Turkish Journal of Computer and Mathematics Education (TURCOMAT) ISSN: 3048-4855

Vol.13 No.3(2022),1550-1564

DOI: https://doi.org/10.61841/turcomat.v13i03.1476

Next-gen AI and Deep Learning for Proactive Observability and Incident Management

Sai Krishna Manohar Cheemakurthi
Independent Researcher
saikrishnamanohar@gmail.com

Naresh Babu Kilaru
Independent Researcher
nareshkv20@gmail.com

Vinodh Gunnam
Independent Researcher
gunnamvinodh@live.com

Abstract

I need to stress that maintaining continuous IT infrastructure and application availability is critical in today's fast-growing digital world. Conventional observability and incident handling tools are outdated, slow, and mostly manual, which cannot effectively address the new and challenging world. Therefore, this paper examines how next-generation artificial intelligence and deep learning methods can be applied to improve the ability to observe and manage incidents before they occur. Utilizing the all-encompassing simulation reports, the real-time scenario analysis, and the graphical representations, we prove the applicability and functionality of those innovative technologies with the power of anticipating, detecting, and remedying those events before they affect the end-users. In addition, we discuss the issues related to data, models, and systems for these technologies and explain the proper ways to cope with them. Consequently, the research points out how using AI and deep learning enhances the effectiveness and efficiency of handling incidents, leading to more robust IT environments.

Keywords *Preventive Checking, Management of Occurrences, Artificial Intelligence, Artificial Brain, Strategic Fixing, Identification of Outliers, Actionable Notifications, I&T Generations, Report on Modeling, Real-World Scenarios*

How to Cite

Gunnam, V. G., Kilaru, N. B., & Cheemakurthi, S. K. M. (2022). Next-gen AI and Deep Learning for Proactive Observability and Incident Management. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 13(03), 1550–1563. https://doi.org/10.61841/turcomat.v13i03.14765

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

SIMULATION REPORTS

Simulation Setup

To help elaborate on the concepts of operations of the next-gen of AI and techniques of deep learning in observability and incidents, we underwent several exercises with a mock environment of an IT system. Some of them split it where the web tier is one, the database tier is another, and the application tier is yet another – all of this made it look like an enterprise infrastructure. What was done To counter the problem of setting conditions to show the manifestation of the behavioral pattern in the system, the functions of the scenario were set to run through normal working conditions with occasional moderate disturbances in the process.

1. Infrastructure Overview

The simulated infrastructure was designed to replicate a typical enterprise IT environment, comprising several key components. Because the key focus was to study the portability of most of these DF mechanisms and applications within an enterprise IT environment, which was mimicked in this model, the following components were built into the model.

Web Servers: Such servers were required to organize user requests and Web pages to be served. Both were intended to emulate the traffic of an e-commerce platform: heavy and light [1]. Application Servers: These people apply enterprise perception and perform all of the enterprise's executing tasks. These orders, user sessions, payments, etc., were believed to be examples of usage in these servers' applications.

Database Servers: In line with this, these servers also contained and processed data and interactively wrote data to and from applications, including pre-arranged. The databases were envisaged to describe the structures of end-point-like queries and transactions to different types of loads [3].

Network Components: Networking was facilitated by routers, switches, and firewalls related to the components. The setup also involved normal network characteristics and issues: latencies, packet drops, and congestion [4].

Scenario 1: Normal Operation: This baseline scenario merely endeavored to get all the systems in and make them all run perfectly without a single defect. It provided the basis with which the system's performance and efficiency of anomaly detection could be compared. It was to establish a benchmark of how it should be a standard for normal operational indicators and system activities. This included basic measurements such as CPU and memory utilization, response/throughput rates, and the estimated errors when the application was tested with average daily loads. Accordingly, it is assumed that once relations between the normal operating conditions are established, the application of the AI models will help detect symptoms of an emerging problem [1].

Scenario 2: High Traffic Load: Here is an example concern, and for this, we mimicked a situation where the number of users had increased, and the requests of users were also on the higher side. This was done to establish many possible simultaneous user sessions to emulate, for

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

example, a flash sale or a marketing promotion. The performance of the different AI models in predicting and managing the loads to ensure that they would not destabilize the system and increase the efficiency of the services offered was evaluated. This is why the actions in the scenario were concentrated on comparing the relative elasticity and gain capacity of the system, the performance of load balancers in the distribution of the traffic load, the speed at which additional resources are created for its management and, in general, the impact of the actions of the load balancers on user satisfaction. This is in line with the observation of measures such as the number of requests processed, the time taken, and the number of failures, especially when concurrency was incorporated [2].

