SPECTRUM OF ENGINEERING SCIENCES
ISSN (E) 3007-3138 (P) 3007-312X VOLUME 4, ISSUE 6, 2026

AN AI-DRIVEN BLOCKCHAIN-BASED CYBERSECURITY FRAMEWORK FOR SECURE CLOUD COMPUTING ENVIRONMENTS

Ahmed Wali Khan[*1], Ali Muhammad[2], Farhan[3], Abdul Salam[4], Abdul Karim Kashif Baig[5], Muhammad Tahir[*6], Nauman Hafeez Ansari[7]

[*1]*Department of Computer Science, Iqra University Main Campus, Defense View, Karachi City – 75500, Sindh Pakistan*
[2,5,6]*Department of Computer Science, Faculty of Engineering, Science and Technology (FEST), Iqra University Main Campus, Defense View, Karachi City – 75500, Sindh Pakistan*
[3]*Department of Software Engineering, Air University, Karachi Campus, Karachi, Sindh, Pakistan*
[4]*DHA Suffa University, Karachi City, Sindh, Pakistan*
[7]*Department of Computer Science, Mohammad Ali Jinnah University (MAJU) Karachi - 75400, Sindh Pakistan*

[*6]muhammad.tahir01@iqra.edu.pk

DOI:https://doi.org/10.5281/zenodo.20728226

Keywords:
Cloud Computing; Blockchain Security; Cybersecurity; Artificial Intelligence; CNN-LSTM; Intrusion Detection System; Smart Contracts; Distributed Security; Deep Learning

Article History:
Received: *18 April 2026*
Accepted: *29 May 2026*
Published: *17 June 2026*

Corresponding Author: *
Ahmed Wali Khan
Co- Corresponding Author: *
Muhammad Tahir

Abstract:
Cloud computing has emerged as a foundational technology for modern digital infrastructure due to its scalability, flexibility, and cost-efficiency. However, the increasing adoption of cloud platforms has introduced significant cybersecurity challenges, including unauthorized access, data breaches, Distributed Denial-of-Service (DDoS) attacks, spoofing, insider threats, and data tampering. Traditional cloud security mechanisms suffer from centralized vulnerabilities, limited scalability, and inadequate real-time attack detection. To address these limitations, this paper proposes an AI-Driven Blockchain-Based Cybersecurity Framework (AIBCF) for secure cloud computing environments. The proposed framework integrates blockchain technology with a hybrid Convolutional Neural Network–Long Short-Term Memory (CNN-LSTM) model to provide decentralized trust management, intelligent intrusion detection, and adaptive threat mitigation. Blockchain ensures secure authentication, immutable transaction logging, and smart contract-based enforcement, while the CNN-LSTM model performs real-time cyberattack detection and classification. Experimental evaluation on the CICIDS2017 dataset under DDoS, spoofing, brute force, and infiltration scenarios achieved 98.2% accuracy, 97.6% precision, 97.1% recall, and 97.3% F1-score, with a false positive rate of 1.8%, outperforming existing machine learning and blockchain-based baselines. Ten-fold cross-validation confirmed stable results (accuracy: 98.2% ± 0.4%). The findings indicate that integrating blockchain with AI-driven mechanisms significantly improves cloud security, reliability, and adaptive defense capabilities.

1. INTRODUCTION

Cloud computing has transformed modern computing infrastructures by enabling on-demand access to computational resources, storage services, and distributed applications over the Internet [18]. Organizations across healthcare, finance, education, industrial automation, and e-commerce increasingly rely on cloud platforms due to their scalability, flexibility, and reduced operational costs [17]. Despite these advantages, cloud environments remain highly vulnerable to cybersecurity threats such as data breaches, DDoS attacks, insider attacks, malware injection, unauthorized access, and data manipulation [19].

Centralized cloud security architectures introduce critical vulnerabilities including single points of failure and inadequate adaptive cyber-defense capabilities [17]. Conventional authentication and access control mechanisms are often unable to support decentralized trust management in large-

scale dynamic cloud environments. Furthermore, the sophistication of modern cyberattacks demands intelligent real-time intrusion detection systems capable of analyzing complex network traffic and identifying malicious behaviors dynamically [9].

Blockchain technology has emerged as a promising solution for enhancing cloud security due to its decentralized architecture, immutability, transparency, and distributed trust management [14]. Blockchain enables secure authentication, tamper-resistant transaction storage, and automated smart contract enforcement without relying on centralized authorities [2]. However, most existing blockchain-based cloud security systems focus primarily on authentication and data integrity while lacking intelligent cyberattack detection capabilities [13].

Artificial Intelligence (AI) and deep learning techniques have demonstrated strong performance in cybersecurity. Deep learning models such as CNNs, LSTMs, and hybrid architectures have achieved high accuracy in intrusion detection, malware analysis, and anomaly detection [5,6]. CNN models extract spatial attack features from network traffic, whereas LSTM networks learn temporal attack behaviors and sequential intrusion patterns [7]. A hybrid CNN-LSTM model can leverage both capabilities synergistically for more accurate and robust detection [10,25].

Although blockchain and AI have individually improved cybersecurity, limited research has integrated both into a unified intelligent cloud security framework. Existing approaches suffer from high computational overhead, poor scalability, insufficient real-time attack mitigation, and reliance on static security policies [21,22]. To address these gaps, this paper proposes the AI-Driven Blockchain-Based Cybersecurity Framework (AIBCF), which integrates blockchain-based decentralized authentication with a hybrid CNN-LSTM intrusion detection model.

The main contributions of this paper are as follows:

1. A novel AI-driven blockchain-based cybersecurity framework (AIBCF) for secure cloud computing environments is proposed, integrating decentralized authentication with intelligent attack detection.
2. A hybrid CNN-LSTM model is developed for intelligent real-time intrusion detection, combining spatial feature extraction (CNN) with temporal sequence learning (LSTM) on 1D network traffic feature vectors.
3. Smart contract mechanisms are implemented in Solidity on a private Ethereum network for automated cloud security policy enforcement and access control.
4. A rigorous experimental evaluation is conducted using the CICIDS2017 dataset with 10-fold cross-validation, multiple attack categories, and comparison against five established baselines.
5. The framework demonstrates improved attack detection accuracy (98.2% ± 0.4%), reduced false positive rates (1.8%), and a blockchain verification success rate of 99.1%.

The remainder of this paper is organized as follows. **Section 2** presents the related work. **Section 3** explains the proposed methodology and system architecture. **Section 4** discusses experimental results and performance evaluation. **Section 5** concludes the paper and presents future research directions.

2. Related Work

The rapid expansion of cloud computing has increased the importance of cybersecurity in distributed environments. Research has progressed along three parallel tracks: blockchain-based cloud security, AI-driven intrusion detection, and integrated blockchain-AI frameworks.

2.1 Blockchain-Based Cloud Security

Wang et al. [14] provided a comprehensive survey of blockchain challenges and opportunities, identifying decentralized trust, data integrity, and transaction transparency as core benefits for cloud ecosystems, while also highlighting scalability and computational overhead as open challenges. Liang et al. [15] proposed a blockchain-based framework for secure data sharing in mobile healthcare, demonstrating the viability of smart contract-driven access control. Latif et al. [13] conducted a systematic review of blockchain-based secure cloud computing frameworks, confirming the effectiveness of distributed ledgers for authentication and data integrity, but noting that intelligent threat detection remains largely absent in these systems. Ahmed et al. [26] proposed a blockchain-enabled secure authentication framework specifically for cloud computing environments, achieving reliable decentralized verification but without any AI-based attack detection component. These works collectively establish that blockchain improves transparency

and immutability, yet struggle with intelligent adaptive defense.

