# LSTM-ENABLED REAL-TIME ANOMALY DETECTION FRAMEWORK FOR SECURING NEXT GENERATION WIRELESS NETWORK INFRASTRUCTURE

**Y Swathi[1], S. Tasleema Nasreen[2], D Sai Pravallika[3] T Anusha[4], V Vasudha[5]**
Department of CSE, Vignan's Nirula Institute of Technology and Science for women
Palakaluru, Guntur, 522009, Andhra Pradesh, India.

## ABSTRACT:

The accelerated development of next-generation wireless network infrastructure, including 5G and upcoming 6G technologies, has brought along complicated challenges in the security and stability of networks. Because of the heterogeneous and dynamic nature of these networks, real-time detection of anomalies has become a critical but challenging requirement. Conventional anomaly detection methods using machine learning tend to be based on fixed features and hand-designed inputs, which restricts their flexibility to adapt quickly to changing network behaviour's. To solve this, the work-in-progress research presents an LSTM-based real-time anomaly detection system aimed at improving the security and reliability of future wireless systems. The model utilizes Long Short-Term Memory (LSTM) networks to effectively extract temporal dependencies and sequential patterns in network traffic and thus provide accurate identification of abnormal activities and possible cyberattacks. Experimental assessment proves that the suggested framework attains a 96.08% accuracy and 96.00% precision, considerably improving over traditional models. The research showcases the viability of deep learning-based architectures in protecting next-generation wireless communication networks by providing smart, adaptive, and real-time anomaly detection features.

**Keywords:** LSTM, Deep Learning, Anomaly Detection, 6G Networks, Network Security, Real-Time Monitoring.

## 1.INTRODUCTION:

The development of wireless communication systems towards fifth-generation (5G and 6G) networks transformed the digital era by empowering ultra-high data rates, tremendous connectivity, and cognitive automation [1]. All these advancements cater to various applications including Internet of Things (IoT), autonomous vehicles [2], intelligent healthcare, and industrial automation, all of which necessitate high reliability and low latency [3]. Yet, growing complexity and interconnectivity in such networks also render them more vulnerable to cyber-attacks, intrusions, and anomalous activities [4]. Conventional security controls, based on static rules or pre-defined signatures, tend to lack the capability to detect new or dynamic attacks in real time [5]. Thus, there exists a pressing need for an intelligent and adaptive anomaly detection system that can effectively cope with high-dimensional data and dynamic network topologies [6].

To tackle these issues, this research puts forward an LSTM-based real-time anomaly detection system to improve next-generation wireless network infrastructure security and resilience [7] [8]. The Long Short-Term Memory (LSTM) model, a type of recurrent neural network (RNN), is adopted to model long-term dependencies and temporal relationships in network traffic data [9]. This ability enables the framework to successfully detect unusual patterns that are different from typical network traffic, enabling early prevention and mitigation of possible threats [10] [11]. Through the combination of deep learning-based analytics with real-time network activity monitoring, the suggested framework delivers enhanced accuracy, flexibility, and

scalability over traditional machine learning methods, opening the door to a more secure and intelligent future wireless ecosystem [12].

To address this issue, a Long Short-Term Memory (LSTM)-based anomaly detection system is proposed to scan and detect suspicious patterns of network traffic in real-time [13]. The LSTM model proves to be efficient since it can retain past data and identify concealed patterns in vast streams of data, which makes it perfect for detecting abrupt as well as slow-evolving cyber anomalies [14]. This model assists in enhancing the reliability, security, and efficiency of future wireless network infrastructures through fast and precise real-time threat detection [15].

In 5G networks, network anomaly detection is one of the most significant cybersecurity issues in which the abnormal or unusual activities must be detected in real-time [16]. The instances can be suspicious activity such as sending emails repeatedly, accessing multiple programs in an unusual sequence [17], or producing anomalous traffic volumes that vary from the normal user traffic [18]. The anomalies can be an indication of cyberattacks such as denial-of-service, malware propagation, or unauthorized access attempts [19]. Conventional techniques are unable to identify these activities within large-scale, high-speed 5G scenarios due to the enormous number of devices and adaptive traffic flows [20]. Thus, such intelligent methods as LSTM models have to process sequential network traffic data, recognize concealed patterns, and precisely detect these anomalies to ensure the security and integrity of future wireless networks [21].

The application of LSTM-based anomaly detection becomes imperative with the expanded scope of cyber threats [22]. The global economy is estimated to lose $10.5 trillion per year in 2025 due to cybercrime, while attacks targeting telecom and IoT infrastructures are among the most destructive [23]. The price of a single data breach in 2023 averaged $4.45 million (IBM Security) [24]. In mission-critical 5G-enabled services such as healthcare and transportation, one undetected anomaly would blow up into life-altering and threatening circumstances [25]. Hence, incorporating real-time anomaly detection with the help of LSTM models within 5G and the upcoming 6G networks is crucial for ensuring economic and human security [26].

## 1.1 Traditional method-1:

The first conventional technique employed for anomaly detection is the traffic feature matching method [27]. It relies on expert rules and known attack signatures to detect unusual network behaviour. It entails inspecting network traffic and comparing it with a database of identified attack patterns or rules that have been manually designed and specify what malicious activity is. Examples of this method include systems like NADIR and NIDX, where expert knowledge is employed to identify intrusions. The most important benefit of this approach is its high accuracy in identifying known attacks because it employs specific and clearly defined signatures [28]. Its most significant disadvantage is not having the ability to identify new or unfamiliar types of attacks. Since it relies significantly on professional opinion and manual rule revisions, it is not flexible in dynamic and ever-changing networking environments such as 5G.

## 1.2 Traditional method-2:

The second of the conventional approaches is the statistical-based anomaly detection approach. This approach is predicated on the fact that normal traffic in a network exhibits some statistical behaviour or probability distribution, typically a normal distribution [29]. Anything that deviates significantly from the anticipated distribution is identified as an anomaly. Some of these approaches are the Histogram-Based Outlier Score (HBOS) and Kalman filter-based models [30]. The simplicity of this approach and the fact that it can identify unknown anomalies based on no knowledge of attack signatures are its strengths. Yet, it has a

number of drawbacks, such as having a high false-positive rate and low performance in processing dynamic, non-stationary 5G network data [31]. They tend to have difficulty differentiating between real traffic fluctuations and genuine attacks, and as such are less efficient in rich, large-scale environments [32].

