International Research Journal of Multidisciplinary Scope (IRJMS), 2025; 6(3):889-901 IRJMS

Original Article | ISSN (O): 2582-631X DOI: 10.47857/irjms.2025.v06i03.04288

Blockchain-Enabled Collaborative Threat Intelligence in IoT Security Using a Hybrid Neural Network Model

Prasanna Simhadati[1], C Hrishikesava Reddy[2], R Gomathi[3], Supriya Telsang[4], K Jayaram Kumar[5], A Barkathulla[6], V Bhoopathy[7*]

[1]Department of Computer Science and Engineering, Artificial Intelligence and Machine Learning, GMR Institute of Technology, Rajam, Andhra Pradesh, India, [2]Department of Computer Science and Engineering, Rajeev Gandhi Memorial College of Engineering and Technology, Nandyal, Andhra Pradesh, India, [3]Department of Artificial Intelligence and Data Science, Bannari Amman Institute of Technology, Sathyamangalam, Tamil Nadu, India, [4]Department of Computer Engineering, Vishwakarma Institute of Technology, Pune, India, [5]Department of Electronics and Communication Engineering, Aditya University, Surampalem, Andhra Pradesh, India, [6]Department of Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R and D Institute of Science and Technology, Chennai, Tamil Nadu, India, [7]Department of Computer Science and Engineering, Sree Rama Engineering College, Tirupathi, India. *Corresponding Author's Email: v.bhoopathy@gmail.com

Abstract

The fast spread of Internet of Things (IoT) devices over many different fields has made network security even more crucial. Conventional security systems can fail to handle the dynamic and complex character of contemporary cyber threats aiming at IoT systems. This paper suggests a novel security framework combining blockchain technology, machine learning (ML), and a centralized iOS application to get past these constraints. The suggested approach guarantees privacy, integrity, and immutability of shared Cyber Threat Intelligence (CTI) data by using smart contracts and the Ethereum blockchain. Fundamentally, a hybrid deep learning model CNNTransLSTM is used to highly precisely detect and categorize threats in real-time. Combining Transformer encoders, Long Short-Term Memory (LSTM) networks, and Convolutional Neural Networks (CNN), this model efficiently records spatial and temporal aspects of IoT network data. By allowing users to report hazards and get alerts, the iOS app serves as an interactive hub improving human-machine cooperation. CNNTransLSTM model beats conventional approaches in terms of accuracy, sensitivity, and loss rate according to experimental evaluations. Moreover, the distributed blockchain architecture enables among stakeholders safe, open, and cooperative threat intelligence sharing. This all-encompassing strategy enables users and cloud providers to make quick, well-informed decisions to reduce risks, hence greatly improving the resilience of IoT ecosystems.

Keywords: Blockchain, Convolutional Neural Network Transformer (CNNTrans), Cyber Threat Intelligence (CTI), Internet of Things (IoT), Long Short-Term Memory (LSTM), Threat Intelligence (TI).

Introduction

In recent years, computer technologies have advanced swiftly and persist in their evolution. This development has also incurred several adverse repercussions. Concurrently with this improvement, there is a consistent rise in cyber-attacks. Due to digitization, both large corporations and small enterprises, including individual users, have grown increasingly aware of the privacy and security of their data, as a significant amount of their personal information is held in cyberspace (1). Numerous firms conduct research on detection and prevention systems to enhance the security of their systems against cyber-attacks. Their experiences are archived as a knowledge base within their systems that generate intelligence. This intelligence can be acquired either through experiencing specific attacks or by retrieving information from servers maintained and shared by various security firms or organizations. The disseminated information is referred to as CTI. Blockchain refers to a sequence of interconnected blocks. Every block is interconnected with both its preceding and succeeding blocks. The blockchain has recently garnered interest. A multitude of researchers from both corporate and academic sectors have commenced investigations into applications that can be built using this technology (2). The blockchain can be characterized as a data storage system functioning as a public ledger. Transactions executed using blockchain technology are recorded in blocks within a chain.

(Received 04th March 2025; Accepted 07th July 2025; Published 26th July 2025)

Upon the addition of a new chain, it undergoes ongoing growth. The primary benefit of blockchain technology is its cryptographic security. Altering a block inscribed on the blockchain is nearly infeasible. Moreover, a blockchain possesses characteristics including decentralization, persistence, and auditability. The blockchain functions in a decentralized setting by incorporating several essential technologies, including cryptographic hashing, digital signatures (utilizing asymmetric cryptography) and distributed consensus techniques.

Blockchain technology facilitates the approval and publication of transactions in a decentralized manner. For instance, funds can be exchanged between two accounts without the involvement of a central authority (bank). This decentralized framework, which abolishes central authority, can diminish expenses and enhance productivity. Blockchain can facilitate monetary transfers and various financial applications, including online payments and the management of digital assets. Moreover, blockchain can be utilized in applications Examples include smart contracts for services for the public, the Internet of Things (IoT), reputational systems, and safety services. Given low latency and little resource use, Proof of Authority (PoA) or PBFT is most likely the consensus mechanism fit for IoT. For limited devices, these techniques are perfect since they do not need great computation. In a permissioned blockchain design, they guarantee fast data validation and integrity.