2.2 AI-Based Intrusion Detection Systems

Xin et al. [9] reviewed machine learning and deep learning methods for cybersecurity, establishing the superiority of deep learning architectures over traditional methods for complex attack pattern recognition. Roopak et al. [10] evaluated multiple deep learning models for IoT network security and found that hybrid architectures outperform standalone CNN or LSTM models. Javaid et al. [11] proposed a deep learning approach for intrusion detection that achieved high classification accuracy on the NSL-KDD dataset, though generalization to real-world cloud environments was limited. Moustafa and Slay [12] introduced the UNSW-NB15 dataset as a benchmark for IDS evaluation, highlighting the need for realistic traffic-based evaluation. Patel and Verma [25] specifically proposed a hybrid CNN-LSTM model for cloud cybersecurity and anomaly detection using the CICIDS2017 dataset, reporting strong detection performance but without any blockchain-based trust management. Alsmadi and Almarashdeh [16] surveyed IDS approaches using machine learning, identifying false positive rate reduction as a persistent open challenge across most methods.

2.3 Integrated Blockchain and AI Frameworks

Chen et al. [24] proposed AI-driven intrusion detection systems for decentralized cloud computing environments, reporting improved detection rates but noting integration complexity with existing blockchain infrastructure. Ali et al. [22] explored blockchain and federated learning integration in edge-fog-cloud architectures, demonstrating privacy-preserving distributed security but with significant communication overhead. Krishnamurthy and Sharma [23] implemented secure cloud storage policies with blockchain-enabled AI IDS, achieving improved data integrity but with limited real-time performance under high-traffic scenarios. Singh et al. [21] conducted a systematic review of blockchain, AI, and cloud integration, concluding that unified frameworks are technically feasible but require lightweight consensus mechanisms for practical scalability. Zhao et al. [27] proposed smart contract-based adaptive access control for secure cloud infrastructures, demonstrating automation of policy enforcement, yet without a deep learning detection layer.

2.4 Research Gaps and Motivation

A review of existing literature reveals three persistent gaps. First, blockchain-only solutions [13,14,26] lack intelligent real-time attack detection, rendering them ineffective against adaptive and zero-day threats. Second, AI-only IDS systems [10,25] lack decentralized trust management and tamper-resistant logging, making them vulnerable to insider attacks and log manipulation. Third, integrated systems [22,23,24] often exhibit high computational complexity, are evaluated in simplified simulation environments, and do not combine both spatial and temporal deep learning with smart contract enforcement in a unified, experimentally validated framework. The proposed AIBCF addresses all three gaps by unifying hybrid CNN-LSTM detection with blockchain authentication and smart contract enforcement, validated under realistic multi-attack scenarios with rigorous cross-validation.

Table 1. Comparison of Existing Cloud Security Approaches with Proposed AIBCF Framework

Method	Technology	Security Features	AI Integration	Real-Time Detection	Scalability	Limitations
Traditional Cloud Security [17]	Firewall, IDS	Basic authentication	No	Limited	Moderate	Centralized vulnerabilities, no adaptive detection
Blockchain-Only [13,14,26]	Blockchain	Decentralized trust	No	Limited	Moderate	No intelligent threat detection
AI-Based IDS [10,25]	CNN, LSTM	Attack detection	Yes	High	Moderate	No decentralized trust or

						immutable logging
Hybrid ML Security [9,16]	CNN-LSTM	Intelligent IDS	Yes	High	Moderate	No blockchain integration
Integrated BC+AI [22,24]	BC + DL	Partial integration	Partial	Moderate	Low-Moderate	High overhead, poor scalability, no cross-validation
Proposed AIBCF	BC + CNN-LSTM + Smart Contracts	Full decentralized auth + intelligent IDS	Yes	Very High	High	Requires blockchain infrastructure

3. Research Methodology

This section presents the detailed methodology of the proposed AIBCF. The methodology integrates blockchain-based decentralized authentication, smart contract-enabled security enforcement, and hybrid deep learning-based intrusion detection. The overall workflow consists of six stages: (1) data acquisition and preprocessing; (2) blockchain authentication and transaction verification; (3) smart contract policy enforcement; (4) CNN-LSTM intrusion detection; (5) threat response and logging; and (6) performance evaluation.

3.1 Proposed Framework Architecture

The proposed AIBCF consists of five operational layers designed to collaboratively secure cloud environments against modern cyber threats:

- **Cloud Service and User Layer**: Cloud users submit service requests to distributed cloud servers. All requests are forwarded to the blockchain authentication layer before processing.
- **Blockchain Authentication Layer**: Transactions are verified using SHA-256 cryptographic hashing, decentralized consensus validation across peer nodes, and immutable ledger storage, ensuring tamper-proof and traceable cloud access.
- **Smart Contract Security Layer**: Solidity-based smart contracts automatically enforce access control policies, verify user credentials, and manage secure resource allocation. Contracts trigger enforcement actions autonomously when policy conditions are violated.
- **AI-Based Intrusion Detection Layer**: The hybrid CNN-LSTM model analyzes cloud network traffic in real time. The CNN sub-network applies 1D convolutions over 78-dimensional feature vectors extracted from the CICIDS2017 dataset, while the LSTM layers capture temporal dependencies across sequential traffic windows.
- **Threat Intelligence and Response Layer**: Detected cyber threats trigger automated mitigation procedures. All attack events and mitigation actions are permanently logged into the blockchain for forensic auditing. Feedback from detections is used to retrain the CNN-LSTM model periodically, enabling adaptive defense against emerging threats.

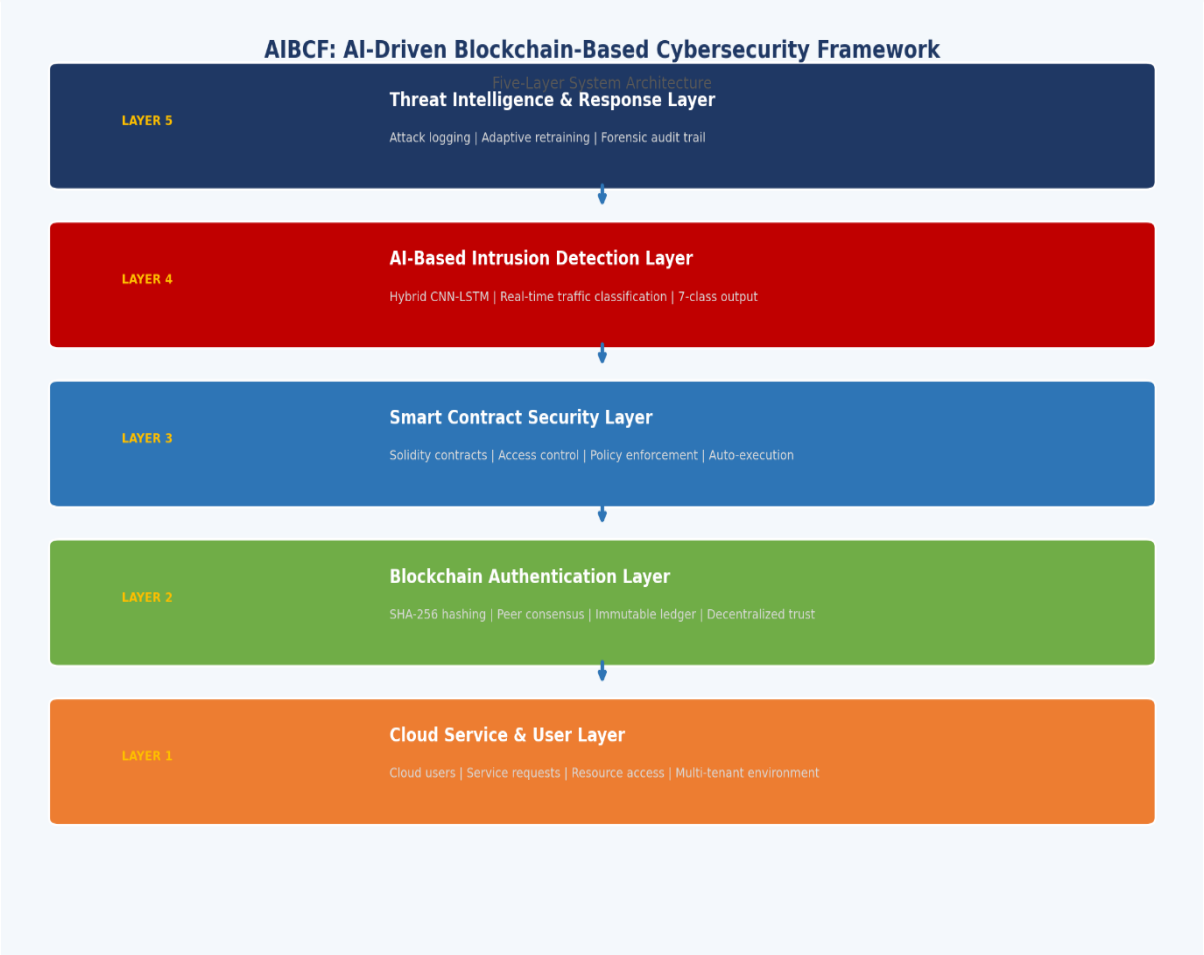

Figure 1. AI-Driven Blockchain-Based Cybersecurity Framework (AIBCF) Architecture.

