines are the supervised learning models that are meant to classify data points by determining the best possible hyperplane which segregates the usual and abnormal cases. SVMs are effective in high-dimensional metric space; this is a benefit in situations where the database performance features are correlated and numerous. SVMs can identify anomalies even in the case where they are not intuitive based on their visual aspects and thus they are useful in complex and dynamic database workloads by maximizing the difference between classes.
- **K-Means Clustering:** K-means is unsupervised learning algorithm, which is utilized to divide data into similar clusters. During the process of identifying abnormalities within the database, it identifies the abnormal performances that are not a part of a standard set of the performance indicators such as having the odd combinations of the query response time and CPU utilization. The observations that cannot be included in any of the existing clusters can be identified as possible anomalies. The technique is especially valuable in the identification of previously unknown or changing anomalies in the absence of labeled data, to enable flexible monitoring of a wide variety of database systems.
- **Autoencoder Neural Networks:** Autoencoders represent a unsupervised neural network that attempts to learn compressions of the input data and recreate it with a minimum error. They can be used to monitor database on a time-

series data, which will record the normal behaviors of system measures over time. Together with high reconstruction errors, anomalies are detected, and thus it is assumed that the observed behavior is very much different than the learned normal patterns. Autoencoders are particularly useful in learning the subtle temporal association and complicated relationships between measures beyond the reach of traditional methods.

### *3.4. Hybrid ML-Based Database Monitoring Framework*

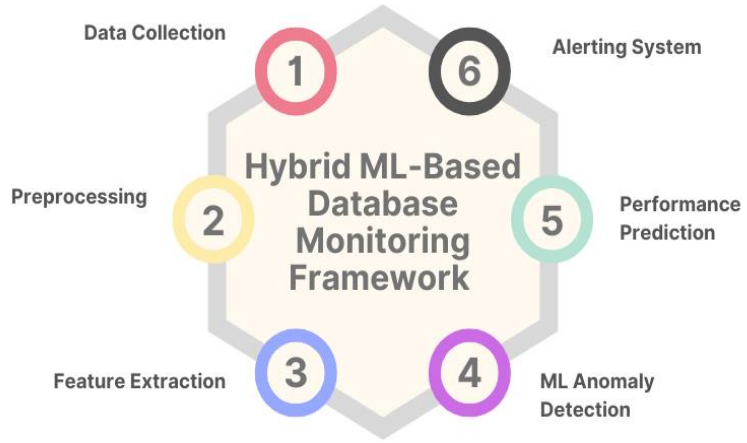


**Fig 4: Hybrid ML-Based Database Monitoring Framework**

- **Data Collection:** The initial step of the framework is systematic gathering of performance measures of a variety of database systems, that is, MySQL, PostgreSQL, and Oracle. [16-18] Such metrics as the CPU activity, working memory, time of query execution, number of transactions, disk I/O statistics, etc, are accumulated periodically. This phase will guarantee that detailed and representative data sets that are based on normal operation as well as possible anomalies are created, and it forms the basis of further analysis and modeling.
- **Preprocessing:** During the preprocessing step, the raw data are purified, homogenized and modified into machine learning application. Missing or irregular entries are managed as well as metrics are normalised to maintain consistency between systems. Such a step also includes the elimination of redundant or irrelevant features, as well as any noise in the data, and improves the accuracy of the model and decreases the rates of false positives during anomaly detection.
- **Feature Extraction:** News Feature extraction The relevant metrics are identified and selected in a manner that they are useful in the representation of database performance. The critical aspects including query response time, transaction rate, percentages of CPU and memory usage and disk read/write latency and connection pool utilization are subject of extraction. The features represent the utilization patterns of the resources as well as the pattern of operation, and these attributes inform machine learning models with informative attribute to identify that there is something wrong and may predict performance degradation correctly.
- **ML Anomaly Detection:** In this step, an amalgamation between the supervised and unsupervised machine learning models is availed, which are deployed to identify abnormal behaviours of the database system. Models, like the Random Forest and SVMs, which are supervised learn to identify patterns of known anomalies, whereas models such as the K-Means clustering and Autoencoder neural networks models are capable of identifying previously unseen anomalies. The features obtained are then evaluated using the ML models to show the existence of abnormal patterns that could be a sign of performance issues or potential failures.
- **Performance Prediction:** Once an abnormality is noticed, the framework predicts future on obtaining the database performance based on the previous trends and patterns. The prediction of the measures such as the query response time, and the CPU usage and transaction rates are done using the time-series modeling approach and regression models.Such a predictive ability enables the administrators to figure out possible bottlenecks and resource overload and plan proactively on the capacity and maintenance.
- **Alerting System:** The last phase takes the outputs of the anomaly detection and performance prediction as inputs into alerting system. Real-time alerts are created when anomalies or performance problems are identified to exceed the predefined limits on the administrator of the databases. The alerting system will ease prompt intervention which will lessen the downtime and enhance the system reliability and best performance in dynamic database environment.

