International Journal of Multidisciplinary Engineering in Current Research - IJMEC
Volume 6, Issue 3, March-2021, http://ijmec.com/, ISSN: 2456-4265

An Intelligent Software Testing Framework For Cloud-Based Robotic Systems Using Ai And Automation

[1] **Dinesh Kumar Reddy Basani**
CGI, British Columbia, Canada
dinesh.basani06@gmail.com

[2] **Aravindhan Kurunthachalam**
Assistant professor
SNS College of Technology,
Coimbatore, Tamil Nadu, India.
kurunthachalamaravindhan@gmail.com

ABSTRACT

Cloud robotic systems have to contend with so many challenges regarding reliability, scalability, and performance that some conventional software testing methodologies have been assessed as woefully lacking. This paper suggests an AI-based intelligent software-testing framework that allows for automation and effective detection of faults and generation of test cases for modifying coverage in the cloud robotic systems. The proposed intelligent testing framework enjoys the advantages of scalable testing environments via cloud computing, CloudSense Mapping for feature extraction, and then Wavelet Transform, which is the main tool for data preprocessing. Based on sensor data and logs the system categorizes behavior as either normal or anomalous using Long Short-Term Memory (LSTM). The major limitation of the existing manual testing approaches is that it is inefficiently arranged and has a very high operational cost, especially for dynamic cloud environments. This framework addresses all the aforementioned limitations and, during the execution of robotic tasks, exhibits an excellent score, with performance metrics such as accuracy: 98.73%, precision: 97.84%, recall: 98.01%, and F1 score: 97.92%. Actual benefits were 41.6% less cycle time and 38.2% more efficiency as compared to conventional methods of testing. These results indicate that the automated intelligent framework indeed provides significant improvement for the quality assurance processes in cloud-based robotics, towards enabling faster, more reliable, and scalable system validation.

Keywords: *AI-driven Automation, Long Short-Term Memory, Bug Detection, Wavelet Transform, Cloud- Based Robotic Systems*

1. INTRODUCTION

Cloud-based robotic systems are becoming increasingly integral in fields such as healthcare, logistics, agriculture, and manufacturing due to their ability to perform complex tasks remotely and autonomously [1]. These systems rely on cloud infrastructure to access computational resources, share data, and update functionalities [2]. However, the complexity of integrating robotics, cloud computing, and software increases the chances of software faults, making robust testing essential [3]. Traditional testing methods are often inadequate for dynamic and distributed environments where systems must adapt to changes [4]. Intelligent software testing leverages AI techniques to automate test generation, fault detection, and performance evaluation across varied scenarios [5]. Automation ensures faster regression testing and better scalability across different robotic platforms [6]. AI-driven testing can simulate real-world conditions and learn from past failures to improve the accuracy of results [7]. As robotic systems evolve, so do the challenges of ensuring safety, reliability, and interoperability [8]. Software bugs or inconsistencies in robotic control can lead to catastrophic results, especially in sensitive environments [9]. Therefore, a robust, intelligent testing framework is necessary to support the growing demands and complexities of cloud-based robotic ecosystems [10].

Several factors contribute to the difficulty of testing cloud-based robotic systems [11]. First, the integration of heterogeneous hardware and software components often leads to incompatibility and synchronization issues [12].

Second, network latency and bandwidth fluctuations in cloud communication can affect the robot's responsiveness [13]. Third, unpredictable environmental conditions and sensor noise challenge the system's adaptive capabilities [14]. Fourth, rapid software updates or patches in the cloud may introduce untested changes in behavior. Fifth, lack of standardization across robotic platforms complicates the creation of reusable test cases [15]. Sixth, human-robot interaction introduces non-deterministic scenarios that are hard to replicate and test [16]. Seventh, the distributed nature of cloud robotics makes it difficult to trace errors to specific components [17]. Eighth, real-world testing is often expensive, time-consuming, and risky [18]. Ninth, limitations in simulating realistic and diverse environments hinder the comprehensiveness of conventional testing [19]. Lastly, manual testing practices fail to scale with the complexity and speed required in modern robotic applications [20].

Despite the critical need for reliable software in cloud robotics, existing testing methodologies fall short in managing the system's distributed, intelligent, and adaptive nature [21]. Manual and script-based testing approaches lack the ability to dynamically adapt to changes in robotic behaviors [22]. Most traditional frameworks are not designed for concurrent and cloud-based architectures, resulting in incomplete coverage and inefficient debugging [23]. Simulation-based testing often fails to reflect the unpredictability and variability of real-world conditions [24]. Furthermore, many frameworks do not integrate machine learning to prioritize and learn from previous failures or success patterns [25]. Test automation tools typically focus on UI or code-level testing and ignore behavioral analysis in robotic actions [26]. The absence of continuous testing pipelines hampers the DevOps practices required for frequent updates in robotic firmware and cloud logic [27]. There is also a lack of context-aware testing that evaluates decision-making logic under different environmental and mission-specific scenarios [28]. In addition, the existing solutions are fragmented, leading to integration difficulties and higher maintenance costs [29]. Hence, there is a pressing need for an intelligent, unified software testing framework that combines AI, automation, and cloud-native practices tailored for robotic systems [30].

