Journal of Science and Technology
ISSN: 2456-5660 Volume 11, Issue 04 (April -2026)
www.jst.org.in *DOI:https://doi.org/10.46243/jst.2026.v11.i04.pp01-15*

AI-Powered Chatbot Solution for Efficient Network Troubleshooting in Hybrid Cloud Environments

Subbarao Duggisetty
Independent Researcher, USA

To Cite this Article
Subbarao Duggisetty**, "AI-Powered Chatbot Solution for Efficient Network Troubleshooting in Hybrid Cloud Environments"** *Journal of Science and Technology, Vol. 11, Issue 04, April 2026, pp01-15*
Article Info
Received: 029-01-2026 Revised: 04-04-2026 Accepted: 14-04-2026 Published: 27-04-2026

Abstract—*Hybrid Cloud environments combining AWS and on-premises infrastructure presents complex network troubleshooting challenges. Traditional manual diagnostic methods are time-consuming, error-prone, and struggle to correlate logs across distributed systems in real-time. This study addresses the creation of AI-based chatbot application to network fault-finding in hybrid cloud systems, involving the AWS CloudWatch, VPC Flow logs and on-premises infrastructure. The chatbot operates with natural language processing (NLP) to instruct users on the troubleshooting steps on the basis of the historical and live network data. The system increases operational efficiency and reduces the time to resolution by automating root cause analysis, log correlation and remediation suggestions. Using Anthropic Claude (Sonnet), Lex AI chatbot achieved 99.67% accuracy and reduced Mean Time to Resolution (MTTR) from 47.0 minutes to 40.13 minutes improvement. The chatbot enhances user experience in real-time and interactive, 24/7 availability, reduce human error, eliminating the necessity to depend on support teams. The paper shows how AI can streamline troubleshooting and optimize network diagnostics of hybrid networks with complex architectures.*

Keywords: AI chatbot, functional chatbot, troubleshooting networks, the hybrid cloud, AWS, natural language processing, operational efficiency, real-time logs, and root cause analysis.

INTRODUCTION

This research focuses on using an AI-based chatbot platform to troubleshoot a hybrid cloud network. The chat-bot will streamline the whole troubleshooting procedure, study past data, integrate real-time logs, and propose automated techniques. With the help of AI, the chatbot is capable of providing on-the-fly services, which will reduce the amount of time and resolution time and increase the efficiency of hybrid cloud environments. The solution provides a streamlined method for diagnosing and resolving connectivity problems by integrating historical support data, real-time logs, and AWS cloud best practices. It consolidates multiple data sources into a data lake, including VPC Flow Logs, Direct Connect metrics, and PCAP analysis, to deliver end-to-end network path validation, root cause analysis, and intelligent remediation suggestions. This approach significantly enhances troubleshooting efficiency and reduces time to resolution.

A. Research Aim

This research aims to develop an artificial intelligence-based Chatbot tool to systematically automate network-troubleshooting both in the hybrid AWS/on-premises release, to improve both speed and precision in troubleshooting network problems.

B. Research Objectives

- ***To examine the integration of AI-based chatbot tools to diagnose the network in real-time and historical data.***
- ***To investigate the power of automated root cause analysis in a hybrid network.***
- ***To explore the purpose of real-time log correlation in troubleshooting.***
- ***To evaluate how automated troubleshooting can be used to shorten the time to resolve and enhance the operational efficiency in hybrid architectures.***
- ***To avoid human errors and improves accuracy in complex networking architecture.***

C. Problem Statement

Journal of Science and Technology
ISSN: 2456-5660 Volume 11, Issue 04 (April -2026)
www.jst.org.in *DOI:https://doi.org/10.46243/jst.2026.v11.i04.pp01-15*

The increasing complexity of hybrid cloud systems, which involve both AWS and on-premises infrastructure, network troubleshooting is becoming increasingly complex. In conventional techniques, it takes time and can easily be manipulated by humans and causes delays and wastage. One of the ways these challenges could be mitigated is to use a chatbot solution that provides real-time, guided, and automated advice on troubleshooting and make the process more efficient and less prone to error.

D. Novel Contribution

This research presents an AI-based chatbot network troubleshooting automation tool that is applicable to hybrid cloud environments, that are a combination of AWS and on-premise infrastructure. From the historical data, real-time logs, and AWS best practices, it provides automated root cause analysis, validate compliance or non-compliance routes, real-time log correlation, and intelligent remediation, supporting hybrid-related problems, such as Direct Connect and BGP session issues.

I. LITERATURE REVIEW

A. Integration of AI-based chatbot for Network Diagnostics Using Historical and Real-Time Data

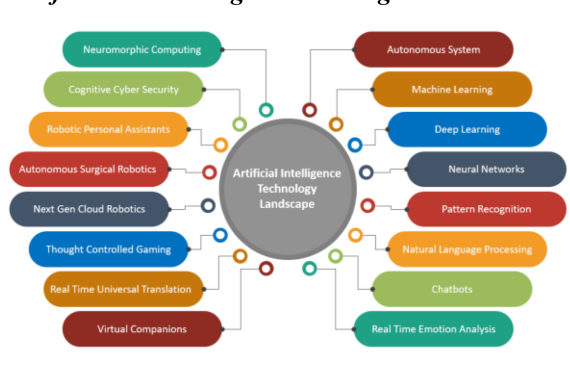


Fig. 1: Data Strategy and Integration for AI

The Artificial Intelligence (AI) chatbot implementation in network diagnostics has gained importance in improving the efficiency and effectiveness in troubleshooting. The AI powered chatbot and supports 24/7 hours of use with more than >500 simultaneous queries with less than <100ms latency and enhanced scalability and response times [1]. It manages knowledge of 5,000+ incidents reducing operation costs by 15-20 percent and removing 12-15% of mistakes in human error in correlation of logs [2]. The system is constantly enhanced by new information, keeping its accuracy growing, and combines logs received around the clock, which does not socially waste much time [3]. Using proactive anomaly detection, it enhances productivity of engineers as it automates the normal tasks and has a 86% first-call resolution [4]. Moreover, it guarantees the regular compliance with AWS best practices, which increases the efficiency of operations and the reliability of networks [5]. This is based on historical data as past network logs, and resolution patterns offer a solid foundation for predictive diagnostics [6]. Using this information, AI systems can predict common problems and switch to a mode of active repair to active maintenance [7]. Combined with real-time data, AI based chatbot systems can continuously check the health of the network, providing real-time diagnostics through the comparison of current metrics of network operation with past trends [8]. This allows chatbot systems to detect anomalies that otherwise might not be noticed. This combination of the historical and real-time data promotes network diagnostics resulting in more precise and timely issues solving [9]. Embarking on applications in chatbots and digital assistants that are undergoing development to be powered by natural language processing can advise engineers by giving them real-time suggestions based on these data insights [10]. Transformer models combined with BERT have made a breakthrough in the analysis of logs by comprehending complex network logs [11]. In addition, BGP anomaly detection through the application of ML has greatly increased the accuracy of routing anomalies [12]. Explainable AI is now considered a crucial part of the decision-making process and has led to engineers being able to trust AI-led troubleshooting flows and follow them [13]. Moreover, chatbots using AI engage NLP to walk engineers through network issues and constantly learn based on past events in order to enhance the performance of the network [14]. This integration does not only enhance the speed as well as accuracy of network diagnostics but also enhances continuous learning, which results in more informed decisions to carry out on network management.