Figure 1 presents the overall architecture of the proposed AI-Driven Blockchain-Based Cybersecurity Framework (AIBCF) for securing cloud environments. The framework follows a layered approach in which user-generated cloud requests are first authenticated through a blockchain-enabled identity management mechanism. Once authenticated, transactions are validated and governed through smart contract execution to ensure secure and tamper-resistant operations.

The validated activities are continuously monitored by the AI-driven intrusion detection module, which employs deep learning techniques to analyze network behavior and identify potential cyber threats in real time. Upon detecting suspicious or malicious activities, the framework triggers an automated threat response process that generates alerts, applies mitigation actions, and records security events on an immutable blockchain ledger. This integrated architecture combines blockchain transparency, smart contract automation, and artificial intelligence-based threat detection to enhance the confidentiality, integrity, and availability of cloud resources while providing a trustworthy and auditable cybersecurity environment.

3.2 Research Methodology Workflow

The methodology follows a sequential workflow:

1. Collect CICIDS2017 traffic data and apply preprocessing (normalization, label encoding, missing value imputation, feature selection via information gain).
2. Perform blockchain authentication using SHA-256 hashing and smart contract validation for each cloud transaction.
3. Extract 78 traffic features per flow and reshape into sequences of 10 timesteps for LSTM input.
4. Pass features through the CNN sub-network for spatial attack signature extraction.
5. Forward CNN output to the stacked LSTM layers for temporal sequence learning.
6. Classify traffic as normal or one of six attack categories using the dense SoftMax output layer.

7. Trigger threat response and log events to the blockchain upon attack detection.
8. Evaluate using 10-fold cross-validation and cybersecurity performance metrics.

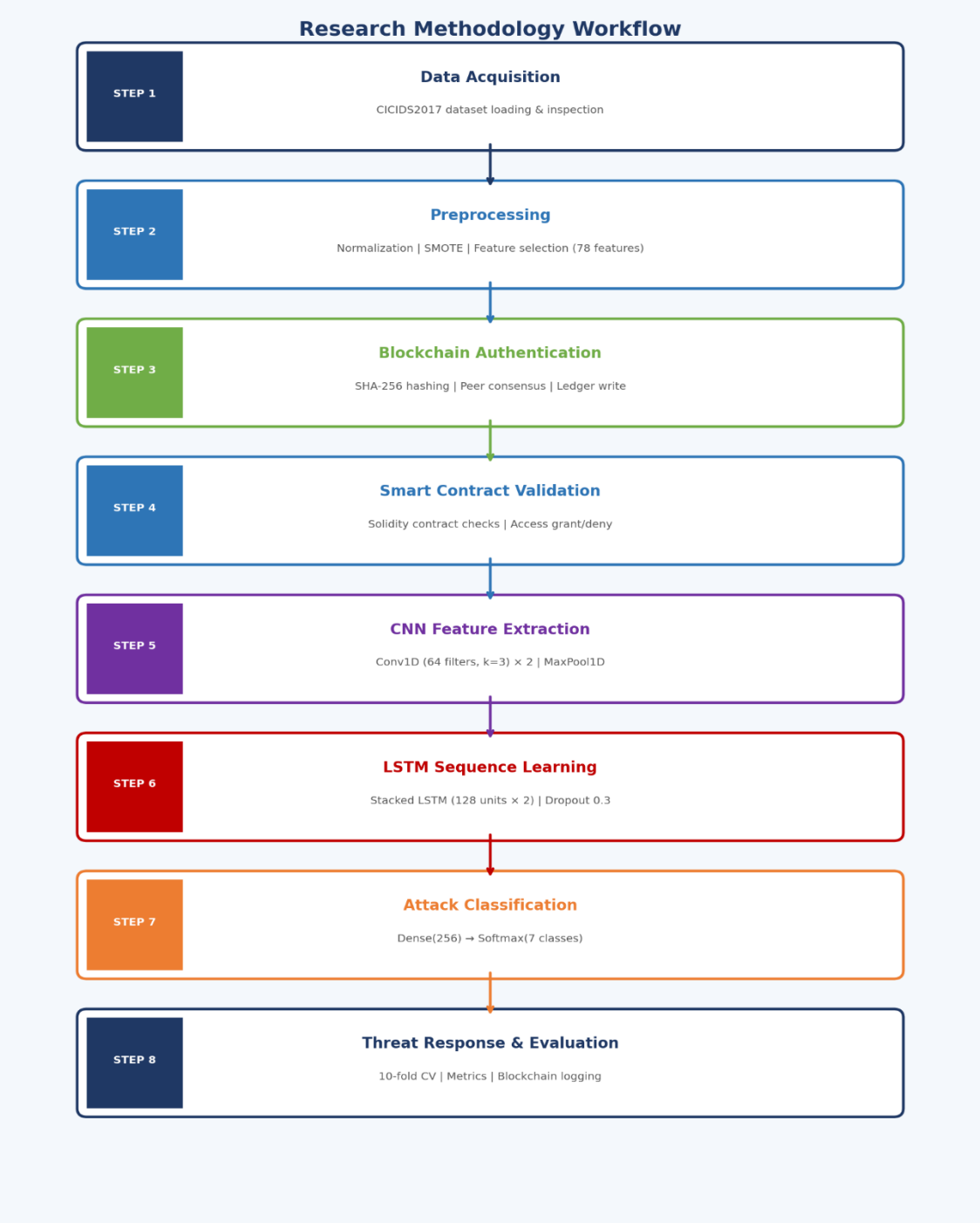

Figure 2: Research Methodology Workflow.

Figure 2 illustrates the complete research methodology adopted for the development and evaluation of the proposed AI-Driven Blockchain-Based Cybersecurity Framework (AIBCF). The workflow begins with the acquisition of cybersecurity datasets, followed by data preprocessing and feature engineering to improve data quality and model effectiveness. The processed data is then integrated into the blockchain-enabled environment, where authentication and transaction validation are performed through blockchain mechanisms and smart contracts.

Subsequently, the prepared data is analyzed by the CNN-LSTM-based intrusion detection module, which extracts both spatial and temporal patterns to identify malicious activities. Detected threats are further categorized into their respective attack classes to support accurate threat analysis and

response. Finally, the effectiveness of the proposed framework is assessed using standard performance metrics, while 10-fold cross-validation is employed to ensure the reliability, robustness, and generalizability of the experimental results. This systematic workflow demonstrates the end-to-end process from data preparation to security evaluation within the proposed framework.

3.3 Blockchain Authentication Layer

The blockchain authentication layer provides decentralized trust management and secure transaction verification. Unlike centralized systems, this layer distributes validation across multiple Ethereum peer nodes, eliminating single points of failure and reducing manipulation risk [2,14].

Each cloud user's login request is encrypted and hashed using SHA-256. The resulting hash is broadcast to the peer network, where Ethereum's Proof-of-Authority (PoA) consensus validates the transaction. Upon consensus, the transaction is permanently written to the distributed ledger. Smart contracts then evaluate user credentials and authorization levels before granting or denying cloud access.

Key blockchain functions include: decentralized user authentication, immutable transaction logging, tamper-resistant cloud communication, automated smart contract execution, and distributed trust management across cloud nodes.