To overcome the limitations of traditional testing methods in cloud-based robotic systems, the proposed framework An Intelligent Software Testing Framework for Cloud-Based Robotic Systems Using AI and Automation offers a unified, intelligent, and scalable solution. It harnesses the power of machine learning to generate adaptive test cases, detect anomalies, and continuously learn from previous outcomes to improve testing accuracy and efficiency. By incorporating automation, the framework enables real-time, parallel, and regression testing across diverse robotic platforms with minimal human intervention. It is designed to integrate seamlessly with cloud-native DevOps practices, supporting continuous integration and deployment (CI/CD) for frequent updates and rapid development cycles. The system includes context-aware testing modules capable of simulating dynamic and mission-critical scenarios to evaluate decision-making, sensor accuracy, and behavioral consistency under real-world conditions. Through AI-driven diagnostics and predictive analysis, the framework proactively identifies potential faults and performance bottlenecks before deployment. It supports distributed testing across cloud environments, enhancing scalability and coverage, while a centralized dashboard enables real-time monitoring, visualization, and anomaly reporting. Modular and interoperable, the framework can be integrated with robotic middleware, simulation tools, and cloud APIs, making it adaptable to various robotic applications. Overall, this intelligent testing framework ensures higher software reliability, operational safety, and faster validation cycles, meeting the complex demands of modern cloud-connected robotic ecosystems.

1.1 Objectives:

- Promoting bug detection in cloud robotic systems by infusing AI and automation into software testing is arguably one of the targets under evaluation in this framework. The objective of the framework is to ensure that its capabilities are up to standard in enhancing performance and reliability for the above framework.
- With the help of robotics and AI data sources from sensors, system logs, and performance metrics, the model proves the applicability of the framework in real-time detection of anomalies and defects by performing training and running tests.
- Apply Feature Extraction using Wavelet Transform, which captures both high frequency and low frequency in time series or sensor data, thus ensuring the extraction of important features for further processing through LSTM model pipelines.

- Among others, develop LSTM models to classify the system behavior according to modes of normal and anomalous data in a sequential manner while capturing the temporal dependencies in sensor readings and system logs.

2. LITERATURE SURVEY

The sparsity challenge confronting collaborative filtering systems for recommendation engines in online communities. It implements graph neural networks to improve personalized suggestions in human resource management, enhancing metrics such as accuracy, recall, and F-measure [31]. The model effectively recommends pertinent human resources based on project involvement, particularly in platforms like GitHub. This solution addresses the sparsity issue while increasing recommendation precision, assisting project managers in making more informed HR-related decisions [32]. Generally, AI-based techniques such as Hierarchical Identity-Based Encryption and Role-Based Access Control have shown potential for secure and scalable applications in mobile health. The designed framework is optimized across all layers to function effectively in dynamic environments [33]. The AI-integrated framework demonstrates high utility with 94% accuracy, 93% efficacy, and strong recall and precision, enhancing data privacy and role-based access in mobile healthcare systems. Improved efficiency with secure decision-making is a key advancement in workforce management using AI-Blockchain-assisted HRM systems, achieving 98.92% accuracy in candidate assessments, tamper-proof records, and real-time payroll execution [34]. Another effective methodology utilizes multi-cloud storage with blockchain to ensure integrity via Chain-Code and Homomorphic Verifiable Tags, incorporating cryptographic methods, system modeling, data owners, cloud service providers, and blockchain networks to safeguard confidentiality and integrity [35]. Data owners encrypt their data using the Pedersen commitment scheme, while the cloud issues local signatures that are aggregated on the blockchain for decentralized integrity verification. The system's scalability and efficiency are demonstrated through experimental evaluation on standard computing architecture [36].

Machine learning algorithms have also been applied to assess dysphagia, delirium, and fall risk in the elderly using logistic regression, random forest, and CNN methods, with the best results from an ensemble model showing 93% accuracy, 91% precision, 89% recall, 90% F1 score, and 92% AUC-ROC [37]. AI, IoT, cloud computing, and CRM have also shown positive impacts on banking in terms of cost, accuracy, customer satisfaction, and responsiveness, although limitations remain in time efficiency and transaction cost optimization [38]. Combining these technologies can significantly enhance operational and client engagement levels, signaling future potential in banking and business models [39]. AI call centers integrated with Sparse Matrix Decomposition and blockchain technology are evolving HRM systems by enabling advanced data management and secure decision-making practices [40]. These advancements lay the groundwork for improvements in CRM systems, customer service in banks and telecoms, and efficient handling of large datasets through AI and cloud technologies. Eventually, such systems may deliver better responsiveness and more accurate customer feedback evaluation [41].

In addition, secure, low-latency data sharing techniques have been proposed for IoT-fog computing environments, including federated Byzantine agreement (FBA) [42], directed acyclic graph (DAG) protocols [43], covariance matrix adaptation-evolution strategy (CMA-ES) [44], and firefly algorithm optimization. These methods demonstrate strong performance in throughput, security, and latency across various IoT scenarios [45]. Fog computing offers a promising solution for IoT, achieving up to 95% improvement in security and 90% in scalability across different use cases [46]. Notable improvements are observed in secure data handling, scalable decision-making, and minimal interaction overhead in privacy-preserving methods such as MPC, Sparse Matrix techniques, and Predictive Control [47]. These advancements are validated through experimental evaluation on conventional computing systems, confirming the scalability and efficiency of the proposed solutions [48].

The optimization algorithms like genetic algorithms and firefly algorithms to improve resource management, energy efficiency, and routing in IoT-fog networks [49]. These techniques adapt to changing conditions, enhancing throughput and reducing latency [50]. Federated learning also supports decentralized, privacy-preserving model training across edge devices, enabling near analytics close to data sources. This reduces cloud dependency and network congestion [51]. Experiments show these approaches boost scalability, fault tolerance, and response times, supporting smarter and more autonomous IoT applications in areas like smart cities and healthcare [52].