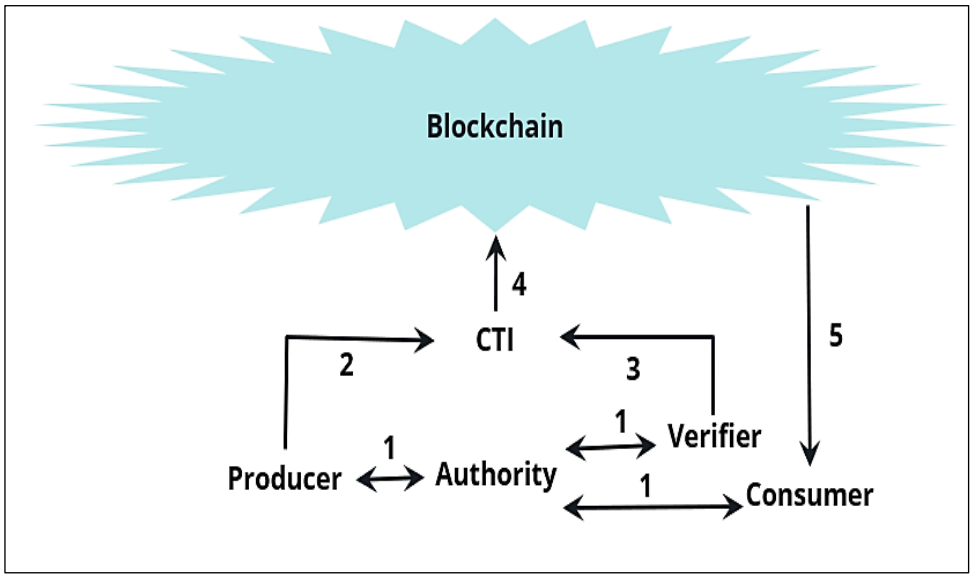

Figure 1: Blockchain Based TI Framework Sharing

Figure 1 illustrates the sequential phases involved in the communication process among the different components of the framework.

Step 1: All parties present their proof of identity to a trustworthy authority. Proof of identification may entail revealing data such as government-issued credentialing (e.g., driver's license or passport), having access to third-party documents, or business accreditation.

Step 2: The producer creates CTI and adds it to a blockchain for confirmation.

Step 3: The verifier evaluates the reliability of the CTI using an established set of standards provided by the network.

Step 4: The CTI considered valid in Step 3 is included on the blockchain.

Step 5: Consumers have access to the CTI which has been integrated into the blockchain.

Figure 2 is a simplified sharing model that demonstrates how blockchain might fundamentally improve CTI sharing.

The proposed system emphasise the benefits and drawbacks of past studies by surveying the literature on issues such as blockchain technology, iOS applications, IoT security, and Machine Learning (ML) techniques (3). There have been a lot of studies looking at how to use ML algorithms to make the Internet of Things more secure. An extremely successful method for identifying malicious activities in IoT network data by relying on Deep Learning (DL) is stated in past study (4). The prospect of detecting botnet attacks and

malware on the Internet of Things using Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), two kinds of DL algorithms are found in past research (5). However, these studies just in terms of IoT security, they emphasise data integrity and authentication; yet, their research fails to utilise collaborative threat intelligence (6). An approach to Sharing information about cyber threats to detect network intrusions based on Federated Learning (FL) researchers invented in the past (7). Confronting the challenge of developing ML-based detection systems, this study makes use of varied data from many sources and organisations. IoT security concerns and Software Defined Networks (SDN) solutions using ML is stated in past study (8). Their paper briefly covered ML approaches that reduce security weaknesses, such as Neural Networks (NNs) for data validity assessment and Bayesian learning for cross-layer harmful assault detection (9, 10). However, the paper does not examine all ML methods. After testing several detection algorithms for each set of characteristics, Naïve Bayes (NB), Random Forest (RF), J48, K-nearest Neighbours (KNN), and Support Vector Machine (SVM) were the most popular. Hybrid analysis enabled flexibility in selecting static and dynamic features to improve detection accuracy. However, this essay only covers one application and one security concern (malware) (11). evaluated if Blockchain techniques could improve IoT data security. The investigation found security vulnerabilities in Blockchain techniques is found in past research (12). The study cited majority assaults, double-spend attacks, and denial-of-service attacks as security threats. The poll advised CNNs and Deep Neural Network (DNNs) for security (13). Using a mix of extreme ML techniques, a better threat-hunting model for IoT malware and ransomware is stated in past study (14). Comparing its performance against that of well-known Deep NN models, such as CNN and stacked Long short-term Memory (LSTM), revealed satisfactory results. Similar to this, the study proposed a method for detecting malware on Windows, Android, and the IoT utilising ensemble learning based on threat-hunting models (15). The authors showed that by combining strong ML algorithms, they may improve the process of searching for IoT computer viruses (16). Its performance was comparable to that of other kinds of deep neural networks, such as CNN and stacking LSTM. Jointly detecting distributed denial of service (DDoS) attacks by merging smart contracts with a Fuzzy Neural Network (FNN) (17). While protecting user privacy, the public can get data on unusual traffic thanks to the Blockchain. Critical for real-time IoT security, quick consensus with minimum delay and high throughput is accomplished with PoA/PBFT. It guarantees good performance and helps to prevent delays typical in PoW systems. Using a permissioned blockchain, only verified entities may safely access, validate, and distribute threat intelligence.

Methodology

The IoT is growing as one of the fastest accepted technologies in the past decade across diverse applications. The intelligent devices are interconnected either wirelessly or via wired connections for communication, processing, computation, and monitoring various real-time situations (18). The authors carefully examine the three principal technologies: DL, AI, and Blockchain, to address security concerns in the IoT. This Proposed study addresses a blockchain-based threat intelligence sharing system based on tamper-proof ledger recording of CTI data production, verification, and access guarantees ensuring data provenance. Smart contracts limiting and recording access to CTI entries (Step 1–5 in Figure 1) and identity verification systems. E.g., government-issued credentials, corporate certifications and implement the access control. Although model synchronization is not specifically addressed in terms of federated learning or multi-node training synchronization, the architecture guarantees that threat intelligence, once validated it is made available to all authorized nodes in a distributed and secure manner utilizing blockchain. Blockchain is thus employed here not for deep model synchronization but rather for provenance/access control mechanism and threat intelligence exchange. This dataset comprises network traffic data that emulates diverse forms of communication among network entities, with particular emphasis on various protocols and potential security vulnerabilities (19). The data encompasses details regarding packets transmitted between sender and receiver entities, their characteristics, and related attack kinds.