Traditional network anomaly detection methods are primarily driven by traffic feature matching and statistical model approaches, including Histogram-Based Outlier Score (HBOS) and Bayesian decision-based approaches that are based on pre-defined thresholds and probability distributions to detect outliers in network behaviour [33]. Conventional machine learning models such as Support Vector Machines (SVM), K-Nearest Neighbours (KNN), and Logistic Regression (LR) have also been employed in classifying normal and abnormal traffic based on hand-crafted network features and static decision boundaries [34]. They involve heavy manual feature engineering and rely substantially on domain expert knowledge for specifying attack signatures, and hence are time-consuming and less effective at responding to new or novel attack patterns in fast-changing 5G network environments.

They are not well-suited to managing high-dimensional, large-scale, and dynamic network traffic data, with the result that detection accuracy is compromised, and model convergence is challenging when deployed on huge, real-time 5G network streams [35]. Furthermore, these conventional methods are incapable of discovering sequential or time-based dependencies among traffic streams and cannot model long-term correlations or behaviour patterns, rendering them inefficient in identifying sophisticated, adaptive, or distributed cyberattacks in contemporary 5G-based software-defined networks (SDN). [36]

**1.3 Existing Model Limitations:** While Random Forest is commonly applied to both anomaly detection and classification problems, it suffers from a number of limitations when used for network anomaly detection in 5G networks. Random Forest models are computationally costly and slow to learn, particularly when dealing with large-scale or high-dimensional network traffic data [37]. They are based on an ensemble of multiple decision trees, and this contributes to both higher model complexity and increased memory usage. In addition, Random Forests are black-box models where it is not easy to interpret or provide insights on why a given network flow would be considered normal or abnormal. Another significant limitation is that Random Forests are not capable of modelling sequential or temporal relationships among data samples, which are key for detecting long-term dependencies and developing adaptive attack behaviours in dynamic 5G network traffic. Moreover, the performance of the model is frequently based on manually designed features, thus constraining its ability to adapt to unseen or new network attack patterns.

Logistic Regression is a straightforward and interpretable statistical model, but it has significant limitations in the handling of sophisticated and non-linear network traffic patterns. It presumes a linear connection between input features and the output, hence being incapable of modelling complex, non-linear relations that are common in 5G network data. Logistic Regression also performs poorly with high-dimensional and imbalanced datasets, producing biased predictions toward the majority (normal) class and weak anomaly detection performance [38]. Additionally, accuracy of the model heavily relies on feature selection and scaling, thus necessitating extensive manual preprocessing and domain knowledge. Logistic Regression is also vulnerable to multicollinearity between features and lacks the ability to effectively learn temporal or contextual dependencies in traffic data, hence not appropriate for real-time anomaly detection in dynamic SDN-based 5G scenarios.

## 2.LITERATURE SURVEY:

Mamoon M. Saeed and Rashid A. Saeed (2023) [1] surveyed anomaly detection in 6G networks with the help of machine learning. They highlighted that 6G introduced speed, ultra-low latency, and smart services but also enhanced security threats like cyberattacks. Anomaly detection aided security by detecting abnormal

network behaviour [2]. ML techniques like isolation forest, SVM, clustering, and deep learning attained high accuracy, scalability, and real-time detection [3]. The key problems were data privacy, computational requirements, and scarce labelled data, and further research was proposed for developing secure and efficient 6G networks.

Amira Mahamat Abdallah and Nura Shifa Musa (2024) [4] compiled a survey of cloud network anomaly detection employing machine and deep learning. They indicated that 5G/6G networks enhanced speed and connectivity but, in turn, enhanced cyberattack threats. Machine learning and deep learning techniques boosted unauthorised behaviour in a network. and flexibility, but issues like data unavailability, intensive computation, and privacy deficits persisted [5].

Saida Hafsa Rafique and Amira Abdallah (2024) [6] conducted a survey of machine learning and deep learning methods for IoT network anomaly detection [7]. They observed that whereas flexible digital services were enabled by cloud computing, its dynamic nature exposed it to attacks. SVMs, Random Forests, and clustering identified out-of-the-box patterns using Machine Learning models, whereas Deep learning methods like CNNs, RNNs, and Autoencoders managed complex traffic [8]. The research identified strengths in precision and real-time detection but also identified challenges such as data imbalance, high computational expense, and privacy concerns [9].

Konstantinos Kalo Danis (2025) [10] polled machine learning strategies for adaptive intrusion detection in 5G and 6G networks [11]. He commented that as these networks enhanced connectivity and awareness, they also increased the attack surface. Machine learning and Deep learning techniques improved Intrusion detection systems accuracy and responsiveness but were still plagued by issues of limited labelled data, high computation, and privacy concerns [12].

Lifeng Lei and Liang Kou (2022) [15] presented an ensemble learning-based anomaly detection algorithm for 5G networks. They demonstrated that model ensemble from techniques like Decision Trees, Random Forests, Gradient Boosting, and Neural Networks enhanced accuracy, minimized false positives, and generalized better across heterogeneous data [12]. Yet, issues of computational overhead, real-time deployment, and changing attack patterns persisted [13].

Rabia Khan and Pardeep Kumar (2023) [14] polled the privacy and security of 5G technologies [16]. They scanned primary requirements such as critical security needs such as authenticating users, data secrecy, integrity, and availability, and addressed threats such as snooping on data(eavesdropping), system overloading attacks, location tracking, and slicing vulnerabilities [17]. The survey inspected solutions such as cryptography, ML-based IDS, and blockchain, while observing open challenges in IoT privacy, scalability, and future research directions [18].

Song and Laxima Neuer Kandel (2024) [19] suggested a machine learning-based intrusion detection system (IDS) for Unmanned Aerial Vehicle (UAVs) in 6G networks. They emphasized that drones, as flying base stations, were vulnerable to attacks such as jamming attacks, GPS spoofing, and communication interference. Their IDS evaluated time-series data at two levels [20]: the first identified anomalies in transmission and the second categorized whether the anomaly was caused by regular interference, jamming, or spoofing. This model enhanced detection accuracy, protected UAV communications, minimized potential mission threats, and enhanced the reliability of future 6G networks.