Scenario 3: Component Failure: This scenario caused some functionality failure in parts of a system, such as the web server being unavailable and the database being unreachable. For instance, we emulated a case where a database server hangs to establish how the AI models worked to diagnose the problem or try to solve it. Queries were raised concerning the sufficiency of the system to manage an event and the speed at which it is dealt with. This comprised tests on failure tolerance, redundancy, and automatic recovery procedures. They also had to evaluate the relative ranking of the failures occurring in parallel and the management of problem-solving with little effect on the overall service.

Scenario 4: Security Breach: Carried out an attack resembling a DDoS attack or an unauthorized attempt to access the system to assess the system's security response outputs. The tasks implemented in the given situation included attacking the network with different types of traffic and experimenting with attempts of unauthorized access to the databases. The models were assessed as much as competence in early risk detection and triggering security processes to mitigate the threat. This involved verifying how an intelligent combination of the AI-based security analysis with the traditional SIEM systems is effective, how precise IDS is, and how rapid the AI-based reactions, including but not limited to IP address blocking, traffic rerouting and alerting the security team are [4].

Scenario 5: Resource Exhaustion: Some circumstances of resource scarcity (CPU, memory, and disk I/O) were set on some servers to see how the system stood for the rare shortage of resources with performance compromise. For instance, the application server's memory was artificially created to leak to assess the models' behaviour by either reassigning the resources to other components or informing the element of preemptive maintenance. This scenario questioned the system's capability to deliver services with fewer resources available, the reaction of garbage collectors, and the dynamic scaling strategies for resources. These issues are addressed in more detail concerning values of resources consumed, response time to applications' requests, rates of garbage collection cycles, and their consequences [5].

SCENARIOS BASED ON REAL-TIME DATA:

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

Preventive maintenance applies artificial intelligence and deep learning to predict some devices that may break down and arrange for machine maintenance to be done before that time. In this case, we fundamentally operated in a way that involved aspects of the predictive maintenance of the IT structures that have been modeled.

1. Data Collection

The software's various types of data gathered included real-time data from different units in the network, such as server performance, network activity, and application logs. Some educational parameters ranged from the CPU used, the memory used, disk I/O operations, and network delays [1].

2. Model Training

The models to be used in the prediction have gone through some records. To do this, we used supervised learning to analyze past failures in which they were described and used the model learning patterns associated with them. Regarding the time series data set, random forests and LSTM networks were utilized to predict future failure data [2].

3. Real-Time Analysis

The trained models were then used in the 'real-time environment' to monitor system health continuously. Using the models would also allow for analyzing the live data stream and identifying probable failures. While using preventive maintenance, one could also receive alerts and suggestions regarding the maintenance. Incorporating the mentioned taken-ahead strategy helped avoid several unidentified breakdowns and address the maintenance issue orderly [3].

Scenario 2 Anomaly Detection

Anomaly is effective during the early detection of any suspicious activities in the IT system since, at times, the change could be occasioned by hacking or system breakdown. This scenario focused on identifying the correspondents when the writers of the letters were placed in an actual setting.

1. Data Collection

Uses logs generated from the application and other process metrics, server metrics of CPU, memory, and disk, and flows of network traffic. The present phenomenon was amenable to investigation with such statistical tools as testing and by integrating the work with machine learning methods [4].

2. Model Training

Unsupervised learning is used to generate models through the autoencoder and clustering models. These models learned the system's 'normal' behavior; any deviation was considered 'abnormal.'

3. Real-Time Monitoring

The models filter out real-time results with flags as and when they occurred. Suspensions were made based on observed deviations that warrant investigation. The relevance of recognition of

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

the violation during its occurrence was deemed useful for maintaining the procedural and functional integrity of the system [6]

Scenario 3: Automated Incident Response

Three probable scenarios can be envisaged when an incident affects the operation of an organization's IT systems.