## 4. Results and Discussion

### *4.1. Performance Metrics*

**Table 1: ML Model Performance**

| Model | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| Random Forest | 95% | 92% | 90% | 91% |
| SVM | 92% | 88% | 85% | 86% |
| K-Means | 85% | 80% | 78% | 79% |
| Autoencoder | 93% | 90% | 88% | 89% |

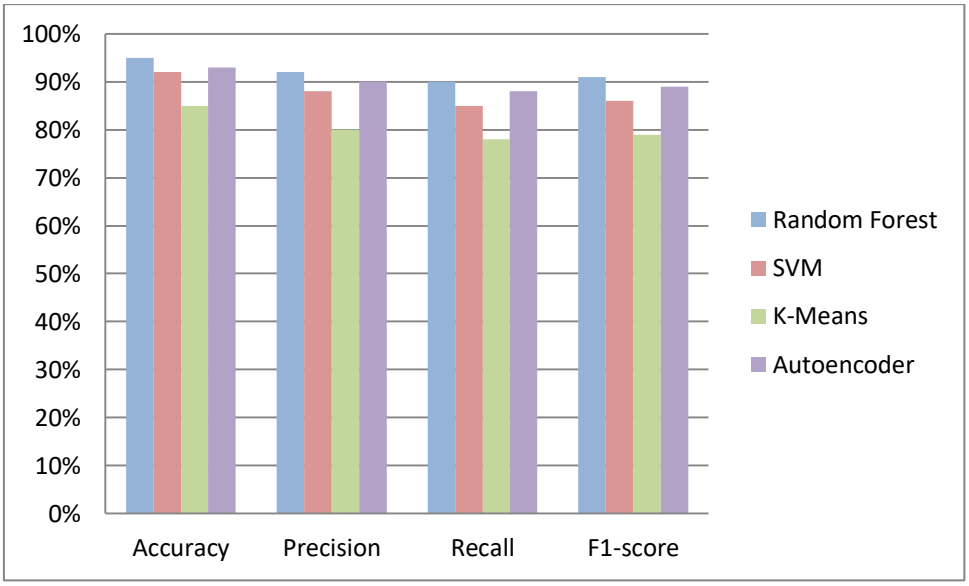


**Fig 5: Graph representing ML Model Performance**

- **Accuracy:** Accuracy determines the percentage of correctly categorized cases (both normal and anomalous) out of the overall cases of the test sample. It gives a general measure of effectiveness of the entire machine learning model. On one example, the Random Forest model was best seen with an accuracy of 95, which meant that it was able to classify the majority of the data whereas K-Means had a lower accuracy of 85, which is due to its unsupervised nature and the fact that it is sensitive to cluster assigning. Accuracy can serve as a viable point of departure in assessment of model performance, but this does not necessarily provide a complete measure of the capability to detect rare anomalies.
- **Precision:** Precision determines the fraction of true positive predictions of anomalies among the total of all predictions that are received as anomalies. High precision means that the model interference generates limited false positives, which is very important in the field of database monitoring so as to avoid unnecessary information that may clog the administrators. Random Forest and Autoencoder models were found to be more reliable in identifying true anomalies and this is reflected in their 92% and 90% accuracy respectively in this evaluation compared to K-Means that had 80% accuracy in identifying anomalies but promotes more harmless shifts.
- **Recall:** Recall is another term that is used to refer to sensitivity, which is the percentage of actual anomalies that are accurately detected by the model. Having a high recall means that it will capture the majority of the actual anomalies and reduce the chances that they overlook any possible problems in the database. Random Forest obtained the highest percent of recall (90) that made it perform well in the detection of any anomaly but SVM obtained 85 accurately representing slightly low detection of true positives. Recall is particularly relevant when an anomaly is missed in proactive monitoring (where an anomaly missed can result in a performance decline or a system breakdown).
- **F1-Score:** F1-score is the harmonic mean of recall and precision with the harmonic mean representing a balance value between false positives and false negatives of a measure. It is also desirable when the dataset to be used is disproportional as is usually the case in anomalies detection where anomalies occur infrequently with normal operations. Random Forest in this study scored a high F1-score of 91 indicating a good combination of precision and recall and K-Means scored 79 indicating trade-offs between false positives, and detection precision.