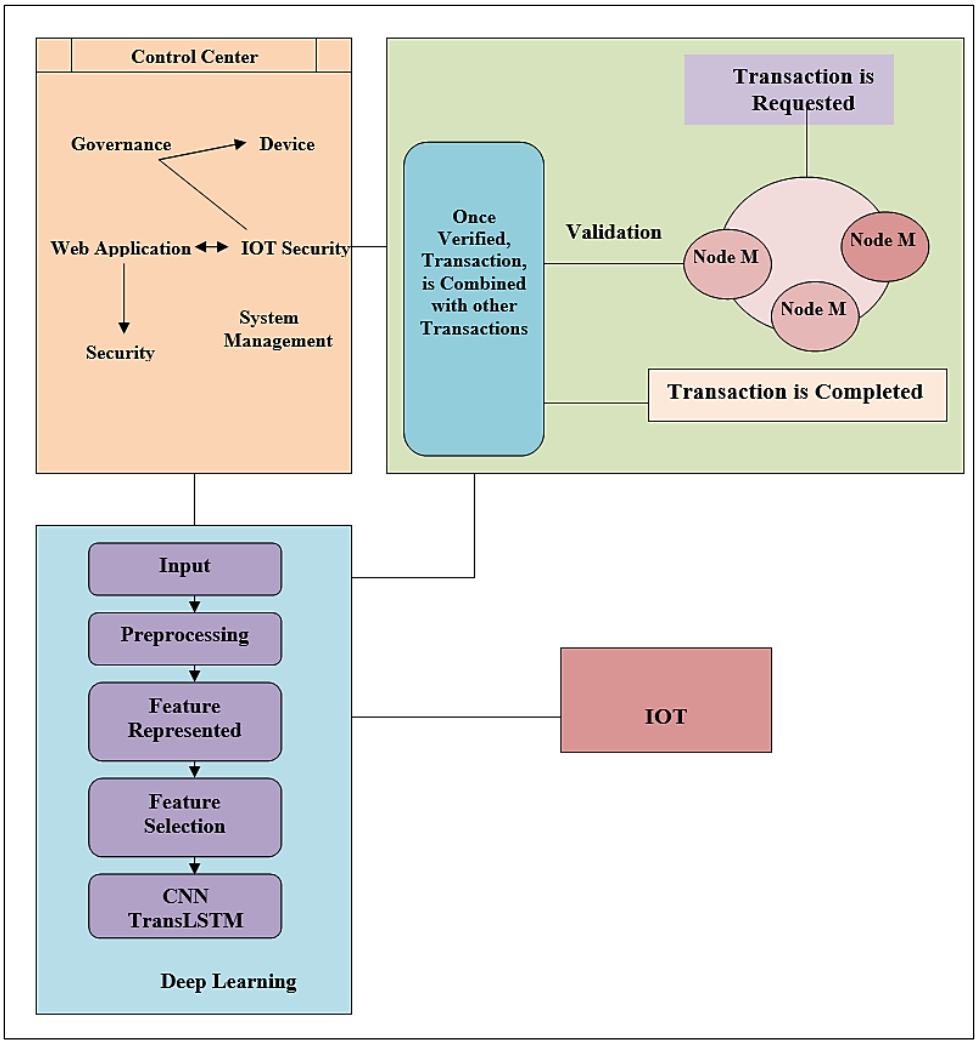

Figure 2: Proposed Model Conceptual Framework

Figure 2 delineates the principal components of the model. IoT devices, which create an interconnected network spanning several domains, are introduced at the beginning of the framework. This section outlines a comprehensive architecture aimed at enhancing the security of IoT networks. Our plan employs an integration of a blockchain technology, iOS central hub, and DL, highlighting the essential contribution of human expertise (20). The technology functions in real time, facilitating swift hazard identification and response while continuously evolving to tackle emerging safety issues. CNNTransLSTM deep learning is included into the suggested architecture for strong temporal and spatial analysis. An iOS central center for hazard alerting and real-time communication. Blockchain for unchangeable, safe storing of danger intelligence. This solution is suitable as centralized models usually create hazards and IoT systems are heterogeneous and resource-limited. Blockchain offers a distributed and tamper-proof way to document threat information. Improved attack detection accuracy (97.25%) using the hybrid CNN-Transformer-LSTM model covers both spatial patterns and sequential temporal dependencies.

Due to their interconnectivity, these devices produce data and are susceptible to security breaches. The ML models constitute the framework's most critical element. The aforementioned models are trained on pertinent datasets, including the Cyber Risk Data, in order to identify trends, identify dangers, and classify security incidents instantly. The iOS app, which acts as a central hub, is the second element. The app has an easy-to-use interface for reporting threats and interacting with machine learning models. It enables users to report potential risks and furnish essential information, like the type of threat, timing, and origin. Moreover, the program utilises features such as notifications and encrypted channels of communication to facilitate

prompt response and identification of threat. The third essential element is the execution of blockchain technology. The deep learning models detect and categorise a threat, and the data is sent between the models of DL and the application of iOS. Upon the identification of a hazard by the DL models, the threat data and device details are securely transmitted to the network of blockchain. The blockchain network acts as an open and unchangeable repository for threat data, keeping it safe and encouraging many people to work together.

Preprocessing

Collection of Data: Cyber threat data can be sourced from multiple origins, including network logs, security apparatus (e.g., firewalls, intrusion detection systems), threat intelligence feeds, and user activity records. These sources offer critical insights on network traffic, system events, and user behaviour, which can be examined to identify potential risks.

Missing Values Handling: Deficiencies in cyber threat data may occur owing to multiple factors, including sensor malfunctions or inadequate log records. Imputation methods, like mean imputation, regression imputation, or algorithms like k-nearest neighbours (KNN), can be utilised to address missing variables (21). If the missing data are considered significant, the relevant instances or characteristics may be eliminated.

Dealing the Outliers: Outliers in cyber threat data may signify unusual activity or measurement inaccuracies. Statistical methods, like z-scores and interquartile range (IQR), facilitate the identification and management of outliers. Outliers may be eliminated, modified, or classified as a distinct category based on the context during model training.

Normalization of Data and Standardization: Methods for data normalisation include min-max scaling and z-score normalisation, adjust numerical features to a uniform scale, ensuring no single feature predominates the study. Standardization methods, like mean removal and unit variance scaling, guarantee that features possess a mean of zero and a variance of one, which can enhance the performance of specific machine learning algorithms.

Feature Representation

A corpus was developed to transform words (tokens) into their corresponding numerical values, depending on the frequency of unique tokens in every class. The quantitative textual representation, specifically TF-IDF, was computed using the subsequent equation:

$$sd_{idh} = sh.log\ log\ \frac{M}{eh} \quad [1]$$

In this context, sh represents the term frequency of the word in a particular occurrence, eh denotes the word document frequency, and M signifies the samples of total number in the dataset. The term sh denotes the frequency of a term's occurrence inside a sample, whereas inverse document frequency idh indicates the reciprocal of the variety of documents that include the word. The larger the word frequency, the inverse document frequency sh_{idh} of the more words in a paper, the document is more relevant. This phase produced three numerical vectors for each sample.