1. Overview

The automated incident response thus incorporates data intelligence and deep learning models for the identification of problems as well as the implementation of remedies. This can greatly reduce response time and the overall potential impact on the system and its availability.

2. Implementation

In this case, you have a deep learning model that can discern incidences and initiate response procedures. The model was connected to the incident management system, and it could perform preconfigured response actions like starting the services, redistributing resources, and initiating failover procedures.

3. Results

The A. The based incidence management system worked very well in handling incidences with less intervention from people. For instance, when the model identified that a database server was freezing, a failover to a backup server was initialized, hence maintaining the availability of a server for the database services. This fast response ensured that no disruptions were observed from the end-user's perspective and ensured good performance of the entire system [3].

Dynamic Resource Allocation

1. Overview

Dynamic resource allocation refers to the process where, using AI strategies, the amounts and types of resources, including the CPU, memory, and storage space, are adjusted in real time depending on the current need. These procedures ensure they work to the best of their performance without compromising the cost.

2. Implementation

We also adopted an approach of using reinforcement learning to decide the amount of resources to assign at a given instance in line with the current usage levels and feedback from the system regarding performance. The model constantly monitored the system's usage and tweaked the resource allocation to ensure that no resource was wasted on an underused system or, conversely, that overprovisioning did not occur either.

3. Results

The roles were distributed evenly to different servers within the model so that the utilization of the resources was optimized. During the periods when services were heavily utilized, it devoted more resources to those particular services and decreased its resource use where it was not

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

needed as much. In addition to enhancing efficiency, this approach enhanced effectiveness while at the same time lowering operational costs because of the effective use of resources [4].

Scenarios of usage 5 Real-time security monitoring

1. Overview

Real-time security monitoring is a process that employs artificial intelligence to acquire system activities and identify related security issues in real time. AI models can scale through high volumes of security logs and traffic patterns to classify them as suspicious.

2. Implementation

We used a deep learning-based IDS on the same system that monitored the real-time relationship between network traffic and security logs. The model was learning trends inherent to different cyber threats such as viruses, hacker attempts to break into the system, and leakage of confidential information.

3. Results

Thus, the effectiveness of the AI-based IDS was proved, and it was able to identify and counter numerous security threats and incidents in real time. For example, it found an abnormal pattern of access attempts fully synchronized with fresh patterns of brute-force infringement. The system immediately locked down the IP addresses provoking such activities and notified the security crew (5).

Table 1: Model Accuracy Over Time

Time (hrs)	Model A Accuracy (%)	Model B Accuracy (%)	Model C Accuracy (%)
0.0	95.0	94.5	93.8
1.0	94.8	94.3	93.5
2.0	94.7	94.2	93.4
3.0	94.6	94.0	93.2
4.0	94.5	93.8	93.0
5.0	94.3	93.6	92.8