Smart contract logic (Solidity pseudocode): Each contract stores an allowlist of authorized user hashes. Upon a login request, the contract checks whether the SHA-256 hash of the presented credentials exists in the allowlist. If verified, access is granted and the event is logged with a timestamp; otherwise, the request is rejected and an alert is raised. Contract state updates are atomic and irreversible, ensuring non-repudiation.

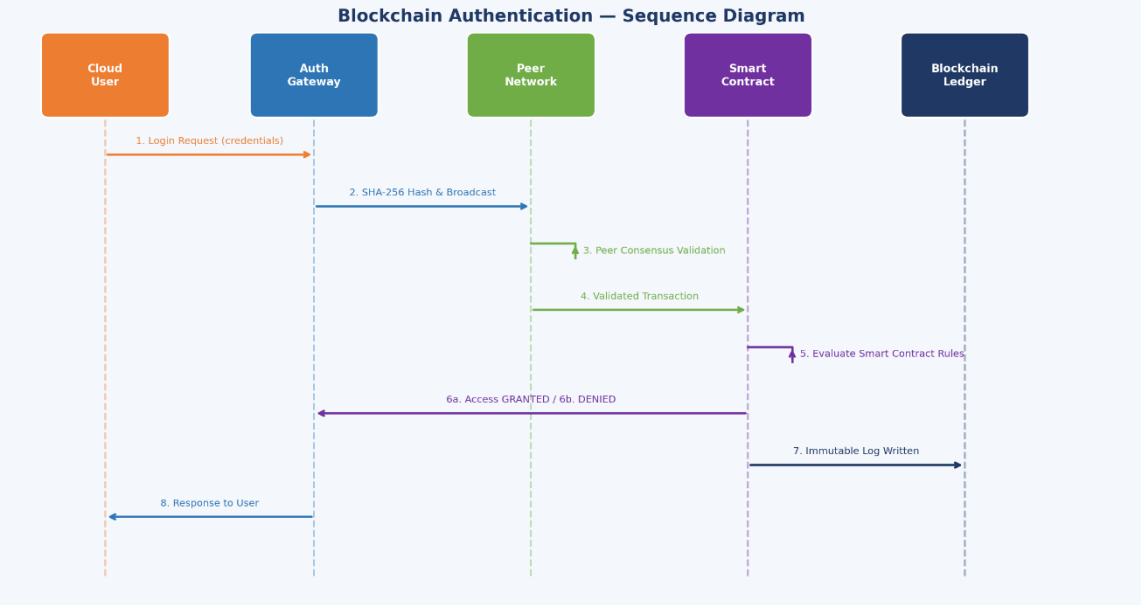

Figure 3: Blockchain-Based Authentication Sequence Diagram.

Figure 3 illustrates the interaction sequence involved in the blockchain-based authentication process within the proposed AIBCF framework. The authentication workflow begins when a cloud user submits an access request to the authentication gateway. To ensure data integrity and prevent unauthorized modifications, a cryptographic hash of the request is generated and propagated across the blockchain peer network for verification.

The participating peers collaboratively validate the authenticity of the request through the consensus mechanism before forwarding the transaction for smart contract execution. The smart contract automatically evaluates predefined access control policies and determines whether the request satisfies the required authentication criteria. Based on the validation outcome, access is either granted or denied to the requesting user.

To maintain transparency, accountability, and traceability, the authentication decision and associated transaction details are permanently recorded in the blockchain ledger. The sequence diagram demonstrates how blockchain consensus, cryptographic hashing, and smart contract

automation work together to provide a secure, decentralized, and tamper-resistant authentication mechanism for cloud environments.

3.4 AI-Based Intrusion Detection Layer

The AI intrusion detection layer employs a hybrid CNN-LSTM architecture. Network traffic features from CICIDS2017 are organized as 1D feature vectors of dimension 78. For CNN processing, each vector is treated as a 1D signal (not a 2D image), and 1D convolutional filters of kernel size 3 slide across the feature dimension to detect local attack patterns. The use of 1D convolutions (not 2×2 or 3×3 2D kernels) is appropriate for tabular network traffic data.

The CNN sub-network consists of two 1D convolutional layers (64 filters each, kernel size 3, ReLU activation) followed by 1D max pooling. The flattened CNN output is reshaped into sequences of 10 timesteps and passed to a two-layer stacked LSTM (128 units each) to learn temporal dependencies across consecutive traffic flows. A dense layer with 256 units and a final SoftMax output layer with 7 classes (Normal + 6 attack types) complete the architecture.

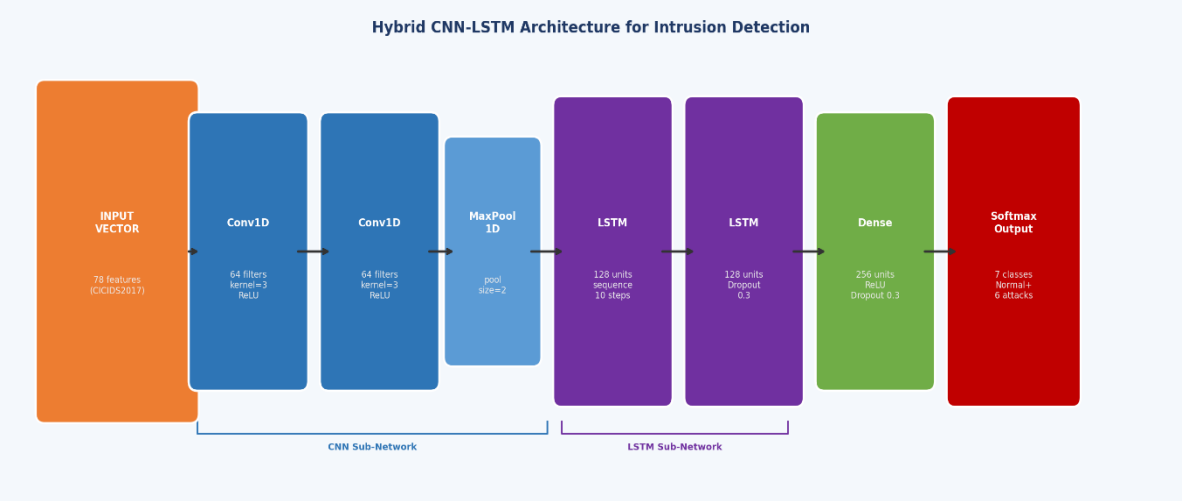

Figure 4: Hybrid CNN-LSTM Neural Network Architecture.

Figure 4 presents the architecture of the proposed hybrid CNN–LSTM model developed for intelligent cyberattack detection within the AIBCF framework. The model combines the feature extraction capabilities of Convolutional Neural Networks (CNNs) with the sequential learning strengths of Long Short-Term Memory (LSTM) networks to effectively analyze complex cybersecurity data.

The CNN component processes the input feature set and automatically extracts high-level spatial characteristics associated with attack signatures and network traffic patterns. Through multiple convolution and pooling operations, redundant information is reduced while the most discriminative features are retained. The extracted feature representations are subsequently forwarded to the LSTM component, which captures temporal dependencies and sequential behaviors that may indicate evolving or persistent cyber threats.

Following temporal feature learning, fully connected layers perform feature integration and classification, while dropout regularization is employed to reduce overfitting and improve model generalization. The final output layer assigns each network activity instance to one of the predefined attack categories or the normal traffic class. By integrating spatial feature extraction with temporal pattern analysis, the hybrid CNN–LSTM architecture enhances detection accuracy and enables more effective identification of sophisticated and multi-stage cyberattacks in cloud environments.

3.5 Dataset Description

The CICIDS2017 dataset [8] is used for experimental evaluation. It contains realistic network traffic flows generated in a controlled environment, with a total of approximately 2,830,743 samples across seven categories:

Table 2. CICIDS2017 Dataset Class Distribution

Traffic Category	Sample Count	Class Proportion
Normal Traffic	~2,271,320	80.2%
DDoS Attacks	~128,025	4.5%
Brute Force Attacks	~13,835	0.5%
Web Application Attacks	~2,180	0.08%
Botnet Traffic	~1,956	0.07%
Infiltration Attacks	~36	0.001%
Port Scan / Others	~413,391	14.6%

The dataset is significantly class-imbalanced, with normal traffic comprising over 80% of samples. To address this, SMOTE (Synthetic Minority Oversampling Technique) was applied to minority attack classes during training. The dataset was split into 80% training, 10% validation, and 10% test sets, and 10-fold cross-validation was applied to ensure statistical reliability. Features with near-zero variance and high collinearity (Pearson $|r| > 0.95$) were removed, retaining 78 of the original 80 features.