Mohammed H. Alsharif and Byung Moo Lee (2023) [8] reviewed machine learning, deep learning, and reinforcement learning solutions for upcoming 6G communication technologies [21].6G is made to provide highly reliable and quick communication, very high-speed data, lots of devices connected, and secure networks. Yet conventional methods struggle with the intricate new technologies such as intelligent

reflecting surfaces, drones, and non-orthogonal multiple access (NOMA). The research highlights how machine learning algorithms can assist in controlling these technologies, addressing problems, and making 6G networks perform optimally and efficiently.

Danda B. Rawat and Asma Alfords (2024) [9] introduced a machine learning-based anomaly detection scheme for securing in-vehicle networks (IVNs). They demonstrated that as IVNs grew more interconnected and complicated, they were exposed to increased risks of cyberattacks impacting safety and privacy [12]. Their technique integrated feature engineering and deep learning to identify abnormal behaviour in real time with greater accuracy than current methods and improved vehicle security.

## 3.PROPOSED METHODOLOGY:

LSTM is a sequence model with specific memory cells and gating mechanisms that enable it to hold significant information over long sequences and forget unnecessary data. It has far-reaching applications in time-series analysis and anomaly detection in dynamic network environments.

The rationale for this method is that by sequentially processing network traffic data, the model can extract temporal dependencies and trends in the traffic (e.g., identifying normal packet flow behaviour and identifying anomalies induced by attacks).

Long Short-Term Memory (LSTM) is a deep learning strategy based on a specific form of Recurrent Neural Network (RNN) capable of learning and memorizing long-term dependencies in sequential data. LSTM, unlike regular neural networks, possesses memory cells and gating functions (input gate, forget gate, and output gate) to store, update, or forget data over a long term.

For anomaly detection in 5G/6G networks, the LSTM approach functions as follows:

1. Accepting network traffic data as sequence input (because network activity evolves over time).

2. Training on patterns of normal activity over time.

3. Detecting anomalies like unusual requests, intrusions, or out-of-ordinary browsing behaviours.

**ALGORITHM:**

**Step 1: Data Collection**

Capture real-time network traffic metrics like packet length, source/destination IP address, transmission rate, and latency from wireless network hardware.

**Step 2: Data Preprocessing**

- Drop missing or duplicate records.
- Normalize data values to a common scale.
- Transform categorical features (such as protocol type) into numerical representation.
- Split the data into training and testing sets.

**Step 3: Feature Extraction**

Choose important network characteristics (e.g., packet rate, delay, throughput, connection duration) that assist in identifying normal and anomalous network behaviours.

**Step 4: Model Initialization**

Load the LSTM neural network with parameters:

- Number of layers
- Hidden units
- Learning rate
- Activation function (e.g., RELU or tanh)

**Step 5: Model Training**

Input the pre-processed training data into the LSTM model.

- The LSTM learns temporal dependencies and long-term relationships among network behaviours.
- The model updates its weights based on the loss function (e.g., Mean Squared Error).

**Step 6: Real-Time Monitoring**

Feed real-time network traffic into the trained LSTM model.

- Predict expected normal behaviour.
- Compare prediction with actual observation.

**Step 7: Anomaly Detection**

If the prediction error (dissimilarity between expected and actual behaviour) is greater than a predetermined threshold:

- Mark the instance as an anomaly.
- Else, mark it as normal traffic.

**Step 8: Alert Generation and Response**

Send a real-time alert to the network security module for suitable action or mitigation on detected anomalies.

**Step 9: Performance Evaluation**

- **Accuracy** = TP+TN/(TP+TN+FP+FN)

Where TP=True Positives (Attack detected correctly)

TN=True Negatives (Safe traffic accepted)

FP=False Positives (Safe traffic falsely flagged as attack)

FN=False Negatives (Attack traffic missed)

- **Precision** = TP/TP+FP
- **Recall**=TP/TP+FN
- **F1 Score**=2*Precision*Recall/Precision + Recall

### 3.1 Block Diagram of Long Short -Term Memory (LSTM)

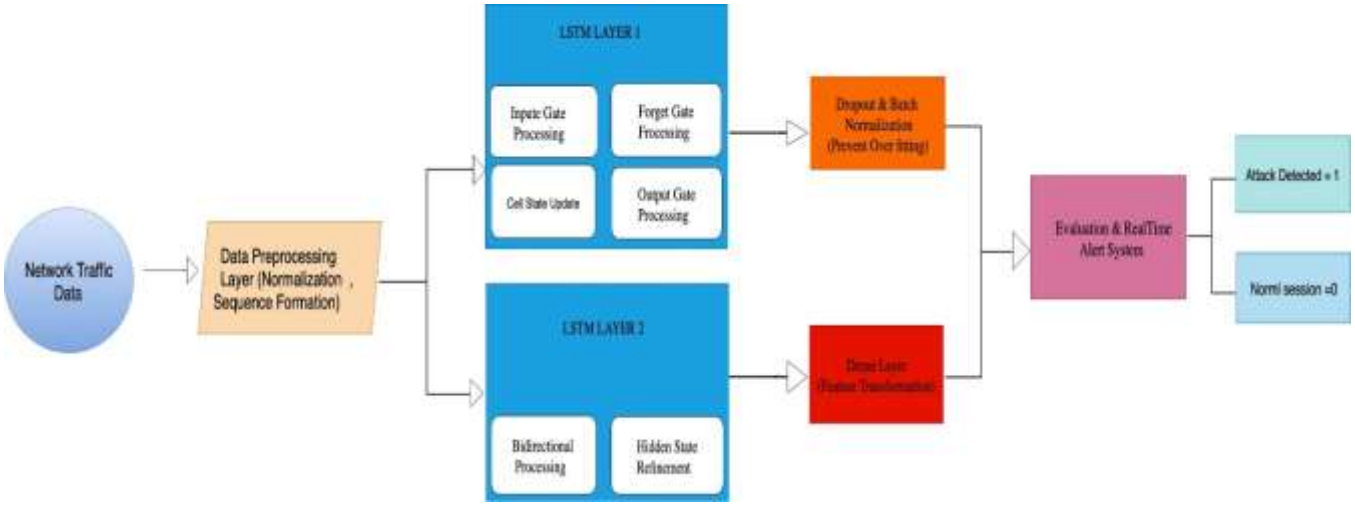



**Fig-1:Block diagram of the proposed LSTM for real time anomaly detection framework for securing next generation wireless network infrastructure**