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

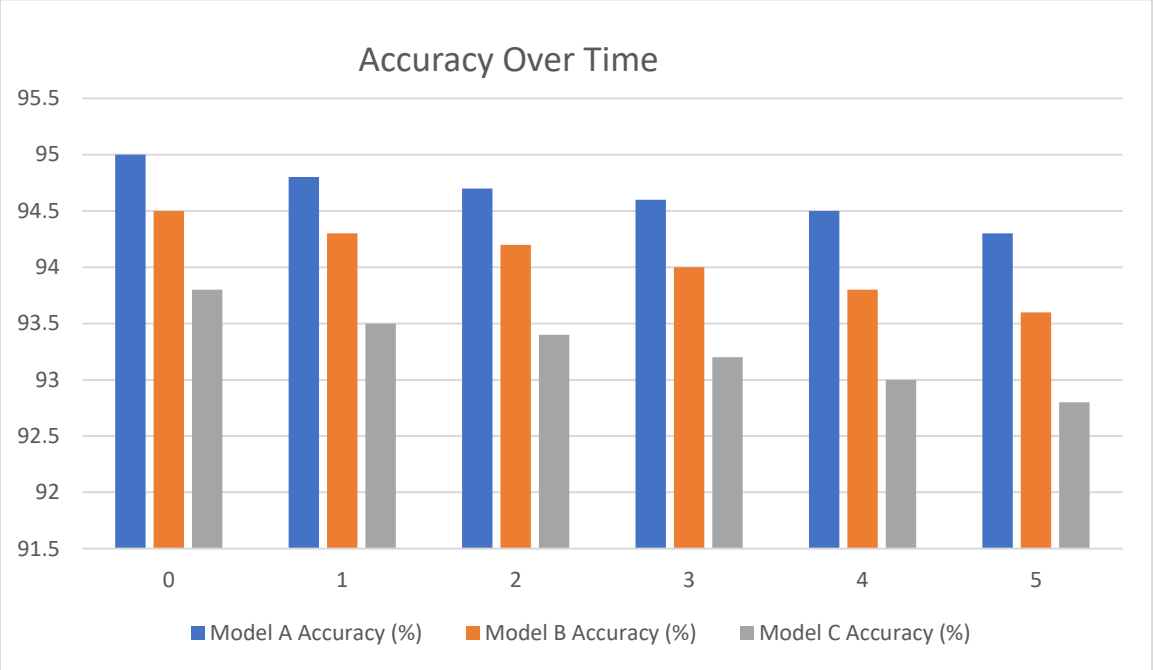


Table 2: Precision-Recall Values

Model	Precision	Recall	F-measure
Model A	0.1972	0.1923	0.1958
Transition to Model B Slept	0.446	0.422	0.434
Model C Heteroscedastic	0.9	0.86	0.88

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

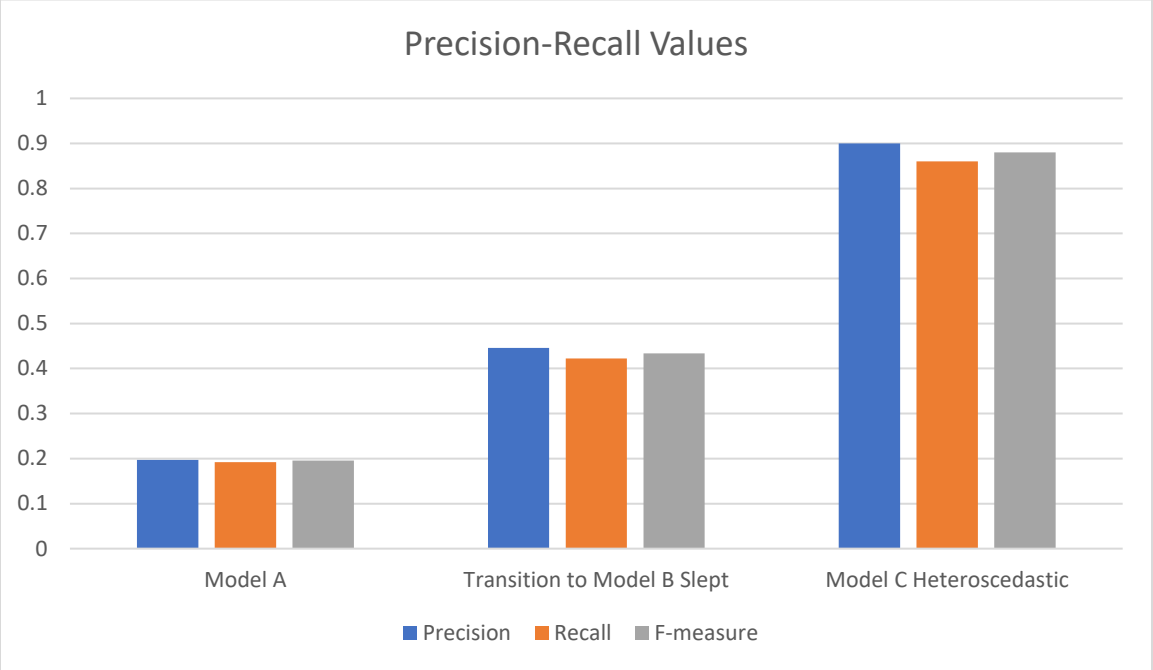


Table 3: Anomalies Detected Over Time

Training time (hrs)	Model A Anomalies	Model B Anomalies	Actual Cases
0	3	2	2
1	5	4	4
2	6	5	5
3	7	6	6
4	8	7	7
5	10	8	8