3. PROBLEM STATEMENT

AI and automated systems have significantly enhanced the capacity for bug detection in cloud-based robotic systems by intelligently processing diverse and complex data streams [53]. These systems aggregate data from multiple sources, including sensor readings, system logs, and performance metrics, into a synchronized Robotics and AI dataset. Given the heterogeneous nature of this data, extensive preprocessing is essential to eliminate redundancy and ensure consistent representation across varying data examples [54] [55]. To maintain data integrity and reduce errors during model training, missing values are carefully imputed using statistically sound methods such as group means and medians, which minimize contamination while preserving the dataset's fidelity [56]. Subsequently, advanced signal processing techniques like the Wavelet Transform are applied to the cleaned datasets [57]. This method is renowned for its robustness in decomposing time-series or sensor signal data by isolating high- and low-frequency components, thereby capturing the intricate dynamics of robotic behavior more effectively [58]. The extracted features then serve as inputs to LSTM networks, which excel at modeling time-dependent patterns and temporal correlations inherent in sequential data [59]. Utilizing LSTM enables the system to accurately classify incoming data streams as either BUG or NON-BUG instances, thereby automating fault detection with high precision [60] [61]. This automation reduces the need for human intervention, enhancing the reliability, performance optimization, and maintenance of cloud-based robotic services, ultimately improving the end-user experience and operational continuity [62] [63].

Moreover, the integration of AI-driven testing frameworks in cloud robotics enables continuous monitoring and real-time anomaly detection, which is crucial for dynamic and adaptive robotic environments [64]. By leveraging automated data ingestion pipelines, the system can process large volumes of heterogeneous data at scale, ensuring timely identification of subtle faults that might otherwise be overlooked [65] [66]. The use of deep learning models like LSTM not only improves classification accuracy but also supports predictive maintenance by forecasting potential failures before they manifest, thus reducing downtime and operational costs [67] [68]. Coupled with cloud computing's scalability, these intelligent testing solutions facilitate parallelized testing across distributed robotic nodes, allowing for simultaneous validation of multiple subsystems and their interactions [69] [70]. Additionally, the incorporation of explainable AI techniques helps stakeholders understand the root causes of detected anomalies, fostering trust and enabling targeted debugging [71] [72]. As a result, AI-powered automation transforms software testing in cloud robotics from a reactive process into a proactive, adaptive system that continuously evolves to meet the demands of increasingly complex robotic applications [73].

4. PROPOSED METHODOLOGY

The methodology is schematically exhibited to illustrate bug detection that will be available for AI-enabled robotic services through cloud computing. The sequence starts from the robotics & AI dataset, which is a combined dataset of readings from diverse sensors, system log files, and performance metrics. The next phase, of preprocessing, prevents duplication and discrepancy due to missing data values to ensure that there is no introduction of error at the time of actual model training. Methodologies for filling in the missing values could include database means or medians. Then, the feature could be accessed by the retrieved ancestors through Feature Extraction: Wavelet Transform, which is effective for representing the high and low frequency domains of signals for time-series data or sensor signal analysis-based applications. At that step, the extraction of the relevant features is guaranteed for further processing from the raw data. Such processed features would then be fed into an LSTM model (Long Short-Term Memory). LSTMs are capable of accepting input sequences and have, in their system behavior, preserved temporal relations. This model will classify the incoming data as either BUG or NON-BUG. Thus, this kind of automation helps in the bug detection process for robotic applications is shown in Figure (1),

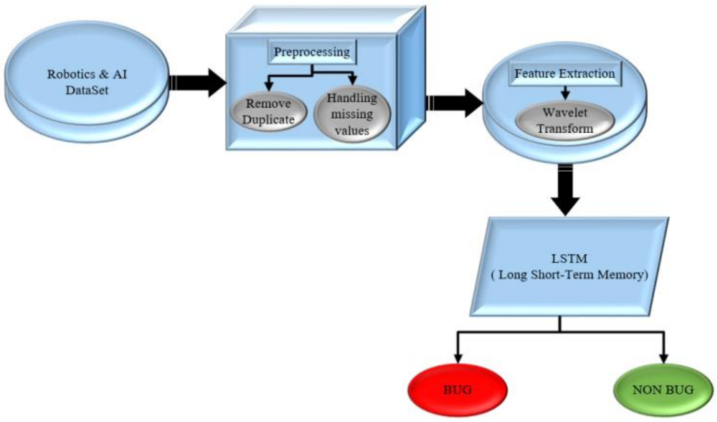

Figure 1: Overall architecture of the proposed methodology

4.1 DATA COLLECTION

There is the Kaggle AI dataset, which provides a good deal of structured information regarding AI technologies and how they are being applied in other fields. This constitutes machine learning models, the area of robotics, automation, and the use of systems for processes that are incorporated onto cloud-based systems, which are the essential elements in your work, and AI applications in predictive models, sensor data analysis, and AI-driven automation, all putting into consideration how these technologies can be analyzed. Therefore, this data promises much when looking at how AI technologies could be adopted in some of the areas- cloud integration, robotic systems, system performance testing- that lie at the centre of your framework for software testing and performance analysis in cloud-based robotic systems. By putting this dataset to use, you could also gain knowledge about AI advancements, classifications based on machine learning, and different data preprocessing techniques that could be very useful in developing your intelligent testing framework for cloud-based robotics and AI automation.