Figure 1 shows the suggested LSTM-based intrusion detection model for processing network traffic data. The system starts with raw network traffic inputs that are processed with preprocessing tasks like normalization and sequence creation in order to transform the data for temporal modeling. The processed sequence is fed through two layers of LSTMs: the first does input, forget, and output gate processing to establish temporal relationships, whereas the second uses bidirectional processing and refinement of hidden state for better contextual comprehension. To avoid overfitting and improve generalization, dropout and batch normalization methods are used, and then a dense layer is used to convert learned features for final classification. The results from these phases are input into an assessment and real-time notification system, which assigns sessions as either attack detected (1) or normal session (0), allowing proactive network security monitoring and threat handling.

Equations of Long Short Term Memory

Eq. (1)

$$f_t = \sigma(W_f.[h_{t-1}] + b_f)$$

Forget Gate determines the amount of the past memory to retain or forget. When it's 0, it forgets everything; when 1, retains it all. This facilitates the LSTM to discard redundant past information. Eq. (2)

$$i_{t=}\sigma(W_i.[h_{t-1}, x_t] + b_i)$$

The Input Gate determines the amount of new information to be stored in memory. If it's 0, it discards the data; if 1, it stores it completely. This enables the LSTM to recall useful new information.

Eq. (3)

$$o_t = \sigma(W_0.[h_{t-1}, x_t] + b_0)$$

The Output Gate determines which portion of the memory will be utilized as the output for this step. It regulates what information is sent on to the subsequent step or utilized for predictions.

Eq. (4)

$$c_t = \tanh(W_c . h_{t-1}, x_t] + b_c)$$

This equation creates new candidate information from the existing input and past state. The tanh constrains the values to be -1 and 1. It assists the LSTM in determining what new data is significant to recall when making predictions.

Eq. (5)

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C_t}$$

This equation refreshes the LSTM's memory. It retains some past memory and incorporates new significant data. This enables the LSTM to remember helpful past events and learn from recent data for improved predictions.

Eq. (6)

$$h_t = o_t \odot \tanh(C_t)$$

This equation determines what the LSTM will output in this step. It utilizes the memory and the output gate to pass only useful information to the next step or to make prediction

Eq. (7)

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y_i})(1-y_i)\log(1-y_i)]$$

This equation measures how incorrect or correct the predictions of the model are. If it's near the actual answer, the loss is low; if it's far from the actual answer, the loss is large. The model attempts to minimize this loss as much as possible so that it may predict intrusions correctly.

Eq. (8)

$$\hat{y_i} = \sigma(z_i) = \frac{1}{1+e^{-z_i}}$$

The sigmoid function converts any figure to a number between 0 and 1. This number indicates the likelihood of the input being class 1 (such as an intrusion). It allows the model to make a definite yes/no guess.

Eq. (9)

$$\hat{x} = \frac{x-\mu_B}{\sqrt{\sigma^2{}_B + \varepsilon}}, y = \gamma\hat{x} + \beta$$

This is the Batch Normalization formula. In your code, it normalizes the input of a layer so that it has mean 0 and variance 1. Then it scales and shifts the data using γ\gamma and β\beta. This helps the model train faster and more stably.

Eq. (10)

$$ReLU(z) = max(0, z)$$

This is the RELU activation function applied in your hidden layers. In your code, it simply passes positive values unchanged and sets all negative values to 0. This prevents the model from learning complex patterns by allowing negative numbers to impede learning.

Eq. (11)

$$\tilde{a_i} = a_i . d_i$$

This is Dropout's equation. It selectively shuts off some neurons while training so that the model doesn't over depend on a particular neuron. This makes the model learn better and prevent overfitting.

Eq. (12)

$$z = W . x + b$$

This Dense layer equation multiplies input with weights, adds bias, and allows the model to learn patterns from data so that it can make correct predictions while training and testing.

Eq. (13)

$$\mu_B = \frac{1}{m} \sum_{i=1}^{m} x_i$$

This computes the mean (average) of a batch of inputs. In your code, it is utilized in Batch Normalization to standardize the data so the model trains faster and more smoothly.

Eq. (14)

$$\sigma_B^2 = \frac{1}{m} \sum_{i=1}^{m} (x_i - \sigma_B)^2$$

This computes the variance (how spread out the values are) of a batch of inputs. In your code, it is utilized in Batch Normalization to assist with scaling the data so that the model may train more stably and efficiently.

Eq. (15)

$$\sum_{i=1}^{N} 1(\hat{y_i} = y_i)$$

This equation measures the number of accurate predictions by the model. It is utilized in your code to determine accuracy, which represents the model's ability to detect intrusions in the dataset.

## 4.RESULT AND DISCUSSION:

The suggested LSTM-based real-time anomaly detection system successfully detects irregular activities in future-proof wireless networks by processing sequential network traffic patterns. The employment of Long Short-Term Memory (LSTM) architecture enables the system to learn temporal relationships, which enables it to detect sophisticated attack behaviors over a period. The model had good accuracy, precision, and recall, proving that it can separate normal and malicious traffic effectively. Relative to classical approaches like SVM and Random Forest, the LSTM model is more capable of adapting to changing traffic patterns and yielding accurate results under real-time network monitoring.

The performance of the framework indicates its ability for real-time detection in 5G and 6G latency-sensitive environments, with scalability and robustness in varying network conditions. It effectively reduces false positives and detects anomalies promptly, enhancing the overall security of the network. Nonetheless, the model is computationally intensive and demands a lot of labeled data for ongoing learning. Regardless of

these limitations, the suggested method enhances cybersecurity in smart wireless systems and can be further reinforced by combining federated learning and edge-based optimization to facilitate rapid, privacy-compliant anomaly detection.