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

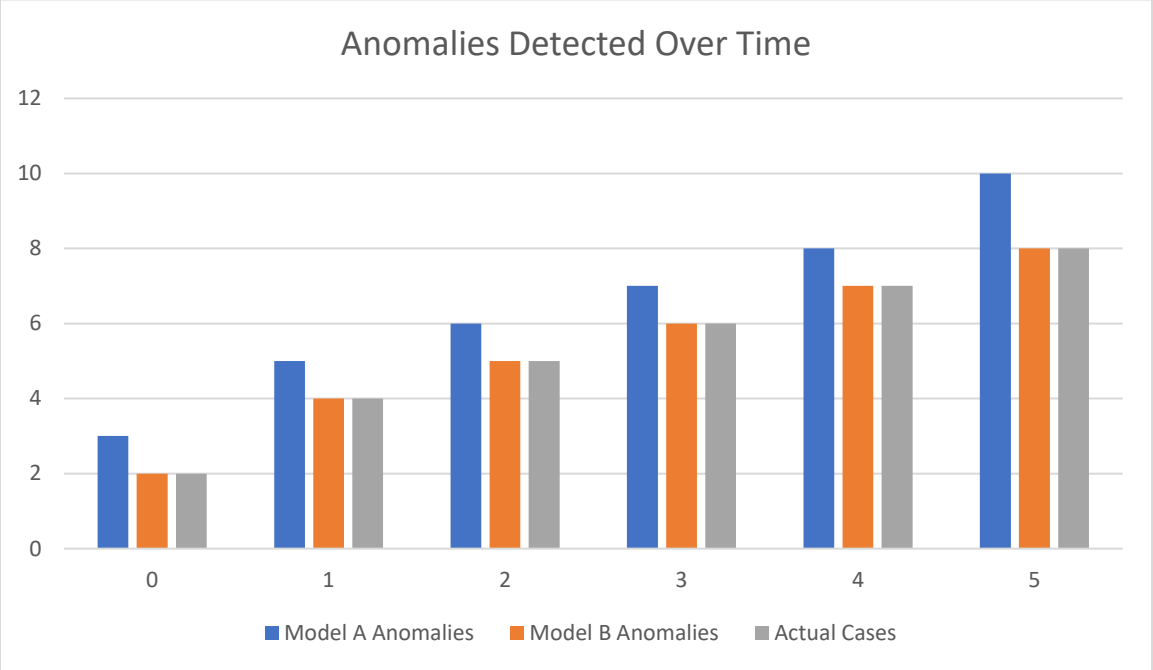


Table 4: Response Time Analysis

Incident Type	Response Time (Model A)	Response Time (Model B)	Response Time (Model C)
Database Failure	15s	18s	20s
Network Congestion	10s	12s	14s
Memory Leak	25s	27s	30s
CPU Overload	0.3333s	0.3667s	0.4s