Feature Selection

The elevated dimensionality of the extraneous information hindered the differentiation between benign and malicious URLs. As a result, the learning problem became more complex, leading to inferior training accuracy. Likewise, the properties of the Who is information and URL included extraneous details, particularly when the N-gram approach was employed. The attributes were doubled according to the n-value of the N-gram. Furthermore, FS is a popular strategy for text characteristics (22). Therefore, choosing features is critical in this research. This study selected the top 5,000 characteristics to limit the risk of information loss while increasing the generalisability of trained models. The informative value of low probability features is higher than that of high probability features (common features). CI-based FS use entropy to assess feature impurity throughout the partitioning of the target variable. Entropy can be calculated using Equation [2]. Increased entropy corresponds to more information content. Entropy is a mathematical expression as:

$$D(q) = -\sum_{j=1}^{m} q_j\ log\ log\ (q_j) \quad [2]$$

Where m is the class of target, and q_j denotes the feature probability partitioning class j. The following formula can be used to calculate the IG, which represents the split's quality.

$$Gain = 1 - D(q) \quad [3]$$

Where m is the class target of the entropy and $Gain$ denotes the split quality. A trait is pertinent for categorisation if it exhibits a substantial gain. An increase in benefit correlates with a reduction in entropy. A zero entropy indicates a less impure division. This phase yields a feature vector consisting exclusively of high-gain features.

CNNTransLSTM Model Training

This section initially presents a formulaic depiction of the CTI problem, followed by an elaboration on the various components of CNNTransLSTM as shown in Figure 3.

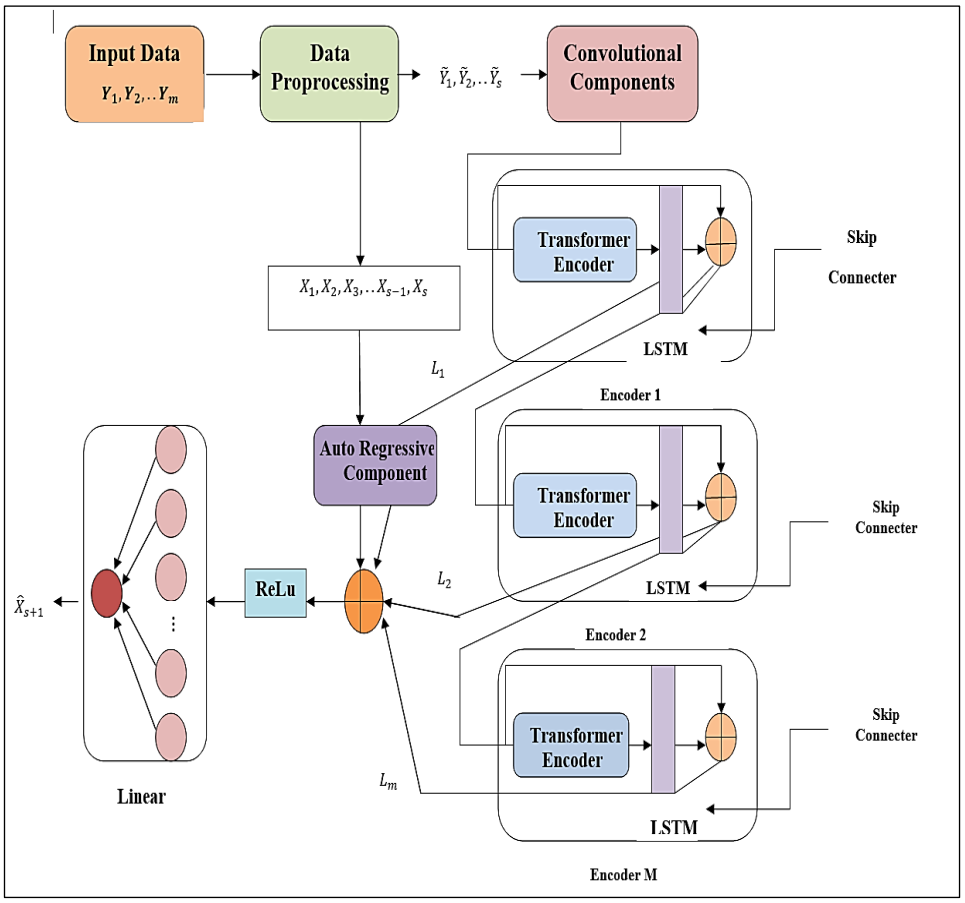

Figure 3: Proposed Algorithm Overall Structure

CNN Component

In contrast to the single-kernel convolutional layer, the multi-kernel convolutional layer possesses a broader receptive field, resulting in more enriched spatial feature extraction, which enhances the prediction accuracy of CNNTransLSTM. Therefore, this paper proposes a module based on the Inception module, referred to as Multi-kernel CNN. The program can standardise the output dimensions of different layers by modifying the kernels and paddings of each layer. The outputs from all convolutional layers can be consolidated to derive the spatial feature of final vectors. When the input data is $\tilde{Y} = (\tilde{Y}_1, \tilde{Y}_2, \ldots \tilde{Y}_{sv})^S$, the total input dimension can be articulated as (s_v, F, V), where s_v denotes the sliding window length, F signifies the height of $\tilde{Y}_1$, and V indicates the breadth of $\tilde{Y}_1$. The dimension of output convolutional component is denoted as $(s_v, b, F-1{,}1)$, with b indicating the channels number, which is established as 1 in this research. To enhance clarity, the extraneous dimensions of the output are removed; hence, the final output dimension is $(s_v, F-1)$. In order to emphasise the temporal information, different multi-kernel CNNs process data from every time step.

Component of Encoder

Given the Transformer's ability to process long text inputs efficiently, we combine the Transformer's encoder layer with an LSTM network to develop an encoder component that can extract temporal information from prolonged input sequences. The main purpose of the skip link between encoder modules is to lessen degradation problems brought on by an excessively deep network. The residual connection in the module alleviates the problem of gradient vanishing.