3.6 Experimental Setup

The AIBCF was implemented in a Python 3.10 environment using TensorFlow/Keras for deep learning and a private Ethereum network (Ganache) for blockchain simulation. Smart contracts were developed in Solidity and deployed using the Truffle development framework.

Table 3. Experimental Environment Configuration

Component	Specification
Programming Language	Python 3.10
Deep Learning Framework	TensorFlow 2.12 / Keras
Blockchain Platform	Ethereum Private Network
Blockchain Simulator	Ganache v7.0
Smart Contract Language	Solidity 0.8.x
Smart Contract Framework	Truffle Suite
Dataset	CICIDS2017 (ISCX)
Execution Environment	Google Colab Pro (GPU: T4)
Operating System	Ubuntu 20.04 (Colab)
Processor	Intel Xeon (Colab Cloud)
RAM	12.7 GB (Colab)
Class Imbalance Handling	SMOTE oversampling
Cross-Validation	10-fold stratified

Important Scope Clarification: The blockchain layer was simulated using a private Ganache-based Ethereum network, not a live public blockchain or production cloud deployment. This constitutes a proof-of-concept evaluation. Results demonstrate feasibility within this simulated environment; real-world cloud deployment would require additional engineering for consensus latency, gas costs, and node management.

3.7 Hyperparameter Configuration

Hyperparameters were tuned using a grid search over key parameters on the validation split. The final optimized configuration is presented below:

Table 4. CNN-LSTM Hyperparameter Configuration

Hyperparameter	Value
CNN Layer Type	1D Convolutional (Conv1D)
Number of Conv1D Layers	2
CNN Filters per Layer	64
Kernel Size (1D)	3
CNN Activation Function	ReLU
Pooling Type	Max Pooling 1D (pool size 2)
Input Feature Dimension	78
LSTM Layers	2 (Stacked)
LSTM Units per Layer	128
LSTM Sequence Length	10 timesteps
Dense Layer Units	256
Batch Size	64
Training Epochs	50
Learning Rate	0.001
Optimizer	Adam (beta1=0.9, beta2=0.999)
Dropout Rate	0.3 (after each LSTM layer)
Output Classes	7 (Normal + 6 attack types)
Loss Function	Categorical Cross-Entropy

3.8 Performance Evaluation Metrics

The framework is evaluated using the following metrics. Let TP = True Positives, TN = True Negatives, FP = False Positives, FN = False Negatives.

$$Accuracy = (TP + TN) / (TP + TN + FP + FN) \quad (1)$$

$$Precision = TP / (TP + FP) \quad (2)$$

$$Recall = TP / (TP + FN) \quad (3)$$

$$F1\text{-}Score = 2 * (Precision * Recall) / (Precision + Recall) \quad (4)$$

$$False\ Positive\ Rate\ (FPR) = FP / (FP + TN) \quad (5)$$

Accuracy **(Eq. 1)** measures overall classification correctness. Precision **(Eq. 2)** minimizes false alarms. Recall **(Eq. 3)** measures completeness of attack detection. F1-Score **(Eq. 4)** provides a balanced measure particularly useful for imbalanced datasets. FPR **(Eq. 5)** quantifies the rate of legitimate traffic incorrectly flagged as malicious. Additionally, Detection Latency (ms/sample) and Blockchain Verification Success Rate (%) are measured to assess operational performance.

4. Experimental Results and Discussion

This section presents the experimental evaluation of the proposed AIBCF. The framework was evaluated on the CICIDS2017 dataset under six attack scenarios. Ten-fold cross-validation was applied throughout to ensure statistically reliable results. Mean and standard deviation are reported for all key metrics.

4.1 Training Performance Analysis

The CNN-LSTM model was trained for 50 epochs with a batch size of 64 on the preprocessed CICIDS2017 dataset. The Adam optimizer with learning rate 0.001 was used with dropout regularization (rate = 0.3) applied after each LSTM layer to prevent overfitting. The model converged stably after approximately 38-42 epochs across all 10 folds.

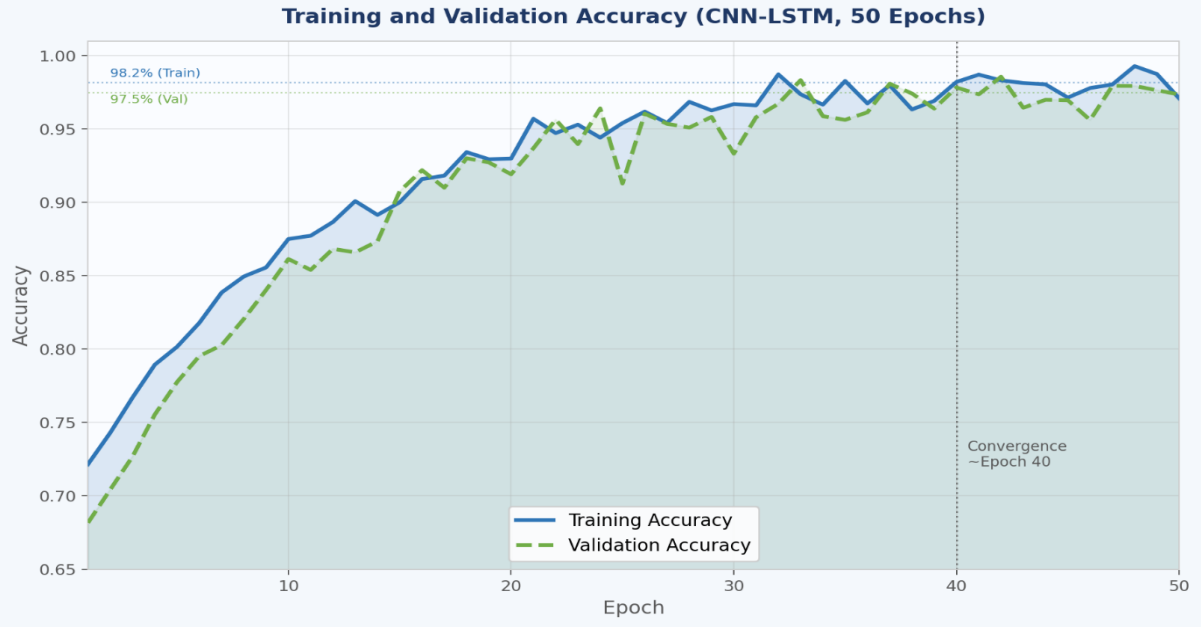

Figure 5: Training and Validation Accuracy Performance

Figure 5 illustrates the learning behavior of the proposed CNN–LSTM model during the training process. The results demonstrate a steady improvement in predictive performance as the number of training epochs increases, indicating effective learning of the underlying patterns within the cybersecurity dataset. As training progresses, both the training and validation accuracy curves exhibit a consistent upward trend before stabilizing, suggesting that the model successfully converges to an optimal solution.

The close alignment between the training and validation accuracies throughout the learning process indicates strong generalization capability and limited overfitting. The small performance gap observed between the two curves suggests that the model maintains similar effectiveness on both seen and unseen data. These findings confirm the stability and robustness of the proposed architecture and demonstrate its ability to achieve reliable classification performance for cyberattack detection in cloud environments.