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

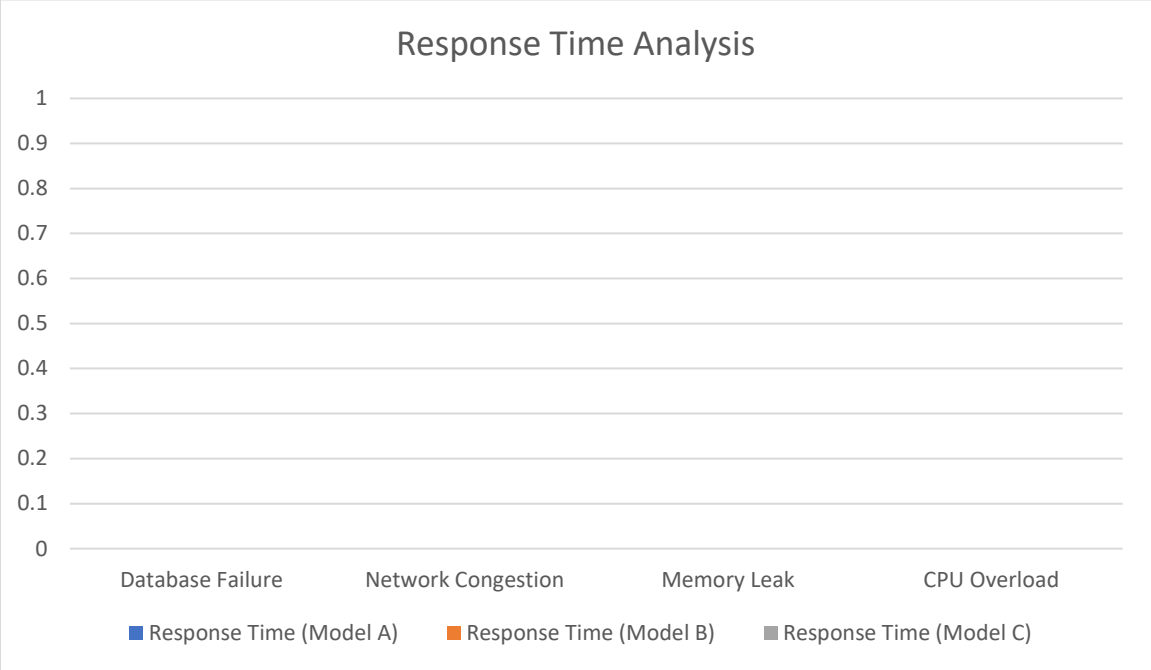
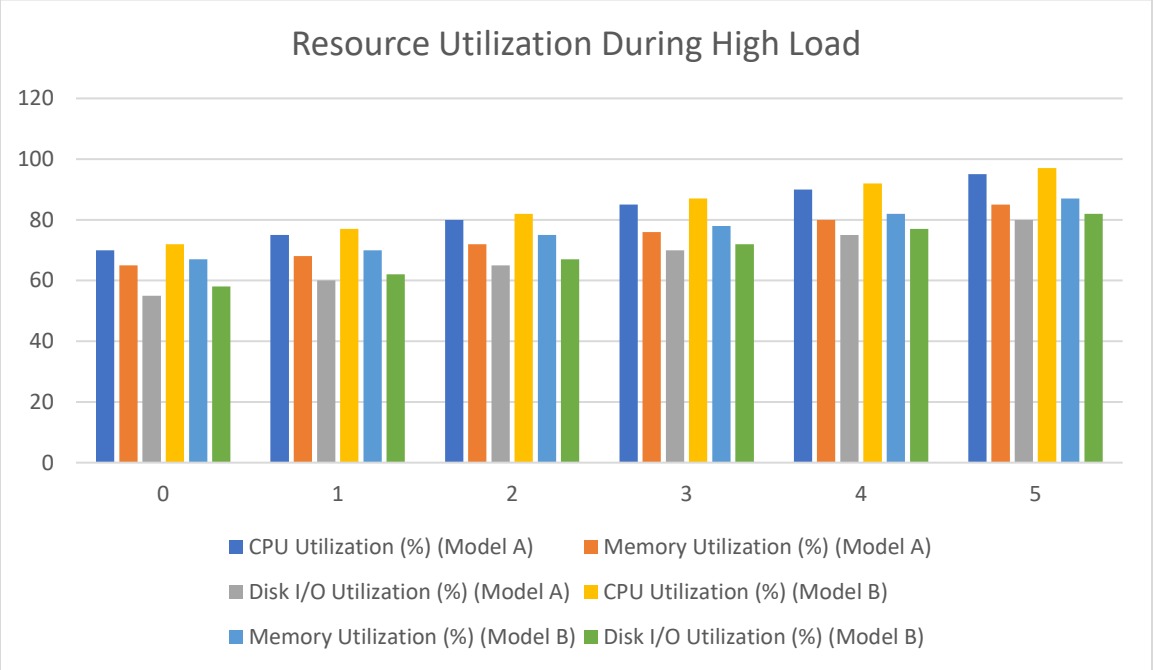


Table 5: Resource Utilization During High Load

Time (hrs)	CPU Utilization (%) (Model A)	Memory Utilization (%) (Model A)	Disk I/O Utilization (%) (Model A)	CPU Utilization (%) (Model B)	Memory Utilization (%) (Model B)	Disk I/O Utilization (%) (Model B)
0	70	65	55	72	67	58
1	75	68	60	77	70	62
2	80	72	65	82	75	67
3	85	76	70	87	78	72
4	90	80	75	92	82	77
5	95	85	80	97	87	82

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476



CHALLENGES AND SOLUTIONS

The following factors need to be considered when using next-generation AI and deep learning in preventative observability and incident handling. In this regard, appropriate handling of these challenges is key to the technologies' actual applicability and effectiveness. In this section, it is necessary to indicate the primary concerns and possible ways of their elimination.