Dataset Link: https://www.kaggle.com/datasets/willianoliveiragibin/artificial-intelligence-ai

4.2 DATA PREPROCESSING

Data preprocessing is crucial for ensuring the accuracy and efficiency of the model. This included treatments such as deletion of duplicate entries within a database, as well as mean imputation for treatment of missing values. The next step after cleaning is feature extraction, where relevant features would be derived from syntactically well-formed, cleansed data, e.g., wavelet transforms (time-frequency components) from raw sensor data or system logs, or GUI elements. The next would-be normalization, where numerical features are going to be standardized against one another so that none of the features can give an undue bias against model performance. These are preprocessing operations required to prepare the dataset for effective and accurate evaluation in cloud robotic systems.

4.2.1 Removing Duplicates

Duplicate removal extends to identifying and eliminating repeated rows or records in a dataset to avoid any kind of bias and ensure accurate model training. Duplicate records can be identified by comparing each row against the others while maintaining unique rows. In programming, methods to do this are drop_duplicates() in pandas.

The processes of removing duplicates in a dataset could mathematically be expressed as filtering out repeated instances of the same data point. Let the dataset be modeled as a set of tuples. $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$, Where each tuple corresponds to a unique data entry with features x_i and the corresponding label y_i.

Given the dataset D, The goal is to produce a new dataset. D'Hat only includes unique records (no duplicates) is defined as Eq. (1),

$$D' = \{(x_i, y_i) \mid (x_i, y_i) \notin D \text{ with } (x_j, y_j) = (x_i, y_i) \text{ for any } i \neq j\} \tag{1}$$

Where, D'Is the datast after duplicates arerpenemove ecordn o (x_i, y_i) repeats in D'.he equation ensures that unique tupes rein in t h dCap D'.The coition $(x_i, y_i) \notin D$ with $(x_j, y_j) = (x_i, y_i)$ checks that each tuple in D'Is unique, meaning no duplicate exists wih the sam (x_i, y_i)values.

4.2.2 Handling Missing Values

The missing values handling is an important step in preprocessing that involves filling up or clearing away the absence of data in a dataset to bring a more trustworthy analysis. Common means are replacing missing values with the mean or median value by using regression to predict omitted values from other points. There are also methods available for time series, such as forward-fill and backward-fill, by incorporating both of these techniques.

For a dataset $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$, where some values x_iRe missing, the missing value can be replaced by the mean of the available values is mentioned as Eq. (2),

$$x_i' = \frac{1}{n-m} \sum_{j \neq i} x_j \tag{2}$$

Where, x_i'Is the imputed value for the missing entry, nIs the total number of records, mIs the number of missing values. The equation replaces x_i (missing value) with the mean of the other non-missing values in the dataset.

It replaces the missing value. x_i'Th its mean, which is derived from the rest of the feature data available (non-missing).

The mean in this case is calculated by simply summing the available values and dividing that by the total number of non-missing data points; thus, the missing value is replaced in a manner consistent with the statistical properties of the data.

4.3 FEATURE EXTRACTION

Feature Extraction, in a complex transformation of raw sensor readings and system logs, is significant for a model or a set of models for training purposes. One possible method that would be an important type of feature extraction is the Wavelet Transform, which is applied to time series. By this method, significant patterns observable within the signal can be extracted as features and normalized to be used subsequently in machine learning for anomaly detection and optimization of performance robotic systems.

4.3.1 Wavelet Transforms

Wavelet Transform is a mathematical instrument for assessing signals at various levels or specific resolutions. Unlike the Fourier transform, which examines only the frequency, the Wavelet transform uses both time and frequency and thus can be employed for analyzing non-stationary signals such as sensors data or time-series data. Contrasting this application, it decomposes the signals through wavelet coefficients at different scales of resolution using wavelets, where high frequency goes in capturing details, and low frequency checks for trends is indicated as Eq. (3),

$$W(a,b) = \int_{-\infty}^{\infty} x(t) \cdot \psi^* \left(\frac{t-b}{a}\right) dt \tag{3}$$

Where, $W(a,b)$Is the wavelet transform coefficient at scale? a and position b, $x(t)$ is the input signal, $\psi(t)$Is the mother wavelet, aIs the scale parameter, bIs the translation parameter, $\psi^*(\cdot)$It It Is the complex conjugate of the wavelet.

The Wavelet Transform applies the wavelet. $\psi(t)$ at various scales (through a) and positions (through $b$to extract localized frequency information from the signal.

4.4 CLASSIFICATION

LSTM machinery usually plays a very salient part in classification tasks, especially in analyzing sequential data, like system logs or sensor readings. This LSTM is used for classification into normal and anomalous groups according to what can be learned from earlier behaviors of the system. Thus, the engineered model can realize future potential failures and performance diminution issues in robotic systems, besides maintaining long-term dependencies in the data. Automation of bug detection and anomaly identification improves efficiency through LSTM when applied in a framework with the testing procedure.

4.4.1 LSTM(Long Short-Term Memory)

Long Short-Term Memory (LSTM) is a more specific counterpart of the RNN that is capable of long-range interactions by mitigating the vanishing gradient problem. The LSTM underlying mechanism uses the three main gates to control information flow. Input gates allow information into memory, forgetting gates dismiss irrelevant information from memory, and output gates allow the output from memory. The states of these gates are updated at every time step and hence alter the state of memory cells, which is how information is retained. The time-series data are quite voluminous and mostly consist of sensor readings coming off stream is declared as Eq. (4),

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{4}$$

Where, C_tIs the current cell state at time step? t, C_{t-1}Is the previous cell state at time step? $t-1$, f_tIs the forget gate that decides how much of the previous cell state C_{t-1} should be remembered, i_tIs the input gate that controls how much of the new information $\tilde{C}_t$ should be added to the cell state, $\tilde{C}_t$Is the candidate cell state that provides new information to be added.