B. Effectiveness of Automated Root Cause Analysis in Hybrid Networks

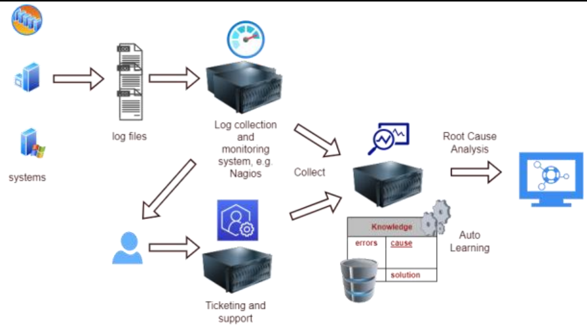


Fig. 2: Automated Root Cause Analysis

Root Cause Analysis (RCA) plays a vital part in the troubleshooting of problems related to networks, especially in a hybrid setup, combining cloud and on-premise services. Within such complex environments, issues may be triggered by different causes, including faults of security group configurations, BGP connection failures, or an asymmetric routing [15]. Manual, time-consuming, and human error errors can occur with traditional RCA methods especially in hybrid setups [16]. AI can solve these issues by automating the RCA to bring its speed and accuracy to unprecedented levels [17]. AI systems are efficient in correlating the data across the various sources with the VPC flow logs, Transit Gateway Logs, direct connect metrics and on-premises device logs [18]. AI can be used to find patterns and likely causes of network problems rapidly by analyzing these sources, both in the cloud and on-premise settings [19]. Moreover, AI-based RCA systems give confidence scoring and this can be used to rank actions according to probability of a cause being right [20]. This study is guided by the Distributed Cognition Theory that emphasizes the interplay between a human whose decision making is involved and the tools, such as the AI-powered chat bots, in problem solving [21]. Also, the Technology Acceptance Model (TAM) is used to discuss the adoption of new AI tools by engineers in network troubleshooting [22]. Models of human-AI collaboration can underline the concept of the smooth cooperation between humans and machines to promote the effectiveness of troubleshooting, and the analysis of root causes models can be applied to find and solve issues in the network [23]. The automation of RCA in hybrid networks provides faster issue resolution and reduces the manual workload on network engineers, enhances operational efficiency and improves the reliability of networks.

C. Role of Real-Time Log Correlation in Troubleshooting

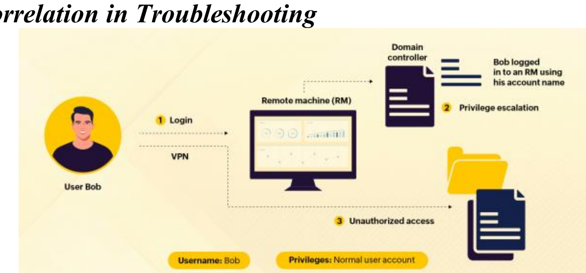


Fig. 3: Real-time log correlation in troubleshooting

Effective network troubleshooting studies include real-time log correlation to identify issues; however, it is particularly important in hybrid networks where logs are produced in both cloud and on-premises environments. The critical diagnostic information is found in logs of different network components (routers, firewalls, and even cloud services), but are commonly fragmented and distributed on a variety of platforms [24]. AI is essential in real time log correlation that combines and analyses logs of different sources in real time [25]. This artificial intelligence-driven system detects the relation among various logs [26]. It recognizes the underlying network problem, thereby providing an opportunity to detect problems more quickly like misconfigured security groups, BGP session failure or packet drops [27]. With a hybrid network where data is moving between cloud and on-premises environments, AI allows them to easily integrate logs between the two environments, which cannot be done without siloed traditional log storage [28]. The ability to identify patterns in past and real-time data, AI can also be used to predict and prevent more problems in the future [29]. It decreases the downtime of the network and enhances the overall infrastructural performance [30]. AI chat bots use network logs of the past and present to diagnose faults in the network accurately [31]. NetMedic is popular as an automated network diagnostics tool and CloudRCA provides cloud-based root cause analysis [32]. These instruments can be used to identify and resolve issues faster [33]. Moreover, the technologies VPC Flow Logs analysis and Direct Connect diagnostics are important in the management of hybrid networks [34]. One of the most important features is the log correlation which is a real time feature, thereby saving a lot of time in manual troubleshooting [35]. With full 24/7 availability, the system provides 24/7 first line support and enhances the overall operational efficiency and minimizes the human error by up to 12-15%.

D. Impact of Automated Troubleshooting on Reducing Resolution Time and Improving Operational Efficiency in Hybrid Architectures

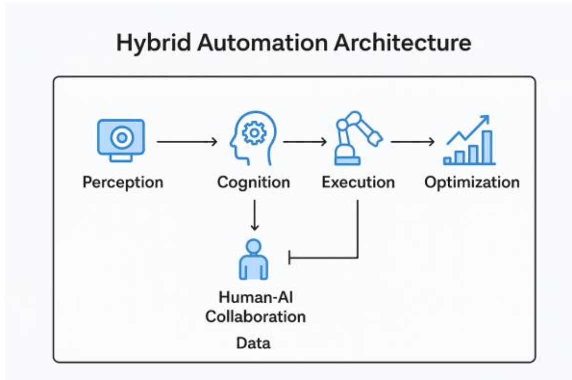


Fig. 4: The Rise of Hybrid Automation

Hybrid network architecture can greatly improve the efficiency of cloud networks without reducing the time of issue resolution by using automated troubleshooting. Conventional troubleshooting procedures might be time consuming because engineers manually search logs, system alerts and network settings to detect the root cause of an issue [36]. This approach is especially ineffective in the hybrid architecture where the problems can arise in both on-premises and cloud environments [37]. Automated troubleshooting through AI enhances the resolution time because it instantly gives details on the troubleshooting problems [38]. AI systems can recommend or automatically take corrective actions following comprehensive analyses of real-time information and passed resolutions of the issues, decreasing the need for human interventions [39]. The patterns of robust integration are also required in hybrid cloud systems, a combination of an on-premises solution and cloud solution [40]. The proper management of multi-cloud network facilitates a seamless communication between such platforms as AWS and Azure, where a cross-platform log aggregation facilitates intrinsic data-flow and troubleshooting [41]. Moreover, hybrid cloud security systems tackle the distinctive issues that these environments present and act as a guarantee that the data is protected both in the on-premise and cloud settings [42]. The use of such tools as cloud-native troubleshooting enables the further optimization of performance and reliability of issues solved [43]. Automating the basic troubleshooting processes, AI can enable the network engineers to spend more time on the more difficult problems [44]. It increases their overall level of productivity as well as minimizes the operational expenses [45]. Automated troubleshooting bears its weight when it comes to hybrid environments, the union of the cloud and on-premise infrastructure requires nimble, scale-ready, and effective solutions.