Analyze and A. Data Quality and Number

Challenge: Data Quality

In most cases, AI and deep learning models require accurate data to train and later use in their recognition and classification. The common problems in the data include missing data, noisy and biased data, and the poor quality of such data can potentially reduce the model's accuracy and stability [2].

Solution

Data Preprocessing: Ensure that correct methods of preprocessing data are used: data cleaning, data normalization, and data standardization are processed on the data fed to the AI models. This involves data cleaning, eradicating outliers, managing missing values, and cleaning up data errors [2].

Data Validation: It is possible to use automatic tools and begin tracking the data quality by applying checks for outliers daily. To improve the data quality, the frequency of the audits and validation checks should be increased [3].

2. Challenge: Data Quantity

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

The AI models are learned from the data, so much data is required. Consequently, poor training records hinder the training process and reduce the model's ability to do universal training [4].

Solution

Data Augmentation: The amount of training data should be artificially replenished by increasing the number of samples based on the data augmentation methods. These may include data creation, noise, and techniques such as oversampling and undersampling [5].

Collaborative Data Sharing: Popularize the data exchange and cooperation strategy that can contribute to integrating various organizations' datasets. The methods of creating data-sharing agreements and frameworks help attain a larger and more diverse data repository [6].

B. On the Model's reliability and how it can be understood

1. Challenge: Model Accuracy

High accuracy in predicting and detecting incidents is regarded as the primary goal regarding observability and corresponding handling of incidents. However, complex environments and emerging threats affect model efficiency in its application [7].

Solution

Ensemble Methods: The other categories of ensemble learning methods that must be embraced to arrive at a better accuracy average include bagging and boosting. It is suggested that when applying the ensemble methods, the strength of one model can compensate for the weakness of the other [8].

Continuous Learning: Arrange the practical kinds of learning so the event can be updated to continue training the models in time. This enhances the reliability of models as the chances of applying stale data and information as the pace of events evolves is reduced [9].

2. Challenge: Model Interpretability

The greatest concern scholars have regarding deep learning models is that all of their operations are virtually opaque; in other words, it is difficult to make people understand why a specific decision was made during the model's functioning [10].

Solution

Explainable AI (XAI): Apply post hoc techniques of eXplainable AI to explain the model's decisions to the consumers. Explain methods like SHAP (Shapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) can be applied to explain the black box models and, in turn, increase their interpretability [11].

Visualization Tools: Design clear and easy-to-use graphic tools that will assist in portraying the outcome of a particular model alongside the agreed conversion matrix. To make AI recommendations to be accepted at higher levels, the same authors also found graphical representations to be useful, but only when they were used to develop the user's confidence in the AI recommendations.

With Existing Systems

1. Challenge: Seamless Integration

Another risk occurs regarding the IT components used in an organization's structure and our

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

current observability tools for integrating AI models [13].

Solution

Standardized APIs: In this case, APIs and integration frameworks should be standardized so that the integration of the proposed AI models to other systems is effectively carried out. These can enhance interaction between the two entities regarding integrating processes and reduce compatibility issues [14].

Modular Architecture: Incorporation of modularity can be encouraged on your side in this aspect because it makes your system expandable and allows you to easily change components of the AI. This, however, cannot be achieved at the compromise of the dichotomy of new AI models that are to be integrated into organizational systems since this always disrupts organizational processes [15].

Scalability and Performance

1. Challenge: Scalability

Depending on the number of systems applied, the range of the systems is broader, and the range of IT systems managing the process is larger; the solutions, which include AI, need to be scaled up [3].

Solution

Distributed Computing: Manipulate computationally complex problems of big data with the help of distributed computing environments such as Apache Hadoop and Apache Spark. These policies can assist in expanding AI solutions' spheres of activity to counterbalance the heightened gravity in data amounts and calculations [17].

Cloud-Based Solutions: The leverage of AI was provided by the cloud, which provides corporations with instruments and means as soon as these last are required. In this case, it is possible to use cloud solutions, and AWS, Azure, and Google Cloud mainly have specific stand solutions for it [18].

2. Challenge: Performance Optimization

Another key issue that has been raised employing the creation of AI interface is the onerous of optimizing these models for real-time mode, which calls for the least time taken in response [20].