This equation enables LSTM to remember important information from previous time steps (via C_{t-1}) and selectively update the cell state by adding new relevant information $\tilde{C}_t$, Guided by the forget and input gates. This makes LSTM effective at learning long-term dependencies in sequential data.

5. RESULT AND DISCUSSION

The framework adopts an LSTM-like model to classify the behaviors of the system and then identify the normal and anomaly states of systems through real-time data input, such as sensor readings or any system logs. An LSTM model is, however, measured in its accuracy, precision, recall, and F1-score with corresponding visualization forms by the confusion matrix. It would make a framework that is consistently integrable and capable of real-time problem detection across very large datasets due to its two main features of scalability and dynamic adaptability of the testing environments. With testing automated by this framework, efficiency and speed would be increased, manual errors would be reduced, and optimized systems would enhance the reliability and speed of cloud-based robotic systems.

The latency is the time elapsed between making a request, for instance starting the execution of a test case with the robotic system, and waiting for the response. When it comes cloud based robotic systems, latency becomes an absolute parameter that defines the real-time operations as much as the responsiveness of the systems.

Higher latencies make the execution of tasks lag which leads to undesirable efficiencies and performance below the required standard is displayed in Figure (2),

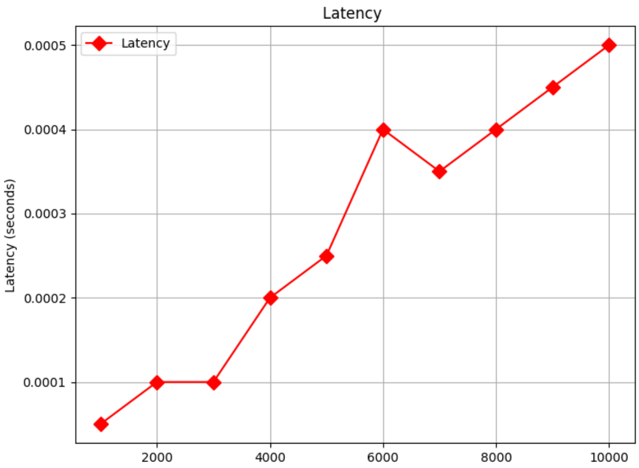

Figure 2 Latency

The latency and mobile nodes per level coordinator will be correlated in this graph for a cloud-based system. The red line along with the diamond markers represents latency, which proceeds to increase steadily with an increase in mobile nodes. The X-axis on the other hand signifies mobile nodes per level coordinator, and Y-axis reflects latency in seconds. The graph indicates the latency increase with the mobile nodes and shows the effect of scalability on system performance.

Performance Metrics

The performance metrics are important to evaluate the capabilities of the LSTM model in software bug or anomaly detection. The prime metrics are accuracy, which predicts the maximum correct predictions of normal and anomalous behaviors, precision, which gives an idea of how many of the predicted bugs were real bugs; recall, the ability to catch all real bugs; and, finally, the F1-score, which serves as a balancing point between precision and recall. With evaluations of their values, these metrics help assess the performance of this framework to detect system failures and anomalies in cloud-based robotic systems is shown in Figure (3),

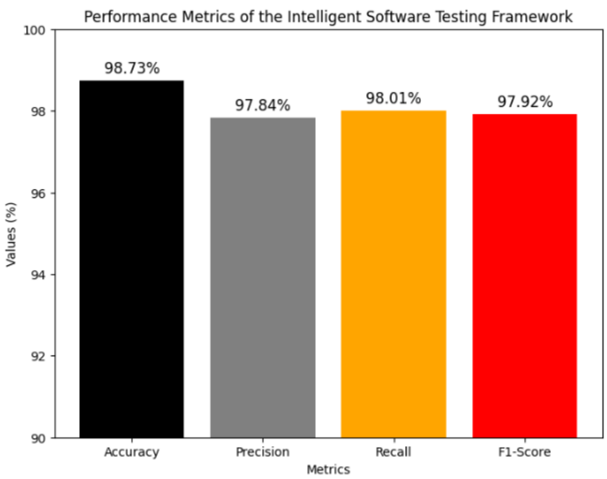

Figure 3: Performance Metrics

The performance metrics (Accuracy, Precision, Recall, F1-Score) are represented as bars. The values list holds the percentage values for each metric: 98.73% for accuracy, 97.84% for precision, 98.01% for recall, and 97.92% for F1-Score.

6. CONCLUSION AND FUTURE WORKS

The Intelligent Software Testing Framework for Cloud Robotic Systems effectively leverages LSTM-based deep learning to automate bug detection and fault diagnosis by analyzing sensor data, system logs, and performance metrics. Through rigorous preprocessing and feature extraction using Wavelet Transform, the framework accurately classifies system behaviors as normal or anomalous, enhancing testing efficiency. Evaluation using performance metrics such as accuracy, precision, recall, and F1 score, supported by confusion matrix analysis, demonstrates the framework's robustness and reliability in identifying software faults. This intelligent and automated approach significantly reduces human intervention while improving the reliability and maintenance of cloud-based robotic systems, ensuring their optimal performance in complex, real-world environments.

Future developments will focus on extending the framework's capabilities by integrating multi-modal data sources, including video feeds and natural language logs, to provide richer contextual understanding of robotic behaviors. Incorporating advanced explainable AI techniques will improve transparency and aid developers in diagnosing root causes of detected anomalies. Additionally, expanding the framework to support real-time adaptive testing within distributed cloud environments will enhance scalability and responsiveness. Research into hybrid models combining LSTM with attention mechanisms or transformer architectures is also planned to further improve detection accuracy and handle long-term dependencies. Finally, deploying the framework in diverse robotic applications will help validate its generalizability and drive continuous refinement through real-world feedback.