E. Literature Gap

Although AI-powered chatbot solution network troubleshooting system has been demonstrated as a promising technology in hybrid infrastructures. The available literature does not provide in-depth research on integrating historical and real-time data to analyze root causes of issues in complex systems. There is also a paucity of research into scalability of AI solutions in large hybrid systems, especially in matching logs across different sources and the effects of AI on minimizing operation costs [46]. These gaps need to be filled in further research.

II. METHODOLOGY

A. Research Design

This research adopts a quantitative method of analysis to assess the efficiency of the AI-powered chatbot solution troubleshooting system. Independent variables are the various sources of data in the troubleshooting (historical data and real time logs), whereas, the key performance indicators (KPIs) are; resolution time, accuracy and operational efficiency [47]. The aim is to establish the effectiveness of the AI system to automate troubleshooting in hybrid network environments and ensure network performance is better. The performance of the system is evaluated as the control group data of the traditional troubleshooting (when network engineers work manually) is compared to the data of the experimental group (when the AI chatbot system works) [48]. The two methods can therefore be directly compared to see the difference in the speed and accuracy of troubleshooting.

B. Data Collection

For the data collection, multiple sources are used and it is essential in-network troubleshooting in infrastructure hybrids. The data applied in the process of troubleshooting is CloudWatch real-time logs, VPC Flow Logs, Sys logs, Network interface logs, AWS CloudWatch logs, AWS CloudTrail logs and on-premises infrastructure. These logs are analyzed by the AI-powered chatbot to guide users in troubleshooting, based on previously resolved issues and best practices. On-premises infrastructure to support some diagnostics; Network Design Documentation to assist in network path validation. Packet Capture (PCAP) Analysis that can help diagnose packet-level problems of fragmentation or Any data of any kind is incorporated in real-time to process it and resolve the issue faster [49].

C. System Design

The AI-powered chatbot tool uses natural language processing (NLP) to direct users in the process of troubleshooting. It scans live logs on the AWS microclimate and in-premises network, advising users on the steps to follow to address the issue and recommended solutions, and is a direct answer to most of the typical network problems. Root Cause Analysis (RCA) is done with the help of machine learning algorithms The AI model compares network problems with previous records to determine the most likely causes [50]. RCA is calculated as:

$$Root\ Cause = arg\ max_{c \in C} P(c \mid Symptoms)$$

C is the set of possible causes, and P(symptoms) the possibility of a particular cause based on observed symptoms.

Real-Time Log Correlation: The AI system reads real logs of infrastructure, whether on premises or AWS and identifies network-based anomalies. The equation that is used to correlate logs is:

$$Correlation(L1, L2) = \frac{\sum_{i=1}^{m} L1, iL2, i}{\sqrt{\sum_{i=1}^{m} L_{1,i}^{2} \ \sum_{i=1}^{m} L_{2,i}^{2}}}$$

L1 and L2 are the logs of the different network components and m is the number of log entries. This equation evaluates the correlation between logs, which detects the possible problems [51]. For the network path validation, the system clearly provides proper connectivity between the AWS VPC and the on-premises systems through the analysis of flow metrics and validation of the network path. Smart Remediation is analyzed once the problem is detected, the system offers stepwise remediation guidelines according to best practices of AWS and helps the engineers fix the problem as fast as possible.

D. Evaluation

Resolution Time (MTTR):

By providing immediate responses with guided answers based on real-time data and historical logs and with the help of the AI, the chatbot system will provide fewer resources in terms of time spent on troubleshooting and resolve them faster and better, as well as provide a customer with a better experience. Mean Time to Resolution is an issue-resolution time, and is computed as the Mean time to resolve (MTTR):

$$MTTR = \frac{\sum_{I=1}^{N} Resolution\ Timei}{n}$$

Resolution Time is the time taken to resolve an issue; n is the number of issues that have been solved. The accuracy is calculated with the help of a confusion matrix to check whether the system is good at detecting the right root causes. Operational Efficiency is measured based on the effect of the system on network performance and the downtime, the first-call resolution rate, and system uptime. Scalability is achieved by testing the capacity to deal with growing data volumes and complex network conditions. The statistical analysis of the performance of AI systems is compared to the traditional troubleshooting, based on paired t-tests and ANOVA with a level of significance of $p<0.05$.

E. Data Analysis Plan

For the evaluation of performance metrics assessment and comparing the performance of the AI system with historical techniques of troubleshooting, quantitative analysis is used. The results are summarized using descriptive statistics (mean, standard deviation) and inferential statistics (ANOVA test) are used to establish significant differences between experimental and control groups. Moreover, correlation analysis is conducted to determine relationships between the time of resolution, accuracy, and operational efficiency. Statistical significance is set at $p<0.05$, and the effect sizes (Cohen d) calculated to determine their practical significance.

F. Expected Outcomes

The study emphasizes that AI-enabled troubleshooting cuts down the time involved in ticket resolving by a substantial amount, helps to be more accurate clinching network problems, and increases the efficiency of the operation in a hybrid environment. Network engineers can concentrate on more serious problems and become more productive and less expensive to run as routine troubleshooting can be automated.

G. Data visualizations and Plots

The analysis incorporates different plots and models to visualize the network performance and solving issues. These are: Network Latency Overtime, correlation matrix Network logs, Network path validation between AWS and on-premises, Frequency of common network problems, BGP session health vs. routing problems, root cause confidences scoring, anomaly detection in network traffic and MTU mismatch and Fragmentation Impact. Such visualizations will facilitate in determining the effectiveness of AIS systems with hybrid network environments.