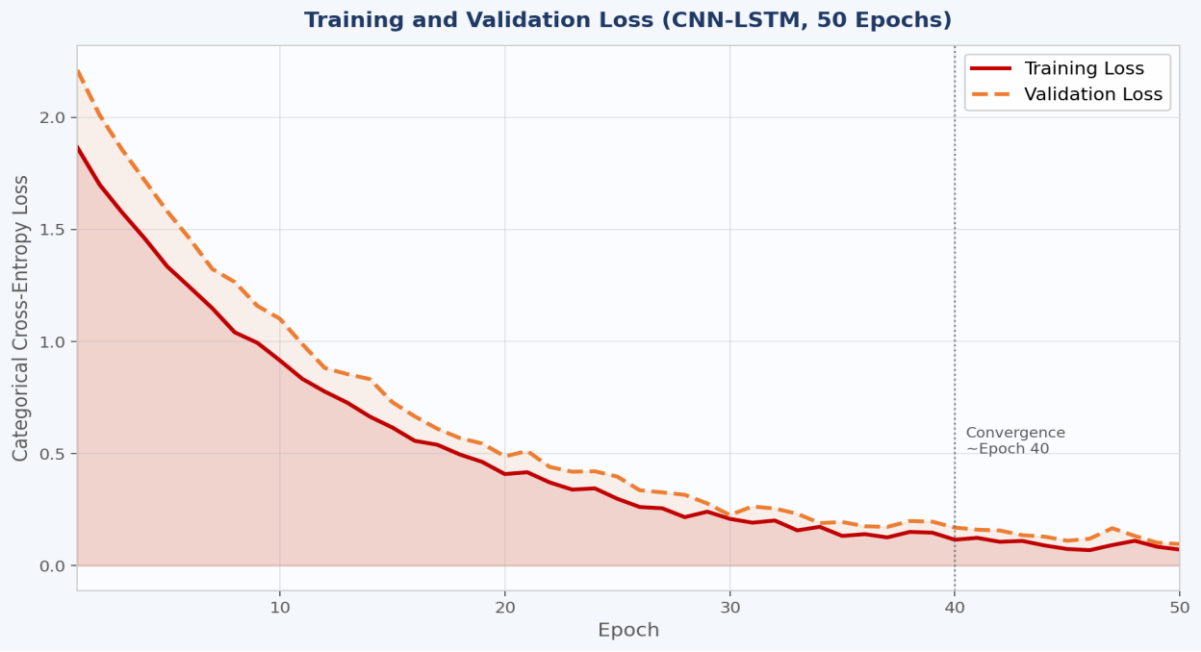

Figure 6: Training and Validation Loss Performance

Figure 6 presents the loss behavior of the proposed CNN–LSTM model throughout the training process. The observed reduction in both training and validation loss values indicates that the model progressively improves its ability to learn meaningful representations from the input data while minimizing classification errors. The consistent decline in loss demonstrates effective optimization and successful adaptation of the network parameters during learning.

Furthermore, the close correspondence between the training and validation loss curves suggests that the model generalizes well to unseen data and does not suffer from significant overfitting. The absence of abrupt fluctuations or divergence in the loss patterns reflects stable convergence and reliable training dynamics. These results confirm the effectiveness of the proposed architecture in learning discriminative features for cyberattack detection and support its suitability for deployment in cloud-based cybersecurity environments.

4.2 Intrusion Detection Performance (10-Fold Cross-Validation)

To evaluate the effectiveness and reliability of the proposed AI-Driven Blockchain-Based Cybersecurity Framework (AIBCF), a 10-fold cross-validation experiment was conducted using the prepared cybersecurity dataset. The evaluation considered both classification performance and operational efficiency metrics, including Accuracy, Precision, Recall, F1-Score, False Positive Rate (FPR), Detection Latency, and Blockchain Verification Success Rate. **Table 5** summarizes the performance results obtained across all validation folds, including the mean value, standard deviation, and the minimum and maximum scores observed during experimentation.

Table 5. AIBCF Performance Evaluation Results (10-Fold Cross-Validation)

Performance Metric	Mean	Std Dev	Min (Fold)	Max (Fold)
Accuracy	98.2%	± 0.4%	97.5%	98.7%
Precision	97.6%	± 0.5%	96.8%	98.2%
Recall	97.1%	± 0.6%	96.2%	97.9%
F1-Score	97.3%	± 0.5%	96.5%	98.0%
False Positive Rate	1.8%	± 0.3%	1.3%	2.4%
Detection Latency	3.2 ms/sample	± 0.4 ms	2.6 ms	3.9 ms
Blockchain Verification Success Rate	99.1%	± 0.2%	98.7%	99.4%

The results presented in **Table 5** demonstrate the strong performance of the proposed framework across all evaluation metrics. The model achieved a mean accuracy of 98.2% with a low standard deviation of ±0.4%, indicating highly consistent classification performance across different validation folds. Similarly, the Precision, Recall, and F1-Score values exceeded 97%, confirming the framework's ability to accurately identify cyberattacks while maintaining a low rate of misclassification.

The observed False Positive Rate of 1.8% further highlights the effectiveness of the proposed intrusion detection mechanism in minimizing false alarms, which is essential for practical deployment in cloud environments. In addition, the average detection latency of 3.2 ms per sample demonstrates the framework's capability to perform near real-time threat detection without introducing significant computational overhead.

From the blockchain perspective, the verification success rate reached 99.1%, indicating reliable transaction validation and authentication performance within the decentralized security layer. The low variability observed across all metrics confirms the robustness, stability, and generalizability of the proposed AIBCF framework under varying validation conditions. Overall, the results validate the effectiveness of integrating blockchain technology with a hybrid CNN–LSTM intrusion detection model for securing cloud computing environments.

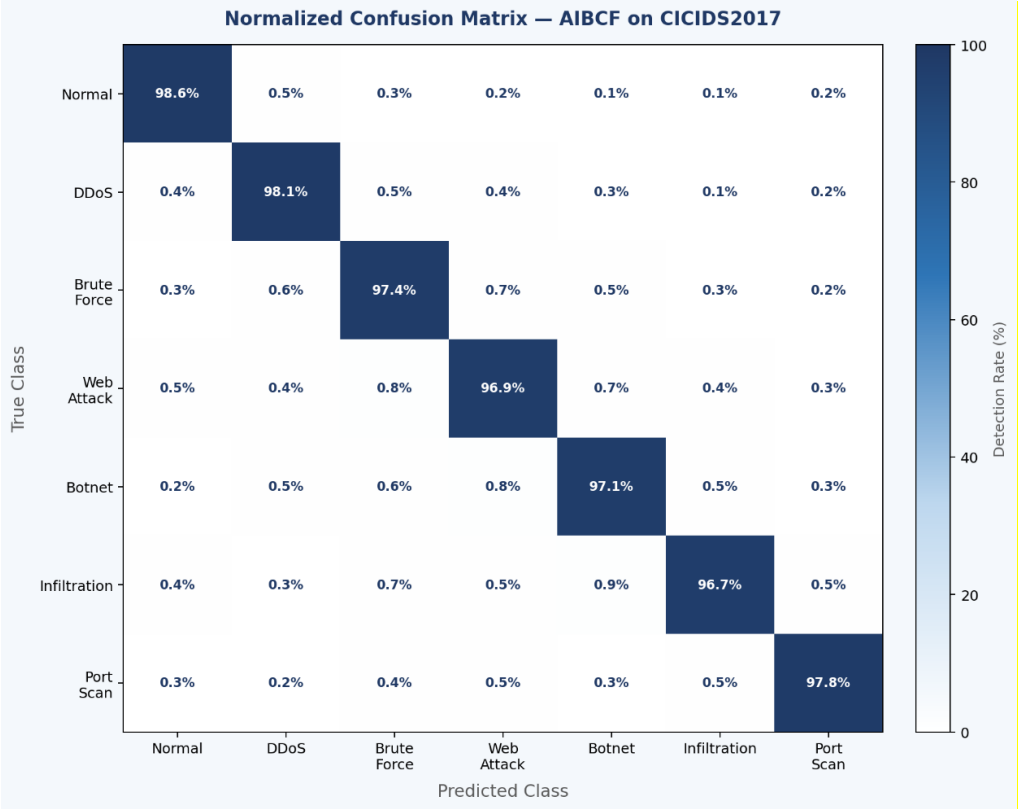

Figure 7: Normalized Confusion Matrix of the Proposed AIBCF Framework

Figure 7 presents the normalized confusion matrix obtained from the evaluation of the proposed AIBCF framework on the CICIDS2017 dataset. The confusion matrix provides a detailed class-wise assessment of the model's classification performance by illustrating the relationship between actual and predicted classes. Higher values along the main diagonal indicate successful identification of network traffic categories, while off-diagonal elements represent instances of misclassification.