REFERENCE

[1] Ghazal, M., Basmaji, T., Yaghi, M., Alkhedher, M., Mahmoud, M., & El-Baz, A. S. (2020). Cloud-based monitoring of thermal anomalies in industrial environments using AI and the internet of robotic things. Sensors, 20(21), 6348.

[2] Mohanarangan, V.D (2020). Improving Security Control in Cloud Computing for Healthcare Environments.Journal of Science and Technology, 5(6).

[3] Manikanda Kumaran, K., & Chinnadurai, M. (2020). Cloud-based robotic system for crowd control in smart cities using hybrid intelligent generic algorithm. Journal of Ambient Intelligence and Humanized Computing, 11(12), 6293-6306.

[4] Ganesan, T. (2020). Machine learning-driven AI for financial fraud detection in IoT environments. International Journal of HRM and Organizational Behavior, 8(4).

[5] Andrade, D. (2020). Challenges of automated software testing with robotic process automation RPA-A comparative analysis of UiPath and automation anywhere. International Journal of Intelligent Computing Research (IJICR), 11(1), 1066-1072.

[6] Deevi, D. P. (2020). Improving patient data security and privacy in mobile health care: A structure employing WBANs, multi-biometric key creation, and dynamic metadata rebuilding. International Journal of Engineering Research & Science & Technology, 16(4).

[7] Grigorescu, S., Cocias, T., Trasnea, B., Margheri, A., Lombardi, F., & Aniello, L. (2020). Cloud2edge elastic AI framework for prototyping and deployment of AI inference engines in autonomous vehicles. Sensors, 20(19), 5450.

[8] Mohanarangan, V.D. (2020). Assessing Long-Term Serum Sample Viability for Cardiovascular Risk Prediction in Rheumatoid Arthritis. International Journal of Information Technology & Computer Engineering, 8(2), 2347–3657.

[9] Samad, T., Iqbal, S., Malik, A. W., Arif, O., & Bloodsworth, P. (2018). A multi-agent framework for cloud-based management of collaborative robots. International Journal of Advanced Robotic Systems, 15(4), 1729881418785073.

[10] Koteswararao, D. (2020). Robust Software Testing for Distributed Systems Using Cloud Infrastructure, Automated Fault Injection, and XML Scenarios. International Journal of Information Technology & Computer Engineering, 8(2), ISSN 2347–3657.

[11] Yang, G., Pang, Z., Deen, M. J., Dong, M., Zhang, Y. T., Lovell, N., & Rahmani, A. M. (2020). Homecare robotic systems for healthcare 4.0: Visions and enabling technologies. IEEE journal of biomedical and health informatics, 24(9), 2535-2549.

[12] Rajeswaran, A. (2020). Big Data Analytics and Demand-Information Sharing in ECommerce Supply Chains: Mitigating Manufacturer Encroachment and Channel Conflict. International Journal of Applied Science Engineering and Management, 14(2), ISSN2454-9940

[13] Kotb, Y., Al Ridhawi, I., Aloqaily, M., Baker, T., Jararweh, Y., & Tawfik, H. (2019). Cloud-based multi-agent cooperation for IoT devices using workflow-nets. Journal of Grid Computing, 17(4), 625-650.

[14] Alagarsundaram, P. (2020). Analyzing the covariance matrix approach for DDoS HTTP attack detection in cloud environments. International Journal of Information Technology & Computer Engineering, 8(1).

[15] Hossain, M. S., Muhammad, G., & Guizani, N. (2020). Explainable AI and mass surveillance system-based healthcare framework to combat COVID-I9 like pandemics. IEEE network, 34(4), 126-132.

[16] Poovendran, A. (2020). Implementing AES Encryption Algorithm to Enhance Data Security in Cloud Computing. International Journal of Information technology & computer engineering, 8(2),

[17] Coito, T., Viegas, J. L., Martins, M. S., Cunha, M. M., Figueiredo, J., Vieira, S. M., & Sousa, J. M. (2019). A novel framework for intelligent automation. IFAC-PapersOnLine, 52(13), 1825-1830.

[18] Sreekar, P. (2020). Cost-effective Cloud-Based Big Data Mining with K-means Clustering: An Analysis of Gaussian Data. International Journal of Engineering & Science Research, 10(1), 229-249.

[19] Puttonen, J., Lobov, A., Soto, M. A. C., & Lastra, J. L. M. (2019). Cloud computing as a facilitator for web service composition in factory automation. Journal of Intelligent Manufacturing, 30(2), 687-700.

[20] Karthikeyan, P. (2020). Real-Time Data Warehousing: Performance Insights of Semi-Stream Joins Using Mongodb. International Journal of Management Research & Review, 10(4), 38-49.

[21] Caggiano, A. (2018). Cloud-based manufacturing process monitoring for smart diagnosis services. International Journal of Computer Integrated Manufacturing, 31(7), 612-623.

[22] Mohan, R.S. (2020). Data-Driven Insights for Employee Retention: A Predictive Analytics Perspective. International Journal of Management Research & Review, 10(2), 44-59.

[23] Park, Y., Woo, J., & Choi, S. (2020). A cloud-based digital twin manufacturing system based on an interoperable data schema for smart manufacturing. International Journal of Computer Integrated Manufacturing, 33(12), 1259-1276.

[24] Sitaraman, S. R. (2020). Optimizing Healthcare Data Streams Using Real-Time Big Data Analytics and AI Techniques. International Journal of Engineering Research and Science & Technology, 16(3), 9-22.