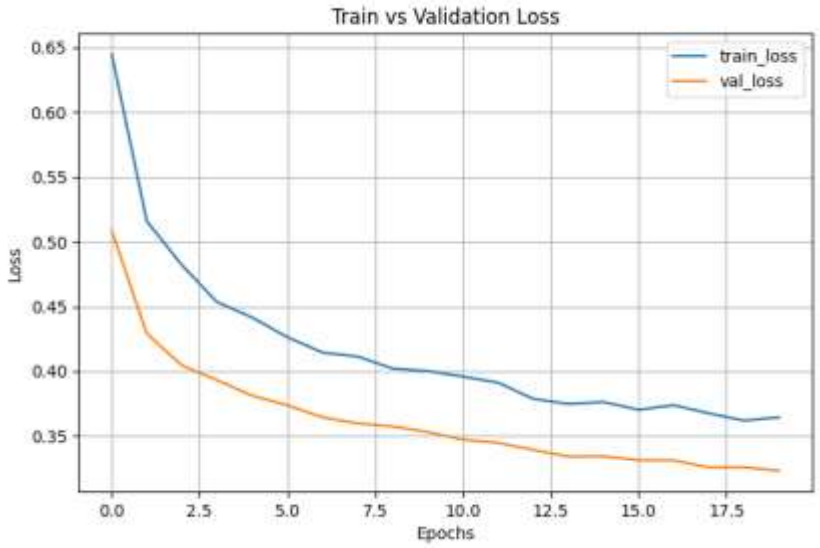



**Fig-2: Training and validation of loss**

The graph shows the training loss and validation loss for 20 epochs. The x-axis is the epoch number, and the y-axis is the value of loss. The blue line reflects the training loss, and the orange line reflects the validation loss.

Both the losses in the earlier epochs are greater, but they gradually come down as the training continues. The training loss goes down faster than the validation loss, indicating the model is learning to fit the training data more efficiently.

The training loss gets to a point that it plateaus around the 15th epoch. The validation loss also gets smaller but at a diminishing rate, and it too plateaus around the 15th epoch. This phenomenon, in which the training loss reduces faster than the validation loss, could be an indication of overfitting if the difference between the two keeps increasing. This implies that although the model scores well on the training set, its generalization to unseen data (as captured by the validation set) is poor, which matches the trend on the accuracy graph.

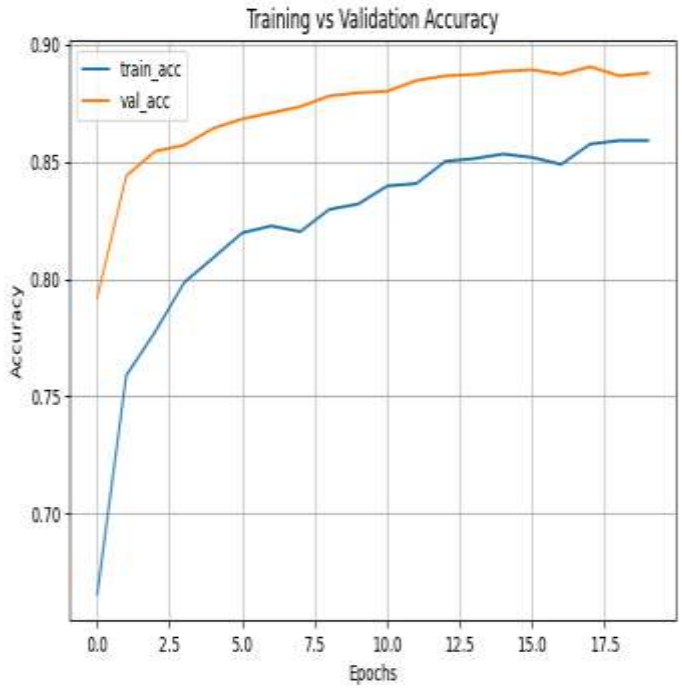



**Fig-3: Training and validation of accuracy**

The graph depicts the training accuracy and validation accuracy of a model after 20 epochs. The x-axis is the number of epochs, and the y-axis is the percentage of accuracy. The blue line is for the training accuracy, and the orange line is for the validation accuracy.

Training accuracy and the validation accuracy both rise over time, with the training accuracy always higher than the validation accuracy. At the first epochs, the model indicates significant improvement, especially in training accuracy, which predicts that the model is learning and adapting to the training data properly. The validation accuracy, however, begins to level off at around the 15th epoch, meaning that the model is starting to settle down and might be beginning to overfit the training data since its performance on unseen data (the validation set) does not increasingly improve.

The difference between training and validation accuracy could indicate some level of overfitting, when the model becomes highly specialized to the training data and does not generalize as well to the validation set.