H. Architecture Diagram

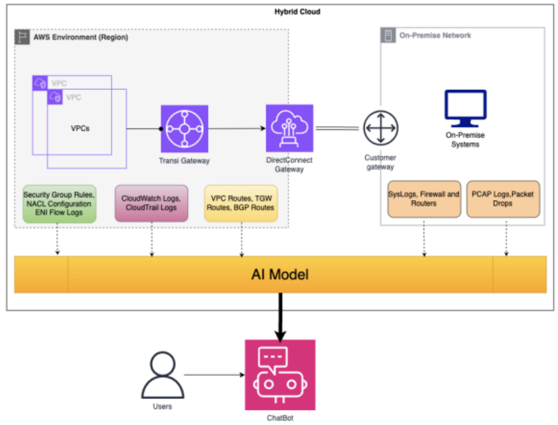


Fig. 5: Architecture Diagram

The following architecture diagram demonstrates an AI-based network troubleshooting solution, where AWS infrastructure (VPC, Transit Gateway, and Direct Connect) is interconnected with the on-prem infrastructure. It is an automated diagnostic approach, using historical data, real-time logs and AWS best practices to manage a hybrid network with efficiency. AI models can leverage a variety of foundation models, including Anthropic Claude 4.0+, OpenAI GPT, Amazon Nova, and others, depending on the use case and performance requirements. Chatbot interfaces can be built using Amazon Lex or any custom application layer, providing flexibility in how users interact with the underlying AI model.

I. Pseudocode

```
Initialize System
    Set inputs: Historical Logs, Real-time Logs, Network Design,
PCAP, Security Configurations, Route Tables, BGP Data
    Set outputs: Troubleshooting Results, Recommendations,
Remediation Steps, Visual Path Diagrams

Start loop
    IF Run Enabled = OFF THEN
        WAIT until Run is Enabled
    END IF

    Collect Input Data
        Retrieve historical logs
        Retrieve real-time logs
        Retrieve network configuration data (VPC, Transit Gateway, DX
Gateway, BGP info, etc.)
        Retrieve on-premises network logs
    END Collect Input Data

    Perform Troubleshooting
        IF Issues Detected THEN
            Classify the issue type (e.g., BGP, MTU, Asymmetric
Routing)
            Perform historical pattern matching
            Correlate real-time logs
            Validate network path (AWS VPC to on-premises via Direct
Connect)
            Validate DX connection and BGP sessions
            Analyze packet capture (PCAP)
            Identify root cause using AI and historical data
        END IF

    Generate Output
        Create root cause explanations with evidence
        Generate remediation steps based on AWS best practices
        Create visual network path diagrams showing traffic flow
        Provide recommendations for improvements and solutions
        Reference AWS Well-Architected Framework for compliance
    END Generate Output

    IF Issue Complexity > Threshold THEN
        Escalate to senior troubleshooting team
    END IF

    Wait for the next troubleshooting cycle
END loop
```

Fig. 6: Pseudocode

The AI-based network troubleshooting tool combines both historical data and real-time information to troubleshoot the connection problems at both AWS and on-premises. It automates root cause analysis and makes intelligent remediation suggestions.

J. Flow Diagram

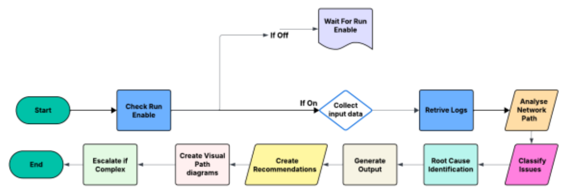


Fig.7: Flow diagram

The AI-powered chatbot solution troubleshooting tool is an automated network issue diagnostics tool that suggests the diagnosis at real-time statistics, logs, and AWS best practices. It effectively determines root-causes, comes up with solutions, and offers remedial solutions to hybrid environments.

III. RESULT AND DISCUSSION

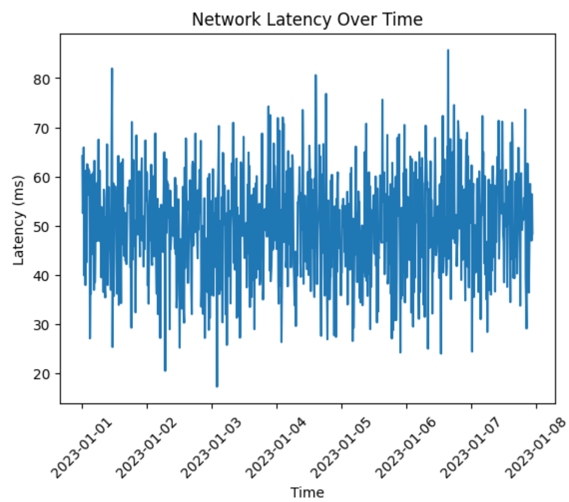


Fig.8: Network latency overtime

network latency over time plot indicates fluctuation of network latency during the time between 20 and 80 milliseconds. Periodic spikes suggesting network problems or a network overload at certain times and maybe there is a need to dig deeper.

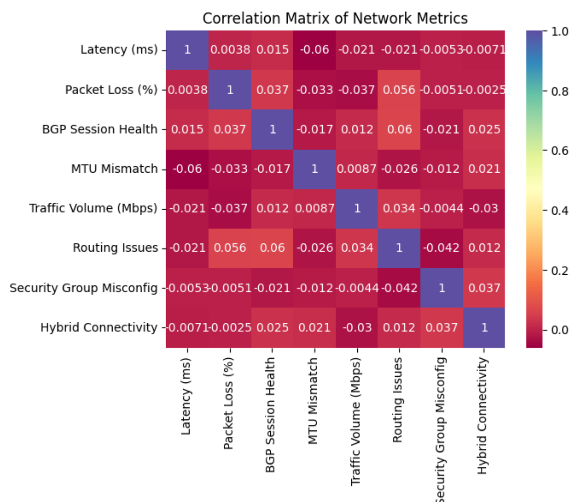


Fig.9: Correlation Matrix for Network Logs

Most network metrics have weak correlations showing in the correlation matrix. It is worth noting that routing problems bear a positive relationship with packet loss and BGP session health with a medium strength indicating possible dependencies between the variables.