[25] Karmore, S., Bodhe, R., Al-Turjman, F., Kumar, R. L., & Pillai, S. K. (2020). IoT-based humanoid software for identification and diagnosis of COVID-19 suspects. IEEE sensors journal, 22(18), 17490-17496.

[26] Panga, N. K. R. (2020). Leveraging heuristic sampling and ensemble learning for enhanced insurance big data classification. International Journal of Financial Management (IJFM), 9(1).

[27] Podpora, M., Gardecki, A., Beniak, R., Klin, B., Vicario, J. L., & Kawala-Sterniuk, A. (2020). Human interaction smart subsystem—extending speech-based human-robot interaction systems with an implementation of external smart sensors. Sensors, 20(8), 2376.

[28] Gudivaka, R. L. (2020). Robotic Process Automation meets Cloud Computing: A Framework for Automated Scheduling in Social Robots. International Journal of Business and General Management (IJBGM), 8(4), 49-62.

[29] Patel, P., Ali, M. I., & Sheth, A. (2018). From raw data to smart manufacturing: AI and semantic web of things for industry 4.0. IEEE Intelligent Systems, 33(4), 79-86.

[30] Gudivaka, R. K. (2020). Robotic Process Automation Optimization in Cloud Computing Via Two-Tier MAC and LYAPUNOV Techniques. International Journal of Business and General Management (IJBGM), 9(5), 75-92.

[31] Villalonga, A., Beruvides, G., Castano, F., & Haber, R. E. (2020). Cloud-based industrial cyber–physical system for data-driven reasoning: A review and use case on an industry 4.0 pilot line. IEEE Transactions on Industrial Informatics, 16(9), 5975-5984.

[32] Deevi, D. P. (2020). Artificial neural network enhanced real-time simulation of electric traction systems incorporating electro-thermal inverter models and FEA. International Journal of Engineering and Science Research, 10(3), 36-48.

[33] El Shenawy, A., Mohamed, K., & Harb, H. (2020). Hdec-posmdps mrs exploration and fire searching based on iot cloud robotics. International Journal of Automation and Computing, 17(3), 364-377.

[34] Allur, N. S. (2020). Enhanced performance management in mobile networks: A big data framework incorporating DBSCAN speed anomaly detection and CCR efficiency assessment. Journal of Current Science, 8(4).

[35] Zhang, T. (2020). Toward automated vehicle teleoperation: Vision, opportunities, and challenges. IEEE Internet of Things Journal, 7(12), 11347-11354.

[36] Deevi, D. P. (2020). Real-time malware detection via adaptive gradient support vector regression combined with LSTM and hidden Markov models. Journal of Science and Technology, 5(4).

[37] Ziouzios, D., Tsiktsiris, D., Baras, N., & Dasygenis, M. (2020). A distributed architecture for smart recycling using machine learning. Future Internet, 12(9), 141.

[38] Dondapati, K. (2020). Integrating neural networks and heuristic methods in test case prioritization: A machine learning perspective. International Journal of Engineering & Science Research, 10(3), 49–56.

[39] Zhang, X., Hu, M., Xia, J., Wei, T., Chen, M., & Hu, S. (2020). Efficient federated learning for cloud-based AIoT applications. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 40(11), 2211-2223.

[40] Dondapati, K. (2020). Leveraging backpropagation neural networks and generative adversarial networks to enhance channel state information synthesis in millimeter-wave networks. International Journal of Modern Electronics and Communication Engineering, 8(3), 81-90

[41] Song, Y., Liu, T., Wei, T., Wang, X., Tao, Z., & Chen, M. (2020). FDA $^ 3$: Federated defense against adversarial attacks for cloud-based IIoT applications. IEEE Transactions on Industrial Informatics, 17(11), 7830-7838.

[42] Gattupalli, K. (2020). Optimizing 3D printing materials for medical applications using AI, computational tools, and directed energy deposition. International Journal of Modern Electronics and Communication Engineering, 8(3).

[43] Ponis, S. T., & Efthymiou, O. K. (2020). Cloud and IoT applications in material handling automation and intralogistics. Logistics, 4(3), 22.

[44] Allur, N. S. (2020). Big data-driven agricultural supply chain management: Trustworthy scheduling optimization with DSS and MILP techniques. Current Science & Humanities, 8(4), 1–16.

[45] Tan, H., Miao, Z., Wang, Y., Wu, M., & Huang, Z. (2020). Data-driven distributed coordinated control for cloud-based model-free multiagent systems with communication constraints. IEEE Transactions on Circuits and Systems I: Regular Papers, 67(9), 3187-3198.

[46] Narla, S., Valivarthi, D. T., & Peddi, S. (2020). Cloud computing with artificial intelligence techniques: GWO-DBN hybrid algorithms for enhanced disease prediction in healthcare systems. Current Science & Humanities, 8(1), 14–30.

[47] Woschank, M., Rauch, E., & Zsifkovits, H. (2020). A review of further directions for artificial intelligence, machine learning, and deep learning in smart logistics. Sustainability, 12(9), 3760.

[48] Kethu, S. S. (2020). AI and IoT-driven CRM with cloud computing: Intelligent frameworks and empirical models for banking industry applications. International Journal of Modern Electronics and Communication Engineering (IJMECE), 8(1), 54.

[49] Zeid, A., Sundaram, S., Moghaddam, M., Kamarthi, S., & Marion, T. (2019). Interoperability in smart manufacturing: Research challenges. Machines, 7(2), 21.

[50] Vasamsetty, C. (2020). Clinical decision support systems and advanced data mining techniques for cardiovascular care: Unveiling patterns and trends. International Journal of Modern Electronics and Communication Engineering, 8(2).