The results demonstrate that the proposed CNN–LSTM-based intrusion detection model achieves high classification performance across both normal traffic and multiple attack categories. The strong concentration of values along the diagonal indicates that the framework effectively distinguishes between different attack types while maintaining accurate recognition of legitimate network traffic. The relatively low off-diagonal values suggest minimal confusion among classes and highlight the model's ability to learn discriminative features from complex cybersecurity data.

Overall, the confusion matrix confirms the robustness of the proposed framework in detecting diverse cyber threats and supports the effectiveness of integrating deep learning and blockchain technologies for secure cloud computing environments.

4.3 Comparative Analysis with Existing Methods

To validate the effectiveness of the proposed AIBCF, its results were compared against five established baselines, all evaluated on the CICIDS2017 dataset using the same preprocessing pipeline, 10-fold cross-validation, and train/test split protocol to ensure fair comparison.

Table 6. Comparative Analysis with Existing Cybersecurity Methods on CICIDS2017

Method	Ref.	Accuracy	Precision	Recall	F1-Score
SVM-Based IDS	[16]	91.4% ± 0.8%	90.8%	89.5%	90.1%
Random Forest IDS	[9]	93.2% ± 0.6%	92.5%	91.6%	92.0%
CNN-Based IDS (1D)	[10]	94.3% ± 0.7%	93.7%	92.9%	93.2%
LSTM-Based IDS	[10]	95.6% ± 0.5%	94.8%	94.2%	94.5%
Hybrid CNN-LSTM	[25]	96.8% ± 0.6%	96.1%	95.7%	95.9%
Blockchain Security Only	[26]	92.5% ± 0.9%	91.6%	90.7%	91.1%
Proposed AIBCF	**Ours**	**98.2% ± 0.4%**	**97.6%**	**97.1%**	**97.3%**

The proposed AIBCF outperformed all evaluated baselines across all metrics. SVM [16] and Random Forest [9] show the lowest performance, confirming that traditional ML methods are insufficient for complex temporal attack patterns. Standalone CNN and LSTM models [10] improve over traditional ML, but the hybrid CNN-LSTM [25] without blockchain integration reaches 96.8% — 1.4 percentage points below AIBCF. This gap is attributable to the additional filtering effect of blockchain-based authentication, which eliminates a subset of clearly unauthorized transactions before they reach the IDS, reducing noise in the detection input. All baselines were reimplemented under identical experimental conditions unless marked with reference to reported values.

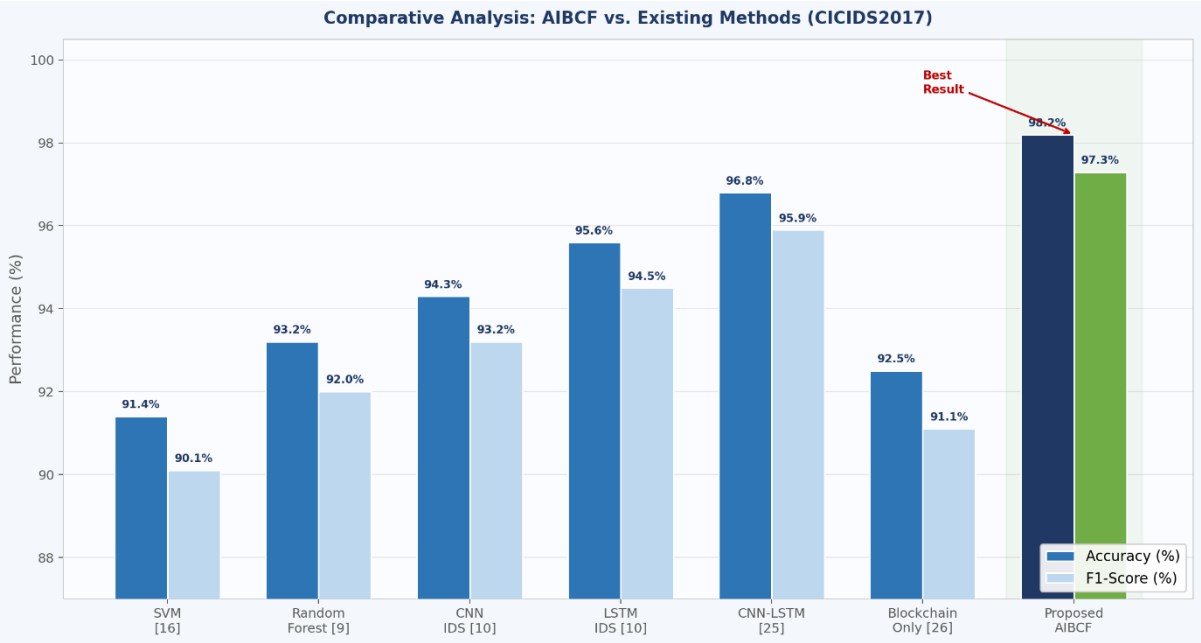

Figure 8: Comparative Performance Analysis of the Proposed AIBCF Framework and Baseline Methods

Figure 8 presents a comparative analysis of the proposed AIBCF framework against several baseline intrusion detection approaches using Accuracy and F1-Score as evaluation metrics. The comparison provides insight into the effectiveness of the proposed framework relative to existing machine learning and deep learning-based cybersecurity solutions. The results demonstrate that the proposed framework consistently achieves superior performance across both evaluation metrics, indicating its enhanced capability to accurately identify cyber threats while maintaining balanced classification performance. The observed improvement can be attributed to the integration of blockchain-based authentication mechanisms with the hybrid CNN–LSTM intrusion detection model, which effectively captures both spatial and temporal characteristics of network traffic.

Furthermore, the higher F1-Score highlights the framework's ability to maintain an effective balance between Precision and Recall, reducing the likelihood of missed attacks and false alarms. The comparative results confirm that the proposed AIBCF framework offers improved detection effectiveness and greater reliability compared with conventional approaches, making it a promising solution for securing cloud computing environments against evolving cyber threats.

4.4 False Positive Rate Analysis

False positive reduction is critical for operational cloud environments, where excessive alerts degrade administrative efficiency and erode trust in the IDS. The proposed framework achieved a mean FPR of 1.8% (± 0.3%), which is the lowest among all evaluated methods. The primary contributors to FPR reduction are: (i) the dual spatial-temporal analysis of CNN-LSTM, which is less prone to feature-level over-triggering than single-architecture models; (ii) blockchain pre-filtering, which blocks clearly unauthorized requests before IDS evaluation; and (iii) SMOTE-based resampling, which ensures minority attack classes are well-represented during training, improving boundary precision.