Component of LSTM

Although the Transformer encoder can obtain positional information through the Encoding layer, some degree of information loss concerning word locations may still occur. Unlike obtaining positional information through functions, the LSTM network directly extracts it from the input sequence, potentially enhancing the model's sensitivity to temporal data. In order to better capture the time dependence of variables, the gating mechanism has the ability to retain temporal information. Thus, we include an LSTM network layer with the Encoder transformer layer to jointly extract temporal data. The input, output, and forgetting gates make up the majority of the LSTM network architecture. The forgetting gate regulates the internal state b_{s-1} from the preceding moment to determine the extent of information to be discarded. The input gate is utilised to regulate the candidate state$\tilde{b}_s$at the present moment to determine the amount of information retained. The output gate l_s regulates the internal state b_s at the present instant to determine the extent of information conveyed to the external statef_s.

The forgetting gate, input gate, and output gate are delineated as follows:

$$f_s = \alpha(v_h.[f_{s-1}, y_s] + c_h) \quad [4]$$

$$j_s = \alpha(v_j.[f_{s-1}, y_s] + c_j) \quad [5]$$

$$l_s = \alpha(v_l.[f_{s-1}, y_s] + c_l) \quad [6]$$

The present candidate state $\tilde{b}_s$, memory unit b_s, and external state $f_s$are computed as follows:

$$\tilde{b}_s = tan\ tan\ f\ (V_b.[f_{s-1}, y_s] + c_b) \quad [7]$$

$$b_s = h_s\ .b_{s-1} + j_s\ .\tilde{b}_s \quad [8]$$

$$f_s = l_s\ .tan\ tan\ f\ (b_s)\ . \quad [9]$$

$V_h, c_h, V_j, c_j, V_l, c_l, V_b, c_b$ are parameters subject to training.$\alpha(.)$ and$tan\ tan\ f(.)$ are both activation functions. The formulas are as follows:

$$\alpha(y) = \frac{1}{1 + d^{-y}} \quad [10]$$

$$f(y) = \frac{d^{y} - d^{-y}}{d^{y} + d^{-y}} \quad [11]$$

Component of Decoder

This component first consolidates the output from the previous M encoder layers with the AR component, then decodes the aggregate using the Linear layer and RELU function to get the final prediction result $\hat{X}_{s+1}$. The following formula is used to determine the decoder layer:

$$\hat{X}_{s+1} = Linear\left(ReLU\left(\sum\ (L_1, L_2, \ldots . L_m, L_{AR})\right)\right) \quad [12]$$

Where L_{AR} denotes the output of theAR model

AR Component

The model's output scale is insensitive to the input scale because to the non-linear properties of the CNN layer, LSTM network and Transformer encoder. As a result, we perceive the CNNTransLSTM's final predicting output as a blend of a linear and non-linear component. This paper uses the AR model for the linear component. To improve prediction accuracy, it may extract the target variable's linear correlation from previous data. The following is how the AR model is expressed:

$$L_{AR} = \tilde{X}_{s+1} = v_1 x_1 + v_2 x_2 +, \ldots . v_s x_s + \in_{s+1}$$
$$= \sum_{r=1}^{s} v_r x_r + \in_{s+1}, \quad [13]$$

v_r and $\in_{s+1}$ are parameters that can be trained. v_r denotes the weight of the variable, $\in_{s+1}$ signifies the random noise, and x_r indicates the historical value of the target variable.

Algorithm 1

Algorithm for Proposed Model

Algorithm 1: Flow of Proposed Model
Procedure DataHandleFlow
Obtain data from the IoT Security Ecosystem
Examine data within the Deep Learning Layer
if a potential threat is recognized, then
Initiate alert
Configure push notifications in the iOS Central Hub
The security of IoT researcher expert corroborates the threat.
Document results in Feedback Loop (Blockchain) and enhance machine learning models.
End if
End

A lot of data and network traffic are made by a lot of gadgets in the Ecosystem of IoT Security. This stream of data is the engine that keeps the system running and is the main source of information about possible risks. The data is quickly sent to the Layer of ML after being continually gathered. Real-time analysis of incoming data is done using a wide range of methods. They have two goals: to find deviations from known data trends and to identify any threats in security within the data. The ML Layer sounds a warning as soon as it finds a possible threat. This starts a chain reaction of cooperation. Emulating real-world network traffic and risks, the simulated IoT setup consists in IoT devices producing data via different communication protocols. Among others, DDoS, C and C, File Download (malware) attack forms. Network design: Data moves from IoT sensors → ML-based detection layer (CNNTransLSTM) → iOS app for alerting → blockchain for logging and distribution. While the LSTM detects temporal relationships, the CNN component uses multi-kernel convolutions for spatial feature extraction; the transformer encoder improves sequential understanding. This framework is intended to record local as well as long-range dependencies of threat patterns.

Results and Discussion

The advent of Internet of Things (IoT) technology has instigated a significant transformation in the cyber threat landscape. Nevertheless, the majority of current systems are predominantly centralised and fail to facilitate information sharing in a dispersed manner. This chapter aims to assess how blockchain technology might mitigate several limitations inherent in current CTI sharing systems. To ascertain the future role of blockchain-based sharing, we outline many overarching difficulties associated with CTI sharing and examine how blockchain can provide secure and efficient solutions to these challenges. Although specific artificial data augmentation methods (e.g., SMote, GANs) are not discussed, the system compensates for imbalanced data via focused sampling which is more especially, under-sampling the majority class by deleting outliers. This method lowers training bias and guarantees a more balanced dataset.