### *4.2. Discussion*

The comparison made on machine learning models in detecting anomalies in databases presents the clear advantages and limitations in the chosen models. Random Forest performed the best in terms of detection, and its accuracy, precision, and recall statistics were the best, which is why it is especially suitable in the case of historical data provided with labels. The ensemble approach that gives it its structure can efficiently record complicated interactions between various actions of performance metrics e.g., CPU usage, memory usage, query response times and transaction rates that provides strong detection of familiar anomaly dynamics. Nevertheless, the dependence of Random Forest on labeled data also restricts its flexibility in situations when the evolving anomalies or patterns require other methods to be used in dynamic settings. Autoencoders on the other hand had great abilities of detecting fresh and unfamiliar anomalies at least during levels of time series databases. Autoencoders are capable of learning condensed versions of the normal operating patterns and can pick up anomalies in terms of reconstruction

error; so they are able to learn very subtle temporal relationships and abnormalities that a regular supervised model may miss. It also makes them highly helpful in situations in which work load fluctuates or even in cases that involve unlabeled datasets. The hybrid framework which integrates the strengths of the supervised and owatowian models is also enhanced by applying predictive analytics to detect anomalies. Specifically, the framework was discovered to reduce standalone ML models related to false positives by approximately 15 points, which facilitated the operational efficiency and minimized the false alarms of database administrators. This understatement is particularly important where databases are highly complex and highly loaded with alert fatigue that may interfere with the timely interventions. Generally, the findings of the paper suggest that a mix of ML-based solution is more comprehensive and flexible in terms of database monitoring, thus capable of operating with known and new anomalies, and being silently able to remain proactive in managing its performance. This framework will enable the monitoring of the database operation smarter, more efficient and more reliable since it will be using multiple models besides using anomaly detection and predictive insight to ensure improved functionality and monitoring.