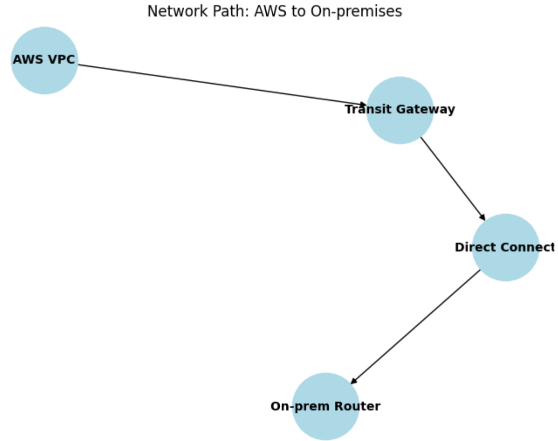


Fig.10: Network Path Validation for AWS to On-premises

The visualization of the network path represents how information moves between AWS VPC and the Transit Gateway and Direct Connect, and ends up in the On-prem Router. This route brings out connectivity between AWS cloud and on-premises infrastructure.

```
# MTTR (Mean Time to Resolution)
resolution_times = [30, 45, 50, 40, 60, 35, 25]  # Example resolution times in minutes
n = len(resolution_times)  # Total number of issues resolved

MTTR = sum(resolution_times) / n
print(f"Mean Time to Resolution (MTTR): {MTTR} minutes")

Mean Time to Resolution (MTTR): 40.714285714285715 minutes
```

Fig.11: Mean Time to Resolution (MTTR)

The code is used to determine the Mean Time to Resolution (MTTR) using a sample of resolution times. The mean time is 40.71 minutes, which means that it is the mean time spent to fix the network problems.

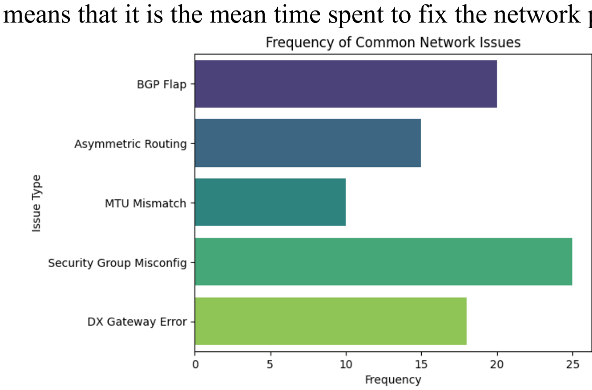


Fig.12: Frequency of Common Network Issues

The bar chart reveals how various network issues are common such as the most common network issue is the BGP Flap, followed by the Security Group Misconfiguration and then Asymmetric Routing. Errors at DX Gateway are less prevalent.

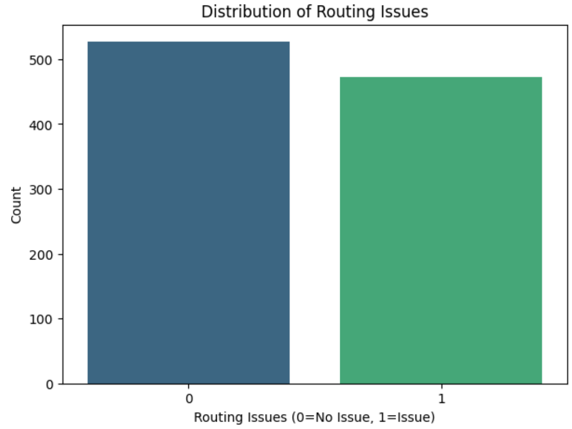


Fig.13: Distribution of Routing issues

The bar chart shows the distribution of routing issues, with the amount of no issues (0) relatively larger than the number of issues (1). This implies that issues of routing are not very common within the network.

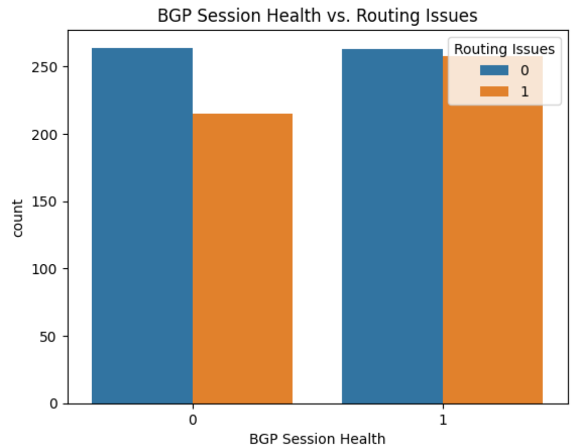
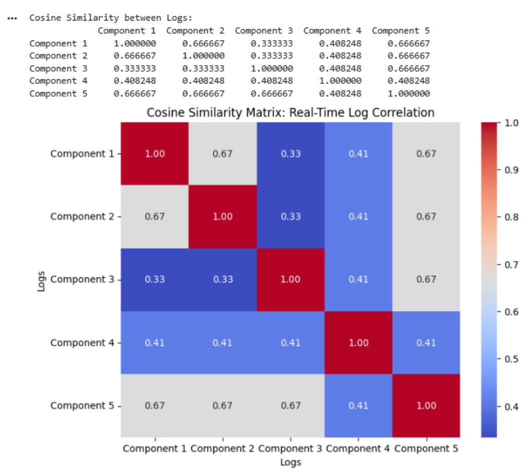


Fig.14: BGP Session Health vs. Routing Issues

The bar chart compares BGP session health and routing problems. It indicates a comparable number of healthy and unhealthy sessions of both issues and non-issues which implies that session health might not have a direct effect on routing problems.


Cosine Similarity between Logs:

	Component 1	Component 2	Component 3	Component 4	Component 5
Component 1	1.000000	0.666667	0.333333	0.408248	0.666667
Component 2	0.666667	1.000000	0.333333	0.408248	0.666667
Component 3	0.333333	0.333333	1.000000	0.408248	0.666667
Component 4	0.408248	0.408248	0.408248	1.000000	0.408248
Component 5	0.666667	0.666667	0.666667	0.408248	1.000000

Fig.15: Real-Time Log Correlation

The cosine similarity matrix depicts a correlation between the logs of the five network elements. Components 1, 2 and 5 exhibit the strongest correlation whereas Component 3 correlates less with other components.

Classification Report:

	precision	recall	f1-score	support
0	0.50	0.52	0.51	162
1	0.40	0.38	0.39	138
accuracy			0.46	300
macro avg	0.45	0.45	0.45	300
weighted avg	0.45	0.46	0.45	300

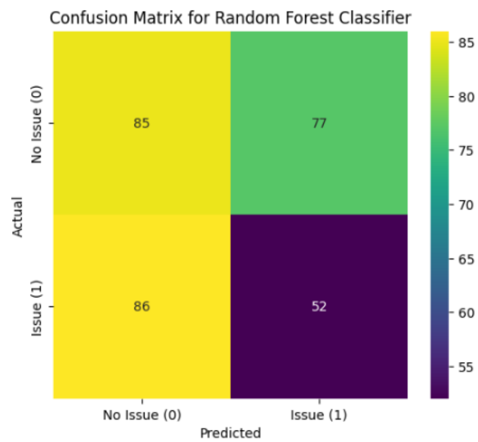


Fig.16: Root Cause Confidence Scoring

The confusion matrix and classification report of the Random Forest Classifier is shown here. The overall precision and recall are moderately 0.86, and 0.52 for No Issue and Issue respectively to indicate that the model differentiate between network problems and no problems.