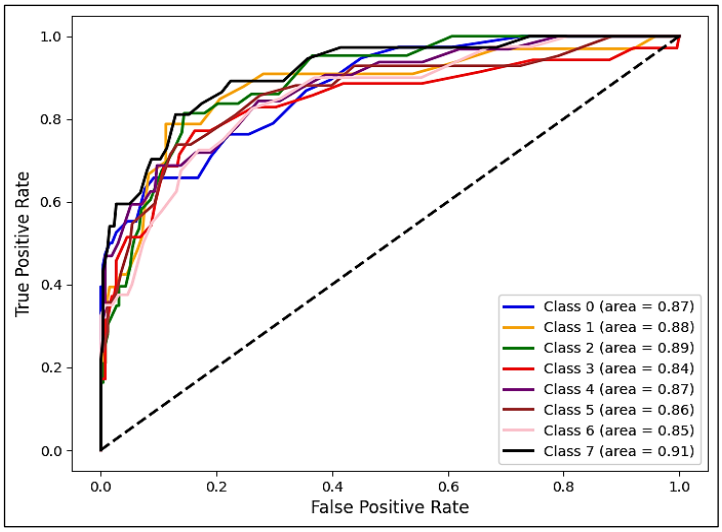

Figure 4: ROC Curve for CNNTransLSTM Model

DDoS, Command and Control (C and C), malware downloads, and botnet traffic are among the system's detects. It makes the presumption that attackers may circumvent conventional protection, start stealthy attacks, and take advantage of IoT weaknesses. Skilled in network evasion and remote access, attackers call for distributed detection and response systems. The ROC curve showed that the predictor really was very good at what it did. The threats 'Attack,' 'C and C-FD', 'FileDownload,' and 'DDoS,' had an exceptional ROC area score of 0.97%. This dual evaluation emphasizes the characteristics of the ensemble classifier and identifies opportunities for enhancing its detection of threat capabilities. Based on the image in Figure 4.

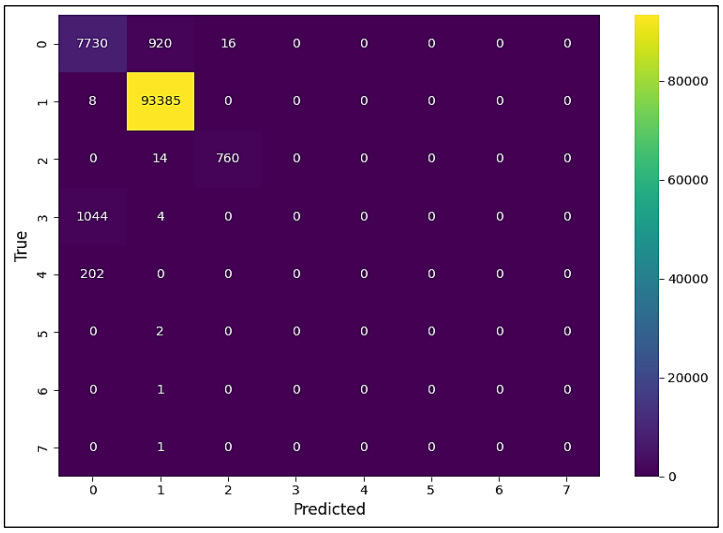

Figure 5: Proposed Model Confusion Matrix

The Random Forest classifier demonstrated exceptionally favourable outcomesROC plots for each class showed how well the model could tell the difference between them. It showed AUC values of 0.97 for classes that represented major risks, like DDoS attacks and Command and Control (C and C) operations. This means that the classifications were very accurate. This highlights the RF's capability in precisely detecting and differentiating harmful network behaviours. These findings bolster the validity of our research. Figure 5 shows how well the CNNTransLSTM Classifier worked on the dataset we used for our study project together.

Table 1: Training and Test Time Analysis of Various Models

Rate of Learning	CNN	Trans-LSTM	GRU	Att-GRU	BiLSTM-BiGRU	CNNTransLSTM
Accuracy	90.86	95.54	91.45	94.32	93.26	97.25
Precision	76.25	90.46	80.39	83.93	85.74	97.12
Sensitivity	76.25	90.46	80.39	83.93	85.74	97.12
Specificity	95.38	98.99	94.83	98.66	97.66	98.45
F-Measure	76.25	90.46	80.39	83.93	85.74	97.12
MCC	82.66	90.73	78.22	86.76	80.53	89.64

The suggested research introduces a technique termed focused sampling to mitigate the uneven impact of datasets by under-sampling the majority class. This strategy entails creating a subset by deliberately identifying and eliminating outliers from the predominant class. Eliminating these outliers facilitates a more equitable representation of the dataset. Table 1 presents the evaluation metrics for many models, including Att-GRU, BiLSTM-BiGRU, CNNTransLSTM, and Trans-LSTM.

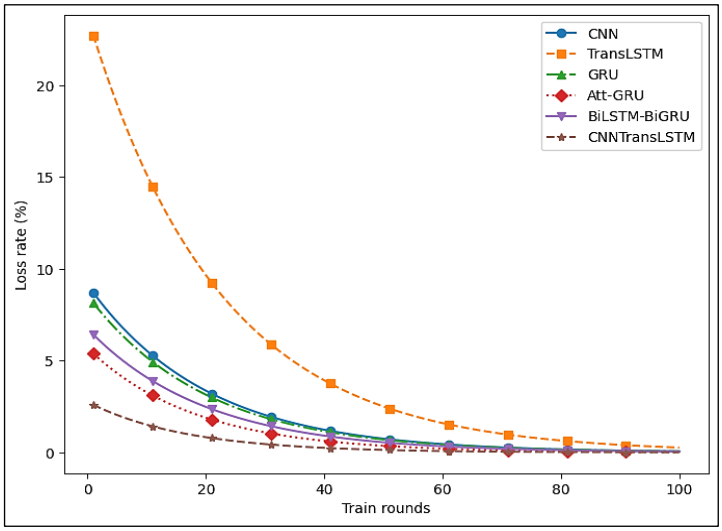

Figure 6: Accuracy Values Comparison of the Models

The examination of Figures 6 and 7 demonstrates that the suggested model in this paper has superior convergence speed and enhanced generalization capability relative to previous models. Furthermore, the precision of the proposed model exceeds that of alternative models, despite an equivalent amount of training iterations. The accuracy of the proposed model exhibits stability during the training phase, showing minimal changes. The suggested model achieves a commendable final training accuracy of 97.25% on the validation set. This research presents a model that demonstrates superior performance and significant advantages. It not only attains superior accuracy relative to conventional models but also exhibits accelerated convergence and enhanced generalization capability. The findings indicate that the suggested model had considerable potential and may yield improved performance in practical applications.