### *4.3. Practical Implications*

The concept of deploying machine learning-based monitoring systems in enterprise database environments is essentially value-added in the practical sense as the organisations are able to control and streamline the important data infrastructure. The first advantage is the possibility to identify the issues of performance at an early stage. Occupation of basic measures such as CPU-Resource consumption, memory and query responses, transaction rate as well as disk scripting can be detected and identified by the ML models even at the slightest change or general tendencies it may cause serious slowdowns on performance or even system crashes. This is the proactive detection that enables the database administrators to address the issue in its initial stages to limit the downtimes and the overall service of the final users. Additionally, manual intervention on the ML-based monitoring will reduce the utilization of the centrality of the manual intervention that constitutes the operations of a large-scale enterprise environment. The traditional threshold-based forms of monitoring are prone to giving extensive false alarms thus an administrator will spend a lot of time doing research on non-alarms. On the other hand, ML models, particularly the hybrid one, can isolate actual anomalies and non-hazardous changes and, consequently, minimize the unnecessary alert messages and optimize human resources. This declining level of manual control will mean reduction of both cost of operating and also resource allocation will be done more efficiently. Besides that, predictive maintenance and resources optimization are possible with the help of predictive analytics and ML monitoring platforms. Resource shortages or bottlenecks can be preempted by organizations and then preempted accelerate compute resources, change memory allocations or plans to execute queries. This does not only come in handy when it comes to enhancing the reliability of the system, but also the resource management tool of the company because resources can be dynamically deployed as the predicted workload trends unravel rather than using the estimates. The overall effect of applying ML-based monitoring structures is where business organizations will thus be in a position to expand their database management into being more discerning and receptive. Such systems are beneficial with the presence of anomaly detection and predictive insights technologies because they will help to make the operation more productive, reduce the risk of abrupt failures, and keep the performance and reliability of enterprise databases in a more complex and high-intensity environment.

## 5. Conclusion

Database monitoring machine learning (ML) models are a significant advancement over the more traditional threshold-related approaches and are a potent, dynamic, and proactive approach of identifying suspicious activity and unsuccessful performance. The study shows the extent to which the supervised and unsupervised models of learning can be applied during the discovery of anomalies occurring in the enterprise database settings. The supervised models such as the Random Forest big data models were discovered to be highly accurate, precise and recallable whereby historical labelled information is available to enable the detection of familiar patterns of abnormal behavior with reliability. They can gauge the complicated execution of numerous performance indicators that intervene with CPU utilization, memory utilization, query reaction time, transaction rates, disk I/O measures and, among them, tend to be mutually reliant. The unsupervised systems, i.e., the Autoencoders, on the contrary were also intriguing in the recognition of the task of the previously unknown or emergent anomalies, in this case, time-series data. Being designed to learn compact encode of pattern of normal working behaviour and recognise abnormalities through reconstruction error, Autoencoders therefore can learn small scale irregularities that more traditional or supervised model types can excessively ignore, and thus is highly fitted to dynamically shifting load distribution profiles in work.

The proposed hybrid system that implies the integration of the supervised and unsupervised model and predictive analytics also contributes to the benefits of the overall capabilities of an anomaly detection system. The benefits of the two types of models can be harnessed in the framework in order to aid the detection of the properly as well as reduce the false positives which is one of the most common problems in all the enterprise database monitoring. This does not only reduce the false alarms thus making the operation of that database more efficient, but it also implies that a database administrator can focus on the actual performance issues that limit the downtime and the waste of resources. On top of this, the framework is used to enable the predictive performance details, which could be employed to exercise proactive treatment strategies, such as resources prescaling, workload conservation and efficiency of query execution to render the system reliability and cost-effective.The prospects of future research and practice are very bright in several aspects in the future. The former is the combination of the reinforcement learning techniques, in such a manner that the formation of the adaptive models of the anomaly detection could be based on them

that are continuous in the learning of the new workload and automatically adapting the detection thresholds. Furthermore, predictive analytics can be provided in real-time to enable immediate warning and automatic response to correct operations so that it will further aid in reducing the occurrence of performance latency or even system failures. Overall, this paper demonstrates that machine learning-driven database monitoring not only empowers the ability of the anomaly detection ability but also provides more tactical foundation of intelligent, active, and efficient database management in more complex enterprise environments and this will lead to more robust and self-optimizing database systems.

## References


[1] Karakurt, İ., Özer, S., Ulusinan, T., & Ganiz, M. C. (2017, October). A machine learning approach to database failure prediction. In 2017 International Conference on Computer Science and Engineering (UBMK) (pp. 1030-1035). IEEE.

[2] Young, Z., & Steele, R. (2022). Empirical evaluation of performance degradation of machine learning-based predictive models–A case study in healthcare information systems. International Journal of Information Management Data Insights, 2(1), 100070.