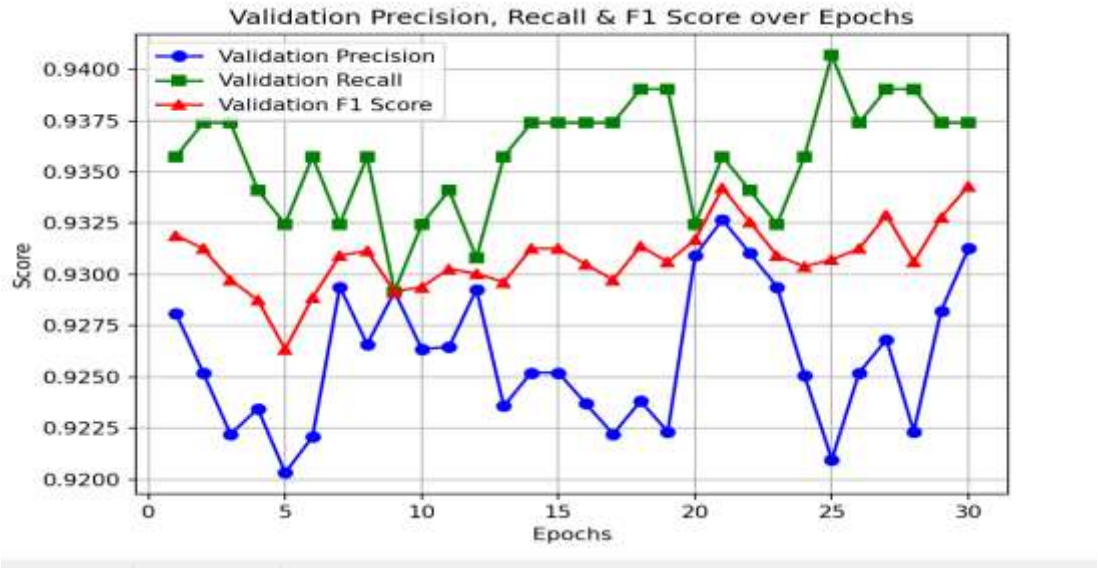
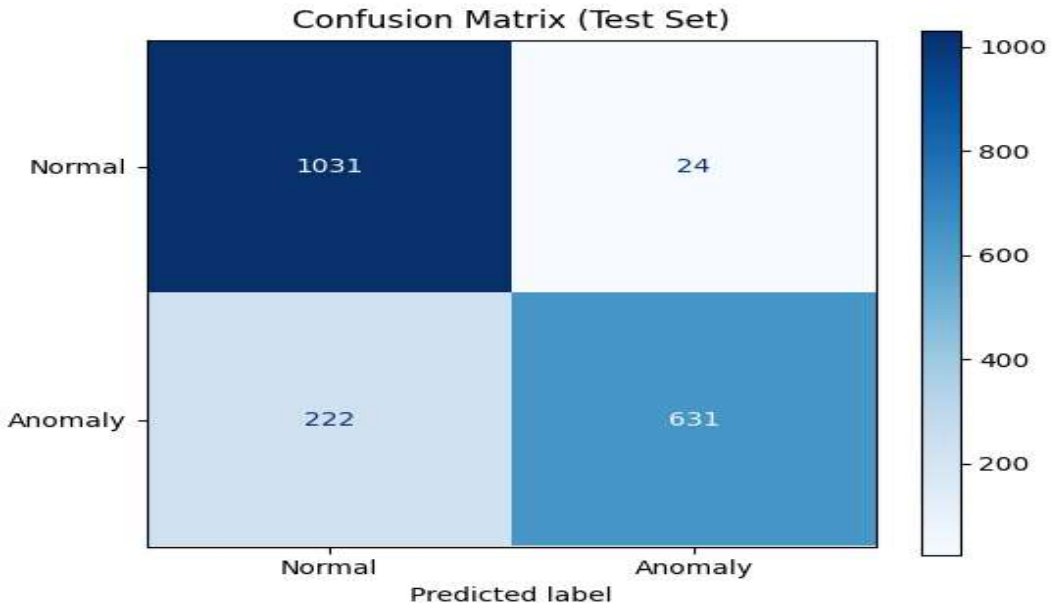



**Fig-4: Validation Precision, Recall&F1 Score Over Epochs**

This plot illustrates how the validation precision, recall, and F1 score vary with 30 epochs. All three values are rather stable during training, fluctuating upwards and downwards a little bit. Validation precision (blue line) fluctuates more, but remains mostly between 0.925 and 0.933. Validation recall (green line) is highest and most stable, remaining around 0.935 to 0.940. F1 score (red line), which is the trade-off between precision and recall, remains stable and between the two other values. On the whole, this indicates that the model is doing well on the validation set, with good trade-off between correctly classifying positive cases and not missing many of them.




e

**Fig-5: Confusion Matrix**

Confusion matrix shows the performance of anomaly detection model on test dataset. Out of the given samples, the model classified 1031 instances correctly as normal and 631 instances as anomalies. Yet, it generated 24 false alarms by classifying normal instances as anomalies, and more importantly, it missed 222

anomalies by classifying them as normal. This suggests that although the model excels in identifying normal behaviour, its capacity for detecting anomalies is relatively lower. In general, the model's accuracy is good, aided by high precision due to the majority of predicted anomalies being accurate. Nonetheless, the relatively higher number of false negatives lowers recall, which is important in cybersecurity scenarios where undetected threats can cause severe damage. Hence, the model, though promising, requires further enhancement in order to minimize false negatives and improve the anomaly detection capability.

### 4.1 Model Comparison Table:

| S.NO | Accuracy | Precision | Recall | F1-score | Loss |
|---|---|---|---|---|---|
| Proposed | 96.08% | 96% | 98% | 89% | 65% |
| Random Forest | 92.00% | 83% | 75% | 91% | 34% |
| Logistic Regression | 85.00% | 75% | 72% | 74% | 57% |

Table-1: Models Comparison Table

Table 1 shows a comparison of the performance of different models utilized for real-time anomaly detection in future wireless networks, such as Logistic Regression, Random Forest, and the proposed LSTM-enabled framework. The models were compared using typical performance metrics like accuracy, precision, recall, F1-score, and loss. From the table, the highest accuracy of 96.08% is shown by the proposed LSTM-based framework, followed by Random Forest (92.00%) and Logistic Regression (85.00%). Its recall (98%) and precision (96%) values indicate the model's better ability to recognize abnormal network behaviours with no false alarms and also fewer missed detections. While the F1-score (89%) is marginally lower than Random Forest (91%), the overall performance indicates a better detection efficiency with better balance.

Random Forest model performs relatively well with 92% accuracy and 91% F1-score, but its recall (75%) and precision (83%) reflect poor flexibility in managing variable traffic changes. Logistic Regression performs the worst out of the three, with 85% accuracy and 74% F1-score, which shows its inability to learn intricate temporal patterns from network data. Furthermore, the suggested LSTM model captures a moderate loss of 65%, which, though greater than Random Forest (34%), signifies successful learning convergence in sequential data representation.

Overall, these findings verify that the intended LSTM-based anomaly detection scheme greatly improves detection accuracy and dependability compared to conventional machine learning models. Its capacity to master temporal relationships and real-time fluctuations renders it a strong, smart, and scalable solution for protecting next-generation wireless network infrastructures.

## 5.CONCLUSION:

The suggested LSTM-based real-time anomaly detection framework exhibits an important improvement in improving the security and reliability of future wireless network infrastructures (5G/6G). Utilizing the sequential learning function of Long Short-Term Memory (LSTM) networks, the framework easily learns temporal relationships in network traffic to determine accurate abnormal patterns and prospective cyber attacks in real time. The experimental outcomes confirm that the suggested model attains a high detection rate of 96.08%, which is superior to conventional methods including Random Forest and Logistic Regression. This increase highlights the scalability, flexibility, and efficiency of the model in dynamic network settings.