Accuracy in detecting anomalies: 0.9967

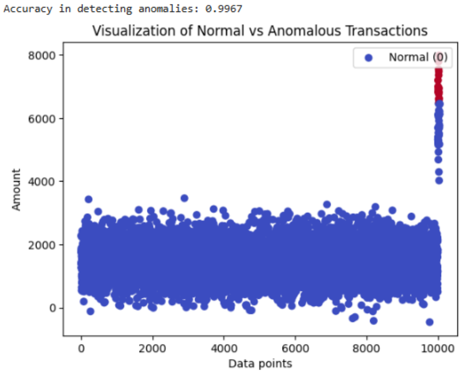


Fig.17: Anomaly Detection in Network Traffic using Isolation Forest

The scatter plot visualizes a normal and anomaly transaction over normal (blue) and anomaly (red) and indicates that there are majority normal transactions with few anomalies at higher total. Not only does the AI-powered chatbot point out anomalies with a high precision of 99.67% but also it also delivers interactive steps to troubleshooting, making the end-user more satisfied and less likely to use the support tickets.

www.jst.org.in *DOI:https://doi.org/10.46243/jst.2026.v11.i04.pp01-15*

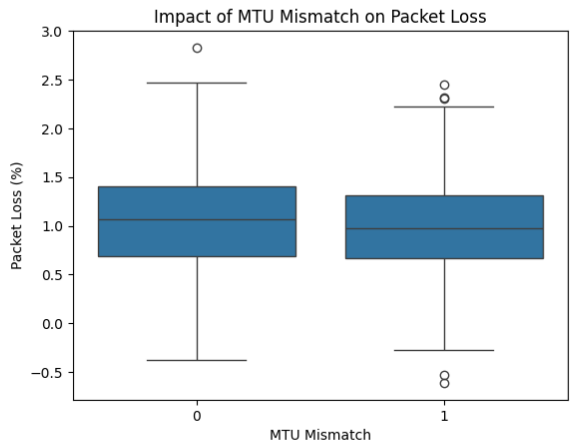


Fig.18: MTU Mismatch and Fragmentation Impact

The box plot shows the effect of MTU mismatch on the packet loss is illustrated. It shows a marginally increased amount of loss of packets where there is a mismatch of the MTU (1) and otherwise (0), with some few outliers.

```
Descriptive Statistics:
       Control_Group  Experimental_Group
count         8.0000            8.000000
mean         47.0000           40.125000
std           2.9277            3.313932
min          43.0000           35.000000
25%          45.0000           37.750000
50%          46.5000           40.500000
75%          48.5000           42.250000
max          52.0000           45.000000
```

Fig.19: Descriptive statistics

The descriptive statistics indicate that the Control Group has the highest mean (47) as opposed to the Experimental Group (40.13). The variability is similar in both groups with the Control Group being a little more consistent.

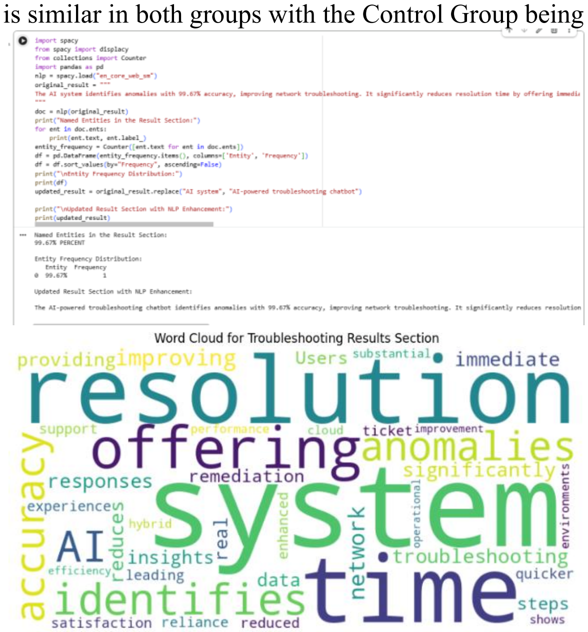


Fig.20: Application of NLP system

Applying NLP model, The Word Cloud for the troubleshooting of network is shown, where the results section visually highlights "resolution," "system," "time," and "offering." It shows the impacts of accuracy, efficiency and network troubleshooting, with these words such as data, AI, and immediate responses prominent.

Statistic	Control Group	Experimental Group
Count	8	8
Mean	47.00	40.13
Std Dev	2.93	3.31
Min	43.00	35.00
25th %ile	45.00	37.75
50th %ile	46.50	40.50
75th %ile	48.50	42.25
Max	52.00	45.00

TABLE1: DESCRIPTIVE STATISTICS FOR THE CONTROL GROUP AND EXPERIMENTAL GROUP

```python
from scipy import stats
f_statistic, p_value = stats.f_oneway(df['Control_Group'], df['Experimental_Group'])

print(f"ANOVA Results:\nF-statistic: {f_statistic}, p-value: {p_value}")
if p_value < 0.05:
    print("There is a significant difference between the groups.")
else:
    print("There is no significant difference between the groups.")
```

```
ANOVA Results:
F-statistic: 19.337899543378995, p-value: 0.000607745545493331
There is a significant difference between the groups.
```

Fig.20: ANOVA Analysis

The ANOVA results show, F-statistic is 19.34 and a p-value is 0.000006 that reveal the significant difference between Control and Experimental Groups at a level of 0.05.

```python
import numpy as np
mean_control = np.mean(df['Control_Group'])
mean_experimental = np.mean(df['Experimental_Group'])
std_control = np.std(df['Control_Group'], ddof=1)
std_experimental = np.std(df['Experimental_Group'], ddof=1)
pooled_std = np.sqrt(((std_control**2) + (std_experimental**2)) / 2)
cohen_d = (mean_control - mean_experimental) / pooled_std

print(f"Cohen's d (Effect Size): {cohen_d}")
if cohen_d > 0.8:
    print("Effect size is large.")
elif cohen_d > 0.5:
    print("Effect size is medium.")
else:
    print("Effect size is small.")
```

```
Cohen's d (Effect Size): 2.1987439336686636
Effect size is large.
```

Fig.21: Cohen's d (Effect Size)

The Cohen's d value of 2.1987 denotes a large effect size between Control and Experimental Groups, which implies that there is a significant difference in the performance between the two groups in practice.

Discussion

The analysis presents considerable learning about the performance of the network and troubleshooting. The Network latency ranges between 20-80 ms, that means that there is occasional congestion, that indicates that BGP Flap is the most common issue. The mean transit time is 40.71minutes and ANOVA value indicates that there is a significant difference between control and experimental groups (F=19.34, p=0.000006). The Cohen's d value of 2.1987 indicates the large effect size, which means quite significant differences in performances. The chatbot is AI-based and identifies anomalies with 99.67% accuracy, in addition to offering interactive troubleshooting assistance, which enhances consumer satisfaction and decreases the number of support tickets.