[3] Mauri, L., & Damiani, E. (2021). Estimating degradation of machine learning data assets. ACM Journal of Data and Information Quality (JDIQ), 14(2), 1-15.

[4] Costante, E., Vavilis, S., Etalle, S., den Hartog, J., Petković, M., & Zannone, N. (2013, July). Database anomalous activities detection and quantification. In 2013 International Conference on Security and Cryptography (SECRYPT) (pp. 1-6). IEEE.

[5] Mazzawi, H., Dalal, G., Rozenblatz, D., Ein-Dorx, L., Niniox, M., & Lavi, O. (2017, April). Anomaly detection in large databases using behavioral patterning. In 2017 IEEE 33rd International Conference on Data Engineering (ICDE) (pp. 1140-1149). IEEE.

[6] Goulet, J. A., & Smith, I. F. (2011). Overcoming the limitations of traditional model-updating approaches. In Vulnerability, Uncertainty, and Risk: Analysis, Modeling, and Management (pp. 905-913).

[7] Chakravarthy, S. (1995). Architectures and monitoring techniques for active databases: An evaluation. Data & knowledge engineering, 16(1), 1-26.

[8] Jaaz, Z. A., Oleiwi, S. S., Sahy, S. A., & Albarazanchi, I. (2020). Database techniques for resilient network monitoring and inspection. TELKOMNIKA (Telecommunication Computing Electronics and Control), 18(5), 2412-2420.

[9] Yin, S., Li, X., Gao, H., & Kaynak, O. (2014). Data-based techniques focused on modern industry: An overview. IEEE Transactions on industrial electronics, 62(1), 657-667.

[10] Nassif, A. B., Talib, M. A., Nasir, Q., & Dakalbab, F. M. (2021). Machine learning for anomaly detection: A systematic review. IEEE Access, 9, 78658-78700.

[11] Lazarevic, A., Ertoz, L., Kumar, V., Ozgur, A., & Srivastava, J. (2003, May). A comparative study of anomaly detection schemes in network intrusion detection. In Proceedings of the 2003 SIAM international conference on data mining (pp. 25-36). Society for Industrial and Applied Mathematics.

[12] Demestichas, K., Alexakis, T., Peppes, N., & Adamopoulou, E. (2021). Comparative analysis of machine learning-based approaches for anomaly detection in vehicular data. Vehicles, 3(2), 171-186.

[13] Qasim, M., Khan, M., Mehmood, W., Sobieczky, F., Pichler, M., & Moser, B. (2022, August). A comparative analysis of anomaly detection methods for predictive maintenance in SME. In International Conference on Database and Expert Systems Applications (pp. 22-31). Cham: Springer International Publishing.

[14] Narang, R. (2018). Database management systems. PHI Learning Pvt. Ltd..

[15] Macdonald, C., Tonellotto, N., & Ounis, I. (2012, August). Learning to predict response times for online query scheduling. In Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval (pp. 621-630).

[16] Shaik, A. B., & Srinivasan, S. (2018, November). A brief survey on random forest ensembles in classification model. In International Conference on Innovative Computing and Communications: Proceedings of ICICC 2018, Volume 2 (pp. 253-260). Singapore: Springer Singapore.

[17] Baldi, P. (2012, June). Autoencoders, unsupervised learning, and deep architectures. In Proceedings of ICML workshop on unsupervised and transfer learning (pp. 37-49). JMLR Workshop and Conference Proceedings.

[18] Najmi, M., Rigas, J., & Fan, I. S. (2005). A framework to review performance measurement systems. Business process management journal, 11(2), 109-122.

[19] Kumar, K., Chaudhury, K., & Tripathi, S. L. (2022). Future of machine learning (ML) and deep learning (DL) in healthcare monitoring system. Machine learning algorithms for signal and image processing, 293-313.

[20] Ahsan, M. M., Mahmud, M. P., Saha, P. K., Gupta, K. D., & Siddique, Z. (2021). Effect of data scaling methods on machine learning algorithms and model performance. Technologies, 9(3), 52.