The narrative encompasses various models, each distinguished by unique markers and line styles for enhanced clarity. CNN (blue solid line with circular markers): Figure 7 Demonstrates a consistent decline in loss, achieving stable convergence near 2%. TransLSTM (orange dashed line with square markers): Initiates with the highest loss rate (>20%) and exhibits a rapid decline, ultimately stabilizing at a comparatively elevated loss (~5%) relative to other models.

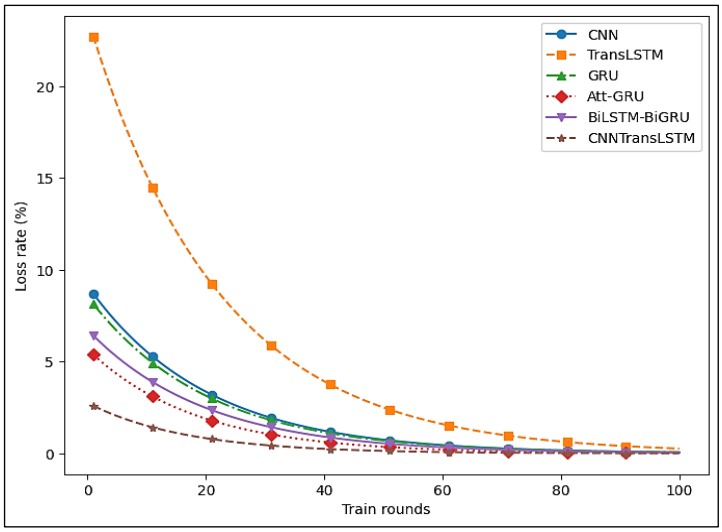

Figure 7: Loss Values Comparison of the Models

GRU (green dash-dot line with triangular markers): Demonstrates a consistent decline in loss, culminating just below 4%. Att-GRU (red dotted line with diamond markers): Rapidly declines and stabilizes at approximately 2%. BiLSTM-BiGRU (purple dashed line with arrow markers): Progressive decline in loss, stabilizing around 3%. CNNTransLSTM (brown dash-dot line with star markers): Demonstrates superior performance, achieving rapid convergence with the minimal loss rate (~1%). The legend on the right enumerates the models along with their respective colours and markers for straightforward identification. The plot demonstrates that CNNTransLSTM attains the minimal loss, establishing it as the superior model among those evaluated.

Table 2: Performance Classification of CNNTransLSTM under Combination of Training, Testing and Validation

Train:Val:Test	Accuracy Classification (%)			Time (ms)
	Train	Val	Test	
60:10:30	97.56	94.73	92.28	13.8
60:20:20	97.25	94.92	92.16	13.4
70:10:20	97.03	95.77	93.68	13.2
70:15:15	97.14	95.58	93.84	12.6
80:10:10	96.28	95.77	94.02	13.4

This section demonstrates the thorough optimisation and classification effectiveness of the hybrid CNN-Trans-LSTM model across several combinations of training, validation, and test data, with results shown in Table 2. The partial discharge classification accuracy in training sets across different combinations is more than 95%, which shows that the model has converged close to the global optimum and fully matched the training data. Additionally, there is no overfitting in the model, as seen by the test set's classification accuracy being equivalent to the validation set's (i.e., accuracy loss not exceeding 4%). The hybrid CNN-TransLSTM model's resilience and generalisability to a wide range of data combinations are demonstrated by its test classification accuracy, which exceeds 97% across all combinations with a variance of less than 3%. With a 97% classification accuracy in discharge pattern identification over 300 test samples, only 12 samples were incorrectly identified, meeting the requirements for real-world applications. On the other hand, the model's inference efficiency will not be impacted by sample set partitioning during the testing phase, suggesting that the model can be successfully applied in a variety of contexts.

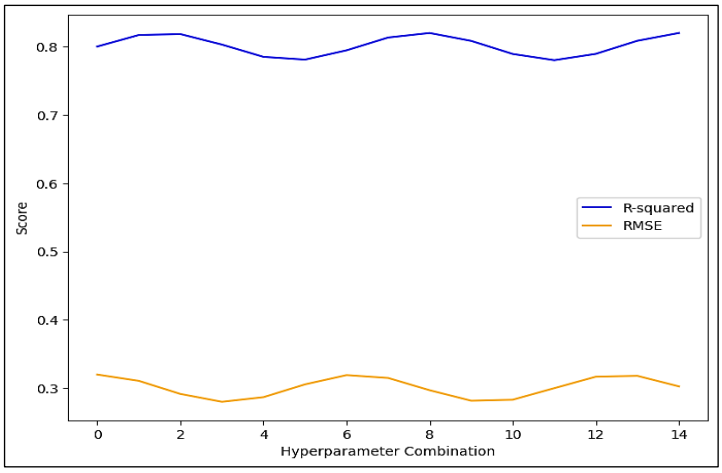

Figure 8: Results of Hyperparameter Tuning

It used cross-validation and a grid search method to look into the hyperparameter space in depth. As part of the grid search, all possible values for each hyperparameter were carefully looked at, and the model's success was judged using different methods, including negative RMSE and R-squared (r2). To get the highest R-squared number, which means that our model fits the data the best, we had to find the best hyperparameters using those factors. This strict process made sure that the model we used was optimised to make the most accurate predictions possible. Figure 8 illustrates the outcomes of hyperparameter tuning, including the optimal hyperparameters and a comparison of models prior to and after to hyperparameter optimization.

Benefits and drawbacks of decentralization consist in Transmission latency in transactions: Slightly greater due to consensus procedures like PoW or PoS; but, this is lessened by only providing authenticated CTI data, which is not common. Agreement Overhead: Though it adds computational expenses, distributed consensus guarantees data integrity. Still, in this paradigm, given the security advantages, it is seen reasonable. burden of storage: Blockchain size can increase with time since all CTI transactions are kept immutably. But the character of shared CTI data—textual threat descriptors, hashes—allows the load to be controlled.