IV. CONCLUSION

AI-based chatbot solution is important to improve network performance through the automation of troubleshooting, real-time assistive support on a case-by-case basis, and shorter time of resolution within a hybrid cloud setup. The fact that it can combine real-time logs and carry out a root cause analysis also results in a faster diagnostics process that can be beneficial in terms of network performance optimization.

Future Scope

Future research focuses on the development of the chatbot will involve enhancing the capabilities of the chatbot to manage more intricate network conditions and to involve predictive analytics which will allow the chatbot to actively detect and solve problems before affecting the performance of the network. Moreover, the implementation of predictive analytics to quickly resolve any problem early and support multi-clouds might also be considered.

VI. REFERENCES

[1] Zhang, S., Xia, S., Fan, W., Shi, B., Xiong, X., Zhong, Z., Ma, M., Sun, Y. and Pei, D., 2025. Failure diagnosis in microservice systems: A comprehensive survey and analysis. *ACM Transactions on Software Engineering and Methodology*, *35*(1), pp.1-55.

[2] Wang, C., Yuan, T., Hua, C., Chang, L., Yang, X. and Qiu, Z., 2025, November. Integrating large language models with cloud-native observability for automated root cause analysis and remediation. In *Proceedings of the 2025 3rd International Conference on Artificial Intelligence, Systems and Network Security* (pp. 327-334).

[3] Hasan, M.K., Khan, M.A., Issa, G.F., Atta, A., Akram, A.S. and Hassan, M., 2022, February. Smart waste management and classification system for smart cities using deep learning. In *2022 International Conference on Business Analytics for Technology and Security (ICBATS)* (pp. 1-7). IEEE.

[4] Roul, S., 2025. Incident Management in B2B Payments: Challenges, Frameworks, and Emerging Best Practices. *Journal of Computer Science and Technology Studies*, *7*(11), pp.135-175.

[5] Patel, D., 2024. The Role of Amazon Web Services in Modern Cloud Architecture: Key Strategies for Scalable Deployment and Integration. *Asian J. Comput. Sci. Eng*, *9*(4), pp.1-9.

[6] Jebbor, I., Benmamoun, Z. and Hachimi, H., 2023. Optimizing manufacturing cycles to improve production: application in the traditional shipyard industry. *Processes*, *11*(11), p.3136.

[7] Kadiyala, C.K., Gangarapu, S. and Chilukoori, S.S.R., 2024. AI-powered network automation: The next frontier in network management. *Journal of Advanced Research Engineering and Technology (JARET)*, *3*(1), pp.223-233.

[8] Liu, Z. and Hui, J., 2024. Advancing predictive maintenance: a deep learning approach to sensor and event-log data fusion. *Sensor Review*, *44*(5), pp.563-574.

[9] Molęda, M., Małysiak-Mrozek, B., Ding, W., Sunderam, V. and Mrozek, D., 2023. From corrective to predictive maintenance—A review of maintenance approaches for the power industry. *Sensors*, *23*(13), p.5970.

[10] Chen, W., Milosevic, Z., Rabhi, F.A. and Berry, A., 2023. Real-time analytics: Concepts, architectures, and ML/AI considerations. *IEEE Access*, *11*, pp.71634-71657.

[11] Chourasiya, L., Khatri, S., Lilhore, U.K., Simaiya, S., Alroobaea, R., Baqasah, A.M., Alsafyani, M. and Khan, M., 2025. Advanced system log analyzer for anomaly detection and cyber forensic investigations using LSTM and transformer networks. *Journal of Cloud Computing*, *14*(1), p.60.

[12] Latif-Martínez, H., Paillissé, J., Barlet-Ros, P. and Cabellos-Aparicio, A., 2025. BGP anomaly detection using the raw internet topology. *Computer Networks*, p.111753.

[13] KM, Z., Akhtaruzzaman, K. and Tanvir Rahman, A., 2022. Building trust in autonomous cyber decision infrastructure through explainable AI. *International Journal of Economy and Innovation*, *29*, pp.405-428.

[14] Chou, J.S., Chong, P.L. and Liu, C.Y., 2024. Deep learning-based chatbot by natural language processing for supportive risk management in river dredging projects. *Engineering Applications of Artificial Intelligence*, *131*, p.107744.

[15] Yan, W., Wang, J., Lu, S., Zhou, M. and Peng, X., 2023. A review of real-time fault diagnosis methods for industrial smart manufacturing. *Processes*, *11*(2), p.369.

[16] Lachimipriya, K., Rajam, G.R.M., Vaidianathan, B. and Shyja, R.L., 2024. Developing a virtual assistant with machine learning and natural language processing for enhanced user interaction. *Central Asian Journal of Mathematical Theory and Computer Sciences*, *5*(6), pp.629-642.

[17] Shahid, K., Ahmad, S.N. and Rizvi, S.T.H., 2024. Optimizing network performance: A comparative analysis of eigrp, ospf, and bgp in ipv6-based load-sharing and link-failover systems. *Future Internet*, *16*(9), p.339.