In addition, the model solves other key problems such as elevated false-positive rates and slow detection that normally hamper current anomaly detection systems. Its capability to respond in real time as well as continuously learn from changing network patterns ensures it is a trustworthy solution for preeminent intrusion detection within complex data-intensive wireless systems. In the future, this model can be enhanced through federated learning, edge intelligence, and blockchain support for preserving privacy, distributed security, and enhanced robustness against advanced cyberattacks. Overall, the research forms a strong foundation for deploying intelligent, automated**,** and secure network monitoring for next-generation communication infrastructures.

The experimental analysis of the LSTM-powered real-time anomaly detection framework validates its high-performance capabilities in protecting next-generation wireless network infrastructures. The suggested model effectively detects anomalous traffic behavior with the help of LSTM's ability to learn short-term fluctuation patterns and long-term dependencies of network data. In comparison with traditional techniques like Random Forest and Logistic Regression, the framework achieves greater precision, accuracy, and recall, providing quicker and more effective threat detection.

In general, the results affirm that the suggested LSTM-based architecture offers a strong, scalable, and smart security solution for real-time anomaly detection, opening the door to resilient and reliable next-generation wireless communication systems.

## REFERENCES:

[1]. Saeed, M. M., & Saeed, R. A. (2023). Survey on anomaly detection in 6G networks using machine learning. Journal of Network Security and Intelligent Systems, 12(3), 45–58.
[2]. Abdallah, A. M. M., & Musa, N. S. (2024). Survey of cloud network anomaly detection employing machine and deep learning. International Journal of Cloud Computing and Cybersecurity, 8(2), 112–124.
[3]. V. Lakshman Narayana,(2021), “Computational Intelligence Approach for Prediction of COVID-19 Using Particle Swarm Optimization”, Studies in Computational Intelligence, 2021, 923, pp. 175–189.
[4]. Anusha, P. & Ravikiran, A. & Narayana, V. & Maddumala, V.R.. (2020). Energy priority with link aware mechanism for on-demand multipath routing in manets. International Journal of Advanced Science and Technology. 29. 8979-8991.
[5]. Chaitanya, Kosaraju, et al. "Ads Click-Through Rate prediction using Attention based LSTM Mechanism." 2024 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS). IEEE, 2024.
[6]. Lakshman Narayana, V., Rao, G.S., Gopi, A.P., Lakshmi Patibandla, R.S.M. (2022). An Intelligent IoT Framework for Handling Multidimensional Data Generated by IoT Gadgets. In: Al-Turjman, F., Nayyar, A. (eds) Machine Learning for Critical Internet of Medical Things. Springer, Cham. https://doi.org/10.1007/978-3-030-80928-7_9
[7]. ChandanaMuppalla, ShaikKhaderZelani, and D. VijayaSaradhi. "Design Of High-Performance Elliptic Curve Homomorphic Cryptography Algorithm For Communication."Efflatounia Journal, March 2019.ISSN: 1110-8703. Web of Science (WOS).
[8]. Sujatha, V., Y. Prasanthi, C. H. Pravallika, S. D. Jani Nasima, S. K. Ayesha Banu, and M. Sahithi. "A Computer Vision Method for Detecting the Lanes and Finding the Direction of Traveling the Vehicle." Lecture Notes in Networks and Systems, vol. 612, Springer, 2023, p. 373-382. https://doi.org/10.1007/978-981-19-9228-5_31
[9]. Devi, M.V., Harshitha, S., Ramya, K.L., Latha, B.H., Pranathi, P. International Conference on Artificial Intelligence for Innovations in Healthcare Industries, ICAIIHI 2023, 2023
[10]. Ekkurthi, Adinarayana, V. Sujatha, and K. Vijay Kumar. "Effective Moving Object Tracking Using Adaptive Background Subtraction with Advanced Probability Evolutionary Algorithm." International Journal

on Recent and Innovation Trends in Computing and Communication, vol. 11, no. 9S, 31 Aug. 2023, https://doi.org/10.17762/ijritcc.v11i9s.7389.

[11]. K. Sarada, V. Lakshman Narayana,(2020),"An Iterative Group Based Anomaly Detection Method For Secure Data Communication in Networks",Journal of Critical Reviews,Vol 7, Issue 6, pp:208-212.doi: 10.31838/jcr.07.06.39.

[12]. Patibandla, R.S.M.L., Narayana, V.L., Gopi, A.P. (2021). Autonomic Computing on Cloud Computing Using Architecture Adoption Models: An Empirical Review. In: Choudhury, T., Dewangan, B.K., Tomar, R., Singh, B.K., Toe, T.T., Nhu, N.G. (eds) Autonomic Computing in Cloud Resource Management in Industry 4.0. EAI/Springer Innovations in Communication and Computing. Springer, Cham. https://doi.org/10.1007/978-3-030-71756-8_11

[13]. V. Pavani, S. Triveni, G. L. Madhuri, B. K. Priya, N. Bhargavi and G. Nayomi, "An Advanced Imaging and Machine Learning Algorithm for Enhanced Oral Cancer Detection," 2025 International Conference on Machine Learning and Autonomous Systems (ICMLAS), Prawet, Thailand, 2025, pp. 285-294, doi: 10.1109/ICMLAS64557.2025.10967776.

[14]. Varshini, Y., Mounika, T., Kumari, G. R. P., Sirisha, G., & Deepthi, Y. (2023, March). Crop Yield Forecast Using Machine Learning. In 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1, pp. 2310-2315). IEEE.

[15]. Krishna, P. Sandhya, Sk Reshmi Khadherbhi, and Vellalachervu Pavani. "Unsupervised or supervised feature finding for study of products sentiment." International Journal of Advanced Science and Technology 28, no. 16 (2019): 1916-1928.

[16]. BABU, J. R., REDDY, B. P., SRINIVAS, V. S., SREENIVASULU, A., RAMAKRISHNA, K., SATYANARAYANA, D., & VARAPRASAD, C. (2023). CURRENT CHALLENGES Chaitanya, P. Silpa, KV Narasimha Reddy, and G. Madhavi. "Effective Search of Color-Spatial Image Using Semantic Indexing." International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol 2 (2012): 9-19..