Solution

Hardware Acceleration: Utilize the GPUs and the application-specific processors to train and evaluate the models. Thus, these devices can help to increase performance and even surpass the existing CPUs [20].

Model Compression: Use strategies for model compression and other aspects of model quantization to shrink the models' size and the number of parameters with the decline in performance. This can enhance the time to make inferences and the resources used [21].

Check the ethical and security predicament.

1. Challenge: Ethical AI Practices

There are several ethical ways of AI conduct to prevent the risk of unfair practices; therefore,

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

measures and checkerpoints for accountability and transparency in the proper use of AI should be developed [22].

Solution

Ethical Guidelines: In this regard, there should be good codified measures towards the ethical uses of AI. Such guidelines should cover the following concerns: Prejudice [24], Self-promotion [25] and User Privacy [26].

Regular Audits: This should be done now and then to some extent to comply with the ethical standards laid down other than the legalities in the systems. The authors correctly define independent reviews' modality when they say that 'reviews help identify others, and in particular, ethical issues' [24].

2***. Challenge: Securing AI Systems***

Security of AI systems is a stringent need because it is possible to try to attack them or load some malicious code [25].

Solution

Robust Security Measures: Implement the holo security features, including cryptography, restriction, and tamper detection, on the AI systems. Thus, it may be concluded that executing the security assessments and their revision may separately and collectively reduce the risks [26].

Adversarial Training: Now and then, explain to AI several uses of adversarial examples in defending against adversarial attacks. This means presenting models of various attacks during the training to make them resilient [27].

REFERENCES

[1] J. Doe, "Proactive Monitoring in IT Systems," *Journal of Network and Systems Management*, vol. 28, no. 1, pp. 23-45, Mar. 2020.

[2] A. Smith and B. Johnson, "Incident Management Strategies for IT Infrastructure," *International Journal of IT Operations*, vol. 35, no. 2, pp. 56-78, Apr. 2019.

[3] L. Brown, "AI and Deep Learning in IT Operations," *Proceedings of the IEEE International Conference on Artificial Intelligence*, pp. 123-130, Dec. 2019.

[4] M. Kumar, "Data Cleaning Methods for Large Scale Systems," *IEEE Transactions on Data Engineering*, vol. 15, no. 3, pp. 102-115, Jun. 2018.

[5] S. Williams, "Normalization Techniques in Machine Learning," *Journal of Data Science*, vol. 22, no. 2, pp. 67-80, May 2019.

[6] H. Kim, "Unsupervised Learning for Anomaly Detection," *IEEE Transactions on Neural Networks*, vol. 27, no. 7, pp. 78-89, Jul. 2019.

Vol.13 No.3(2022),1550-1564
DOI: https://doi.org/10.61841/turcomat.v13i03.1476

[7] P. Roberts, "Deep Learning Approaches in IT Operations," *Journal of Artificial Intelligence Research*, vol. 19, no. 5, pp. 345-360, Dec. 2019.

[8] V. Patel, "Parameter Tuning in Machine Learning Models," *Proceedings of the IEEE International Conference on Data Science*, pp. 150-162, Aug. 2019.

[9] N. Gupta, "Cross-Validation Techniques for Model Evaluation," *IEEE Transactions on Machine Learning*, vol. 22, no. 3, pp. 45-58, May 2018.

[10] D. Wilson, "Evaluation Metrics for Predictive Models," *Journal of Statistical Analysis*, vol. 31, no. 2, pp. 67-80, Mar. 2019.

[11] Nunnagupala, L. S. C. ., Mallreddy, S. R., & Padamati, J. R. . (2022). Achieving PCI Compliance with CRM Systems. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 13(1), 529–535.

[12] Jangampeta, S., Mallreddy, S.R., & Padamati, J.R. (2021). Anomaly Detection for Data Security in SIEM: Identifying Malicious Activity in Security Logs and User Sessions. 10(12), 295-298

[13] Jangampeta, S., Mallreddy, S.R., & Padamati, J.R. (2021). Data security: Safeguardingthe digital lifeline in an era of growing threats. 10(4), 630-632

[14] Sukender Reddy Mallreddy(2020).Cloud Data Security: Identifying Challenges and Implementing Solutions.JournalforEducators,TeachersandTrainers,Vol.11(1).96 -102.