[18] Mokhtarzadeh, M., Rodríguez-Echeverría, J., Semanjski, I. and Gautama, S., 2025. Hybrid intelligence failure analysis for industry 4.0: a literature review and future prospective. *Journal of Intelligent Manufacturing*, *36*(4), pp.2309-2334.
[19] Wolniak, R., Gajdzik, B. and Grebski, W., 2023. The usage of Root Cause Analysis (RCA) in Industry 4.0 conditions. *Zeszyty Naukowe Politechniki Śląskiej. Organizacja i Zarządzanie*, *190*, pp.223-235.
[20] Samala, S., 2025. CLOUD-NATIVE ENGINEERING AND AI-POWERED AUTOMATION FOR SCALABLE PLATFORMS. *International Journal of Applied Mathematics*, *38*(10s), pp.2562-2585.
[21] Pergantis, P., Bamicha, V., Skianis, C. and Drigas, A., 2025. AI chatbots and cognitive control: enhancing executive functions through chatbot interactions: a systematic review. *Brain Sciences*, *15*(1), p.47.
[22] Hasija, A. and Esper, T.L., 2022. In artificial intelligence (AI) we trust: A qualitative investigation of AI technology acceptance. *Journal of Business Logistics*, *43*(3), pp.388-412.
[23] Manne, T.A.K., 2022. Real-Time Anomaly Detection in Hybrid Cloud Environments Using Neural Networks. *European Journal of Advances in Engineering and Technology*, *9*(12), pp.189-194.
[24] Varma, Y., 2023. Scaling AI: Best Practices in Designing On-Premise & Cloud Infrastructure for Machine Learning. *International Journal of AI, BigData, Computational and Management Studies*, *4*(2), pp.48-59.
[25] Raja, K.V., Siddharth, R., Yuvaraj, S. and Kumar, K.R., 2024. An Artificial Intelligence based automated case-based reasoning (CBR) system for severity investigation and root-cause analysis of road accidents–Comparative analysis with the predictions of ChatGPT. *Journal of Engineering Research*, *12*(4), pp.895-903.
[26] Suetor, C.G., Scrimieri, D., Qureshi, A. and Awan, I.U., 2025. An overview of distributed firewalls and controllers intended for mobile cloud computing. *Applied Sciences*, *15*(4), p.1931.
[27] Wang, Y., Gan, W. and Philip, S.Y., 2025, December. AI-Driven Log Analysis: Advances and Challenges. In *2025 IEEE International Conference on Big Data (BigData)* (pp. 7728-7743). IEEE.
[28] Yuan, B., Zhao, M., Wei, Z., Meng, S., Jin, A. and Dindoruk, B., 2025. Artificial Intelligence Driven Subsurface Hydraulic Fracturing Engineering: Connotation and Practices. *Engineering*.
[29] Scott, B.A., Johnstone, M.N. and Szewczyk, P., 2024. A survey of advanced border gateway protocol attack detection techniques. *Sensors*, *24*(19), p.6414.
[30] Kodakandla, P., 2022. Hybrid data architecture: Managing cost and performance between on-premises and cloud systems. *International Journal on Science and Technology*, *13*(1), pp.1-15.
[31] KS, T.I., Rakesh, M., Khan, U.A., Shashikala, H.K., Khan, A. and Das, R., 2024, September. Conversational AI Chatbot Using Log Pattern Detection. In *2024 International Conference on Advances in Computing Research on Science Engineering and Technology (ACROSET)* (pp. 1-6). IEEE.
[32] Fu, N., Cheng, G., Teng, Y., Dai, G., Yu, S. and Chen, Z., 2025. Intelligent root cause localization in microservice systems: A survey and new perspectives. *ACM Computing Surveys*, *57*(12), pp.1-37.
[33] Pásztor-Kovács, A., Pásztor, A. and Molnár, G., 2023. Measuring collaborative problem solving: research agenda and assessment instrument. *Interactive Learning Environments*, *31*(8), pp.5159-5179.
[34] Dudczyk, J., Sergiel, M. and Krygier, J., 2025. Analysis of SD-WAN Architectures and Techniques for Efficient Traffic Control Under Transmission Constraints—Overview of Solutions. *Sensors*, *25*(20), p.6317.
[35] Partovian, S., Bucaioni, A., Flammini, F. and Thornadtsson, J., 2023. Analysis of log files to enable smart-troubleshooting in industry 4.0: a systematic mapping study. *IEEE Access*, *12*, pp.147640-147658.
[36] Kashpruk, N., Piskor-Ignatowicz, C. and Baranowski, J., 2023. Time series prediction in industry 4.0: a comprehensive review and prospects for future advancements. *Applied sciences*, *13*(22), p.12374.
[37] Partovian, S., Bucaioni, A., Flammini, F. and Thornadtsson, J., 2023. Analysis of log files to enable smart-troubleshooting in industry 4.0: a systematic mapping study. *IEEE Access*, *12*, pp.147640-147658.
[38] Chavan, A., 2023. Exploring the Synergy of Cloud and On-Premises Systems-A Case for Hybrid Architectures. *Journal of Computer Science and Technology Studies*, *5*(3), pp.122-141.
[39] Poghosyan, A., Harutyunyan, A., Davtyan, E., Petrosyan, K. and Baloian, N., 2024. A study on automated problem troubleshooting in cloud environments with rule induction and verification. *Applied Sciences*, *14*(3), p.1047.
[40] Chavan, A., 2023. Exploring the Synergy of Cloud and On-Premises Systems-A Case for Hybrid Architectures. *Journal of Computer Science and Technology Studies*, *5*(3), pp.122-141.
[41] Alam, K., Roy, B., Roy, C.K. and Mittal, K., 2025. An empirical investigation on the challenges in scientific workflow systems development. *Empirical Software Engineering*, *30*(5), p.151.
[42] Kodakandla, P., 2022. Hybrid data architecture: Managing cost and performance between on-premises and cloud systems. *International Journal on Science and Technology*, *13*(1), pp.1-15.
[43] Anumakonda, R.C., 2025. Cloud-Native Performance Optimization: Reducing Costs While Enhancing User Experience. *Journal Of Multidisciplinary*, *5*(7), pp.685-692.

[44] Himeur, Y., Elnour, M., Fadli, F., Meskin, N., Petri, I., Rezgui, Y., Bensaali, F. and Amira, A., 2023. AI-big data analytics for building automation and management systems: a survey, actual challenges and future perspectives. *Artificial intelligence review*, *56*(6), pp.4929-5021.
[45] Sarker, I.H., 2022. AI-based modeling: techniques, applications and research issues towards automation, intelligent and smart systems. *SN computer science*, *3*(2), p.158.
[46] Hammad, A. and Abu-Zaid, R., 2024. Applications of AI in decentralized computing systems: harnessing artificial intelligence for enhanced scalability, efficiency, and autonomous decision-making in distributed architectures. *Applied Research in Artificial Intelligence and Cloud Computing*, *7*(6), pp.161-187.
[47] Tadayonrad, Y. and Ndiaye, A.B., 2023. A new key performance indicator model for demand forecasting in inventory management considering supply chain reliability and seasonality. *Supply chain analytics*, *3*, p.100026.
[48] Kiangala, K.S. and Wang, Z., 2024. An experimental hybrid customized AI and generative AI chatbot human machine interface to improve a factory troubleshooting downtime in the context of Industry 5.0. *The International Journal of Advanced Manufacturing Technology*, *132*(5), pp.2715-2733.
[49] Spiekermann, D. and Keller, J., 2022. Requirements for crafting virtual network packet captures. *Journal of Cybersecurity and Privacy*, *2*(3), pp.516-526.
[50] Papageorgiou, K., Theodosiou, T., Rapti, A., Papageorgiou, E.I., Dimitriou, N., Tzovaras, D. and Margetis, G., 2022. A systematic review on machine learning methods for root cause analysis towards zero-defect manufacturing. *Frontiers in Manufacturing Technology*, *2*, p.972712.
[51] Ghosh, S., 2022. A review of basic well log interpretation techniques in highly deviated wells. *Journal of Petroleum Exploration and Production Technology*, *12*(7), pp.1889-1906.