[51] Kumar, S., Raut, R. D., & Narkhede, B. E. (2020). A proposed collaborative framework by using artificial intelligence-internet of things (AI-IoT) in COVID-19 pandemic situation for healthcare workers. International Journal of Healthcare Management, 13(4), 337-345.

[52] Kadiyala, B. (2020). Multi-swarm adaptive differential evolution and Gaussian walk group search optimization for secured IoT data sharing using supersingular elliptic curve isogeny cryptography,International Journal of Modern Electronics and Communication Engineering,8(3).

[53] Ma, Q., Ma, T., Lu, C., Cheng, B., Xie, S., Gong, L., ... & Liu, C. (2020). A cloud-based quadruped service robot with multi-scene adaptability and various forms of human-robot interaction. IFAC-PapersOnLine, 53(5), 134-139.

[54] Valivarthi, D. T. (2020). Blockchain-powered AI-based secure HRM data management: Machine learning-driven predictive control and sparse matrix decomposition techniques. International Journal of Modern Electronics and Communication Engineering.8(4)

[55] Wang, Y., Lin, Y., Zhong, R. Y., & Xu, X. (2019). IoT-enabled cloud-based additive manufacturing platform to support rapid product development. International Journal of Production Research, 57(12), 3975-3991.

[56] Jadon, R. (2020). Improving AI-driven software solutions with memory-augmented neural networks, hierarchical multi-agent learning, and concept bottleneck models. International Journal of Information Technology and Computer Engineering, 8(2).

[57] Asif-Ur-Rahman, M., Afsana, F., Mahmud, M., Kaiser, M. S., Ahmed, M. R., Kaiwartya, O., & James-Taylor, A. (2018). Toward a heterogeneous mist, fog, and cloud-based framework for the internet of healthcare things. IEEE Internet of Things Journal, 6(3), 4049-4062.

[58] Boyapati, S. (2020). Assessing digital finance as a cloud path for income equality: Evidence from urban and rural economies. International Journal of Modern Electronics and Communication Engineering (IJMECE), 8(3).

[59] Jiang, H., Yi, J., Zhou, K., & Zhu, X. (2019). A decision-making methodology for the cloud-based recycling service of smart products: a robot vacuum cleaner case study. International Journal of Computer Integrated Manufacturing, 32(1), 58-71.

[60] Gaius Yallamelli, A. R. (2020). A cloud-based financial data modeling system using GBDT, ALBERT, and Firefly algorithm optimization for high-dimensional generative topographic mapping. International Journal of Modern Electronics and Communication Engineering8(4).

[61] Mishra, A., Karmakar, S., Bose, A., & Dutta, A. (2020). Design and development of IoT-based latency-optimized augmented reality framework in home automation and telemetry for smart lifestyle. Journal of Reliable Intelligent Environments, 6(3), 169-187.

[62] Yalla, R. K. M. K., Yallamelli, A. R. G., & Mamidala, V. (2020). Comprehensive approach for mobile data security in cloud computing using RSA algorithm. Journal of Current Science & Humanities, 8(3).

[63] Ding, H., Gao, R. X., Isaksson, A. J., Landers, R. G., Parisini, T., & Yuan, Y. (2020). State of AI-based monitoring in smart manufacturing and introduction to focused section. IEEE/ASME transactions on mechatronics, 25(5), 2143-2154.

[64] Samudrala, V. K. (2020). AI-powered anomaly detection for cross-cloud secure data sharing in multi-cloud healthcare networks. Journal of Current Science & Humanities, 8(2), 11–22.

[65] Koubâa, A., Ammar, A., Alahdab, M., Kanhouch, A., & Azar, A. T. (2020). Deepbrain: Experimental evaluation of cloud-based computation offloading and edge computing in the internet-of-drones for deep learning applications. Sensors, 20(18), 5240.

[66] Ayyadurai, R. (2020). Smart surveillance methodology: Utilizing machine learning and AI with blockchain for bitcoin transactions. World Journal of Advanced Engineering Technology and Sciences, 1(1), 110–120.

[67] Erasmus, J., Grefen, P., Vanderfeesten, I., & Traganos, K. (2018). Smart hybrid manufacturing control using cloud computing and the internet-of-things. Machines, 6(4), 62.

[68] Chauhan, G. S., & Jadon, R. (2020). AI and ML-powered CAPTCHA and advanced graphical passwords: Integrating the DROP methodology, AES encryption, and neural network-based authentication for enhanced security. World Journal of Advanced Engineering Technology and Sciences, 1(1), 121–132.

[69] Wan, J., Li, X., Dai, H. N., Kusiak, A., Martinez-Garcia, M., & Li, D. (2020). Artificial-intelligence-driven customized manufacturing factory: key technologies, applications, and challenges. Proceedings of the IEEE, 109(4), 377-398.

[70] Narla, S. (2020). Transforming smart environments with multi-tier cloud sensing, big data, and 5G technology. International Journal of Computer Science Engineering Techniques, 5(1), 1-10.

[71] Hartley, J. L., & Sawaya, W. J. (2019). Tortoise, not the hare: Digital transformation of supply chain business processes. Business Horizons, 62(6), 707-715.

[72] Alavilli, S. K. (2020). Predicting heart failure with explainable deep learning using advanced temporal convolutional networks. International Journal of Computer Science Engineering Techniques, 5(2).

[73] Foresti, R., Rossi, S., Magnani, M., Bianco, C. G. L., & Delmonte, N. (2020). Smart society and artificial intelligence: big data scheduling and the global standard method applied to smart maintenance. *Engineering*, *6*(7), 835-846.