Conclusion

Applications that track numerous variables, such as smart city monitoring systems, rely on the security of the IoT. Using blockchain technology to keep tabs on IoT networks is the focus of this research. Objective functions that are parameterised are used in the analysis. In order to monitor and assess the progress of each activity in real-time, it is essential to establish distinct job execution intervals in the IoT. To strengthen data security in smart city apps, the proposed method combines CNNTransLSTM algorithms with blockchain technology. Data security in processing and storage units is compromised due to the deployment of IoT throughout the process. Therefore, at every step, monitoring units depend on utmost confidence. CNNTransLSTM is specifically integrated in the architecture for accurate threat detection; blockchain for safe CTI sharing; and an iOS hub for real-time alerting. It guarantees decentralization, anonymity, and group work. Important contributions include low 1% loss rate, 97.25% detection accuracy, 0.97 ROC AUC for severe assaults, and blockchain-based validation procedures for safe CTI sharing.

Abbreviations

TI: Threat Intelligence, CNN: Convolutional Neural Network, IoT: Internet of Things, NN: Neural Networks.

Acknowledgement

The author would like to appreciate the effort of the editors and reviewers.

Author Contributions

All authors are equally contributed.

Conflict of Interest

The authors declare that they have no conflicts of interest.

Ethics Approval

There are no human subjects in this article and

informed consent is not applicable.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

References

1. Büber E, Şahingöz ÖK. Blockchain based information sharing mechanism for cyber threat intelligence. Balkan Journal of Electrical and Computer Engineering. 2020 Jul;8(3):242-53.
2. Dunnett K, Pal S, Jadidi Z. Challenges and Opportunities of Blockchain for Cyber Threat Intelligence Sharing. Smart Sensors, Meas Instrum. 2022;43:1–24.
3. Latif S, Idrees Z, e Huma Z, Ahmad J. Blockchain technology for the industrial Internet of Things: A comprehensive survey on security challenges, architectures, applications, and future research directions. Trans Emerg Telecommun Technol. 2021;32(11): e4337.
4. Ravi V, Chaganti R, Alazab M. Recurrent deep learning-based feature fusion ensemble meta-classifier approach for intelligent network intrusion detection system. Computers and Electrical Engineering. 2022 Sep 1;102:108156.
5. Idrissi I, Boukabous M, Azizi M, Moussaoui O, El Fadili H. Toward a deep learning-based intrusion detection system for IoT against botnet attacks. IAES International Journal of Artificial Intelligence. 2021 Mar 1;10(1):110.
6. Zhao J, Yan Q, Li J, Shao M, He Z, Li B. TIMiner: Automatically extracting and analyzing categorized cyber threat intelligence from social data. Computers & Security. 2020 Aug 1;95:101867.
7. Sarhan M, Layeghy S, Moustafa N, Portmann M. Cyber threat intelligence sharing scheme based on federated learning for network intrusion detection. Journal of Network and Systems Management. 2023 Jan;31(1):3.
8. Restuccia F, D'oro S, Melodia T. Securing the internet of things in the age of machine learning and software-defined networking. IEEE Internet of Things Journal. 2018 Jun 11;5(6):4829-42.
9. Khurana N, Mittal S, Piplai A, Joshi A. Preventing poisoning attacks on AI based threat intelligence systems. In2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP). IEEE. 2019 Oct 13:1-6. https://ieeexplore.ieee.org/abstract/document/8918803/
10. Hassija V, Chamola V, Saxena V, Jain D, Goyal P, Sikdar B. A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures. IEEE Access. 2019;7:82721–43.
11. Sharmeen S, Huda S, Abawajy JH, Ismail WN, Hassan MM. Malware Threats and Detection for Industrial Mobile-IoT Networks. IEEE Access. 2018;6:15941–57.
12. Reyna A, Martín C, Chen J, Soler E, Díaz M. On blockchain and its integration with IoT. Challenges and opportunities. Future generation computer systems. 2018 Nov 1;88:173-90.
13. Liu Q, Li P, Zhao W, Cai W, Yu S, Leung VC. A survey on security threats and defensive techniques of machine learning: A data driven view. IEEE access. 2018 Feb 13;6:12103-17.
14. Darabian H, Dehghantanha A, Hashemi S, Taheri M, Azmoodeh A, Homayoun S, Choo KK, Parizi RM. A multiview learning method for malware threat hunting: windows, IoT and android as case studies. World Wide Web. 2020 Mar;23(2):1241-60.
15. AL-Hawawreh M, Sitnikova E, Den Hartog F. An efficient intrusion detection model for edge system in brownfield industrial internet of things. ACM Int Conf Proceeding Ser. 2019:83–7. https://dl.acm.org/doi/abs/10.1145/3361758.3361762
16. Bai J, Wang J. Improving malware detection using multi-view ensemble learning. Secur Commun Networks. 2016;9(17):4227–41.
17. Han X, Zhang R, Liu X, Jiang F. Biologically inspired smart contract: A blockchain-based DDoS detection system. In2020 IEEE international conference on networking, sensing and control (ICNSC). IEEE. 2020 Oct 30:1-6. https://ieeexplore.ieee.org/abstract/document/9238104/
18. Alkadi O, Moustafa N, Turnbull B, Choo KKR. A Deep Blockchain Framework-Enabled Collaborative Intrusion Detection for Protecting IoT and Cloud Networks. IEEE Internet Things J. 2021;8(12):9463–72.
19. Zunxi H. Cyber Threat Data for New Malware Attacks. Kaggle. https://www.kaggle.com/datasets/zunxhisamniea/cyber-threat-data-for-new-malware-attacks
20. Nazir A, He J, Zhu N, et al. Collaborative threat intelligence: Enhancing IoT security through blockchain and machine learning integration. J King Saud Univ - Comput Inf Sci. 2024;36(2):101939.
21. Brightwood S, Seraphina Brightwood A. Data Preprocessing and Feature Engineering for Cyber Threat Detection. 2024 Mar 16. https://www.researchgate.net/profile/Seraphina-Brightwood/publication/379078896_Data_Preprocessing_and_Feature_Engineering_for_Cyber_Threat_Detection/links/65f9fa02a8baf573a1c5dc21/Data-Preprocessing-and-Feature-Engineering-for-Cyber-Threat-Detection.pdf
22. Alsaedi M, Ghaleb FA, Saeed F, Ahmad J, Alasli M. Cyber threat intelligence-based malicious URL detection model using ensemble learning. Sensors. 2022 Apr 28;22(9):3373.