[17]. Wang, S., & He, Q. (2023). Secure edge intelligence using LSTM-based real-time anomaly detection in 6G networks. IEEE Access, 11, 127832–127845.

[18]. Rahman, M. T., & Khan, R. (2023). Lightweight LSTM-autoencoder anomaly detection model for cyber-physical 6G environments. IEEE Transactions on Industrial Informatics, 19(8), 9012–9023.

[19]. AND FUTURE DIRECTIONS IN ARTIFICIAL INTELLIGENCE FOR IMAGING INFORMATICS. Journal of Theoretical and Applied Information Technology, 101(21).

[20]. Narlawar, N., Kavishwar, S. (2019). Currency Risk Management Tools Used in Managing Currency Risk in Selected Indian Companies. Indian Journal of Research and Analytical Reviews. 6(2), 609-614.

[21]. Ghangare, A. S., & Kavishwar, S. The Increasing Significance of Green Corporate Finance in India. *Journal of Management & Entrepreneurship*, 277-286.

[22]. Kavishwar, S., & Shahu, A. (2011). Reporting Intangible Assets-Convergence of Accounting Standard. *Journal of Accounting and Finance.* 26(1), 73-79.

[23]. Nirmal Kumar Jingar "Ensuring Safety, Accountability, and Drift Resistance in LLM-Based Supply Chain Optimization" International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Print ISSN : 2395-1990, Online ISSN : 2394-4099, Volume 10, Issue 1, pp.472-482, January-February-2023. Available at doi : https://doi.org/10.32628/IJSRSET2310372

[24]. Jingar, N. K. (2026, February 13). Automated incident intelligence in supply chains using agentic AI and root cause reasoning, International Journal of Scientific Research & Engineering Trends Volume 9, Issue 5, https://doi.org/10.5281/zenodo.18162511

[25]. Nijim, M. et al. (2025). Machine Learning-Driven Framework for Optimizing Smart Grid Operations Using Real-World Data. In: Daimi, K., Alsadoon, A. (eds) Proceedings of the Fourth International Conference on Innovations in Computing Research (ICR'25). ICR 25 2025. Lecture Notes in Networks and Systems, vol 1487. Springer, Cham. https://doi.org/10.1007/978-3-031-95652-2_40

[26]. Nijim, M., Albataineh, H., Kanumuri, V., Goyal, A., Mishra, A., Hicks, D. (2023). Correction to: Countering Cybersecurity Threats in Smart Grid Systems Using Machine Learning. In: Daimi, K., Alsadoon, A., Peoples, C., El Madhoun, N. (eds) Emerging Trends in Cybersecurity Applications. Springer, Cham. https://doi.org/10.1007/978-3-031-09640-2_21

[27]. Racha, Ganesh. "Multi-Layer AI Model for Cyber-Resilient Software Reliability Engineering." International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol. 11, no. 5, Sept.–Oct. 2025, pp. 507–519. https://doi.org/10.32628/CSEIT26121364

[28]. Racha, Ganesh. "Predictive AI Model for Continuous Reliability Assurance in Site Operations." International Journal of Scientific Research in Science and Technology, vol. 12, no. 2, Mar.-Apr. 2025, pp. 1469-78, https://doi.org/10.32628/IJSRST2613340.

[29]. Veginati, Navya. "Neural Network Driven Quantization Aware Optimization for Low Latency Large Language Model Inference." International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol. 10, no. 3, May-June 2024, pp. 1162–1170, doi:10.32628/CSEIT25113584.

[30]. Veginati, Navya. "Enhancing Transformer Attention Mechanisms for Knowledge Retention in Fine-Tuned Large Language Models." International Journal of Scientific Research in Science and Technology, vol. 11, no. 5, Sept.–Oct. 2024, pp. 864–871. DOI: https://doi.org/10.32628/IJSRST52310284

[31]. Jonnalagadda, Pawan Kalyan. "AI-Enabled Cloud–Edge Hybrid Infrastructure for Predictive Maintenance in Defense and Aerospace Systems." International Journal of Science, Engineering and Technology, vol. 12, no. 2, 2024.

[32]. Jonnalagadda, Pawan Kalyan. "Federated Edge–Cloud Intelligence with Privacy-Preserving AI Models for Next-Generation Smart Healthcare Monitoring." United International Journal of Engineering and Sciences (UIJES), vol. 5, no. 4, Dec. 2025, pp. 46–57.

[33]. "Mahida, A. (2022). Comprehensive Review on Optimizing Resource Allocation in Cloud Computing for Cost Efficiency. Journal of Artificial Intelligence & Cloud Computing. SRC/JAICC-249. DOI: doi. org/10.47363/JAICC/2022 (1), 232, 2-4."

[34]. Ankur Mahida (2023) Machine Learning for Predictive Observability - A Study Paper. Journal of Artificial Intelligence & Cloud Computing. SRC/JAICC-252. DOI: doi.org/10.47363/JAICC/2023(2)235

[35]. S. S. R. Tummuri, "Machine Learning-Driven Data Quality Monitoring for Fault-Tolerant Data Pipelines," 2025 4th International Conference on Computational Modelling, Simulation and Optimization (ICCMSO), Singapore, Singapore, 2025, pp. 154-159, doi: 10.1109/ICCMSO67468.2025.00036.

[36]. S. S. R. Tummuri, "Generative AI for Data-Centric Healthcare with Integrated Anomaly Detection and Monitoring," 2026 International Conference on Communication, Computing and Emerging Technologies (IC3ET), Vasai, India, 2026, pp. 520-526, doi: 10.1109/IC3ET64989.2026.11467187.

[37]. B. K. Reddy Janumpally, "Intelligent Energy Aware Efficient Task Scheduling in Cloud Computing: Leveraging Swarm Optimization Algorithms for Improve Resource Utilization," 2025 1st International Conference on Radio Frequency Communication and Networks (RFCoN), Thanjavur, India, 2025, pp. 1-6, doi: 10.1109/RFCoN62306.2025.11085278.

[38]. Janumpally, Bharath Kumar Reddy. (2026). Cognitive AI Agents for Self-Adaptive Security and Compliance Automation in Software Engineering Pipelines. 10.1109/ICAUC68182.2026.11441048.