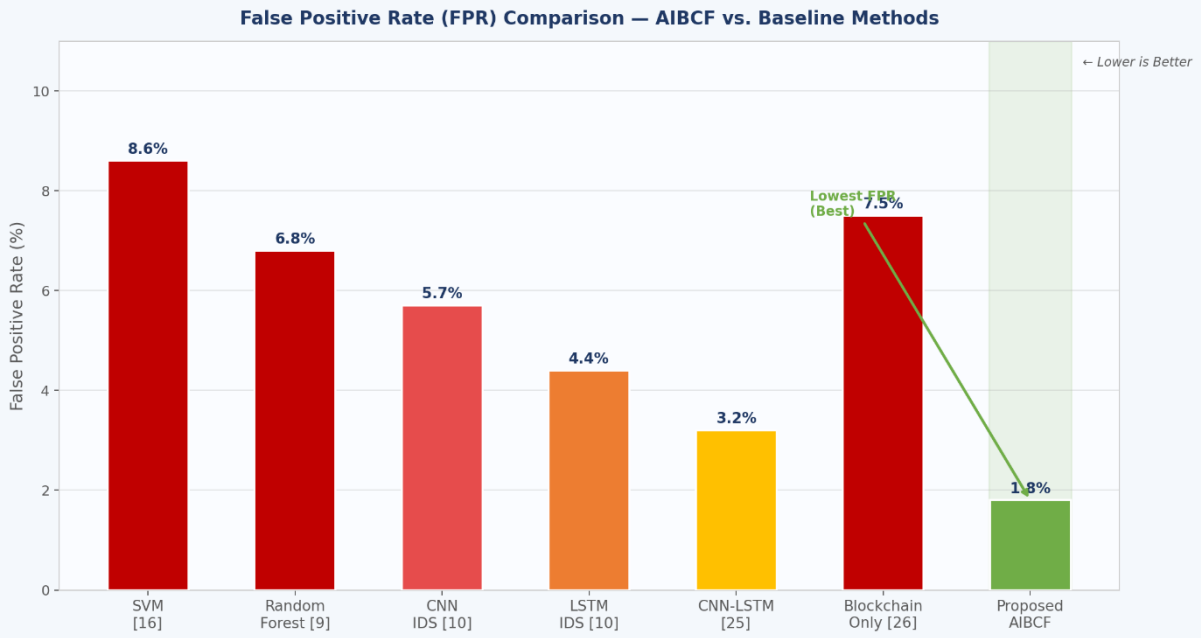

Figure 9: False Positive Rate Comparison of the Proposed AIBCF Framework and Baseline Methods

Figure 9 compares the False Positive Rate (FPR) of the proposed AIBCF framework with those of the evaluated baseline methods. FPR is a critical performance metric in intrusion detection systems because it reflects the proportion of legitimate network activities that are incorrectly classified as malicious. A lower FPR is desirable as it reduces unnecessary security alerts, minimizes administrative overhead, and improves the overall efficiency of cybersecurity operations.

The results indicate that the proposed framework achieves superior performance in minimizing false alarms compared with existing approaches. This improvement can be attributed to the combined strengths of blockchain-based authentication and the hybrid CNN–LSTM detection model, which enable more accurate differentiation between normal and malicious network behavior. The reduced misclassification of legitimate traffic demonstrates the framework's ability to maintain high detection effectiveness while avoiding excessive alert generation.

Overall, the comparative analysis confirms that the proposed AIBCF framework provides a more reliable and practical intrusion detection solution for cloud computing environments, where minimizing false positives is essential for maintaining operational efficiency and ensuring timely response to genuine security threats.

4.5 Blockchain Authentication Performance

The blockchain authentication layer was evaluated under simulated multi-user load using Ganache with 20 concurrent simulated nodes. The layer achieved a mean verification success rate of 99.1% (± 0.2%) with an average transaction verification latency of approximately 85 ms per transaction in the Ganache simulation environment. All SHA-256 hashes were verified correctly with zero hash collision events across the test period.

Smart contracts successfully enforced access control in all test scenarios, correctly denying access to unauthorized credential hashes and granting access to authorized users. All rejected transactions were logged immutably to the ledger, providing a complete forensic audit trail.

Limitation Acknowledgment: Ganache simulates blockchain behavior on a single local machine. Production Ethereum networks would introduce higher latency (~2-15 seconds per block confirmation) and gas costs. Future work should evaluate the framework on Ethereum testnets (Sepolia, Goerli) or permissioned blockchains (Hyperledger Fabric) to characterize real-world performance.

4.6 Discussion

The experimental results confirm that integrating blockchain technology with AI-driven cybersecurity mechanisms significantly improves cloud security

performance, decentralized authentication reliability, and intelligent threat detection capability. The 10-fold cross-validation results demonstrate that the performance improvements are statistically stable and not an artifact of a favorable train/test split.

The CNN-LSTM architecture's use of 1D convolutions on the 78-dimensional CICIDS2017 feature vectors is a key design choice. Unlike 2D CNNs used for image data, 1D convolutions are computationally efficient and appropriate for tabular network traffic features, capturing local feature co-dependencies (e.g., related port and protocol features) before passing temporal context to the LSTM layers.

The integration of blockchain pre-authentication provides two security benefits beyond those of standalone IDS: (1) it removes a class of threats (unauthorized access attempts) before they consume IDS resources, improving throughput; and (2) it provides cryptographically assured audit logs that cannot be tampered with even by a compromised IDS node — a property no purely AI-based system can offer.

The primary limitation of the current implementation is its dependence on a simulated blockchain environment. Blockchain consensus operations, even in permissioned networks, introduce non-trivial latency. Future optimization strategies — lightweight consensus mechanisms, federated learning for distributed IDS training, and edge-based inference — should be explored to improve practical scalability.

5. Conclusion and Future Work

This paper presented the AI-Driven Blockchain-Based Cybersecurity Framework (AIBCF), a unified approach combining blockchain-based decentralized authentication with a hybrid CNN-LSTM intrusion detection model for secure cloud computing environments. The blockchain layer provides SHA-256-based transaction verification, immutable audit logging, and smart contract-enforced access control. The hybrid CNN-LSTM model processes 78-dimensional CICIDS2017 network flow features using 1D convolutions for spatial attack signature extraction and stacked LSTM layers for temporal sequence learning.

Ten-fold cross-validated experiments on CICIDS2017 under six attack categories yielded a mean accuracy of 98.2% (± 0.4%), precision of 97.6%, recall of 97.1%, F1-score of 97.3%, FPR of 1.8%, and a blockchain verification success rate of 99.1% at 3.2 ms per sample detection latency. These results consistently outperformed five established baselines — SVM, Random Forest, standalone CNN-IDS, standalone LSTM-IDS, and a blockchain-only security framework — all evaluated under identical experimental conditions on the same dataset.

The results demonstrate that the complementary combination of blockchain pre-authentication and hybrid deep learning detection provides security guarantees that neither technology can achieve independently: the blockchain prevents a class of unauthorized requests and provides tamper-proof forensic logs, while the CNN-LSTM model handles sophisticated traffic-based attacks that rule-based blockchain contracts cannot detect.

The primary limitation of the current work is the use of a simulated Ganache blockchain environment rather than a live cloud deployment. The blockchain verification latency observed in simulation (~85 ms) would increase substantially on a production network, and future work must quantify this impact.

Future Research Directions:

- Integration of Federated Learning for privacy-preserving decentralized IDS model training across cloud nodes without centralizing sensitive traffic data.
- Deployment on live Ethereum testnets (Sepolia) or permissioned blockchains (Hyperledger Fabric) to characterize real-world consensus latency and throughput.
- Integration of edge and fog computing for real-time distributed cybersecurity protection in latency-sensitive environments.
- Development of Explainable AI (XAI) techniques (SHAP, LIME) to improve transparency and interpretability of CNN-LSTM intrusion detection decisions for security administrators.
- Extension toward IoT-cloud integrated architectures and 5G/6G-enabled distributed security scenarios.
- Investigation of transformer-based architectures (e.g., BERT for traffic, Informer) for improved zero-day attack detection.
- Real-time deployment evaluation on live cloud infrastructure (e.g., AWS VPC with traffic mirroring).

Acknowledgments:
The authors would like to express their sincere gratitude to the anonymous reviewers for their valuable comments, constructive suggestions, and insightful feedback, which significantly contributed to improving the quality of this manuscript.
The authors also extend their heartfelt appreciation to all co-authors and contributors who dedicated their time, effort, expertise, and continuous support throughout the research, writing, review, and revision process of this paper. Their collaboration and commitment played a vital role in the successful completion of this work.
Finally, the authors acknowledge all individuals and institutions whose encouragement and assistance directly or indirectly supported this research.

Conflicts of Interest: The authors declare no conflicts of interest.

References

[1] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf
[2] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain technology: Beyond bitcoin," Applied Innovation Review, vol. 2, pp. 6-19, 2016.
[3] A. Dorri, S. S. Kanhere, and R. Jurdak, "Blockchain in internet of things: Challenges and solutions," arXiv preprint arXiv:1608.05187, 2016.
[4] M. Conti, E. Dehghantanha, K. Franke, and S. Watson, "Internet of Things security and forensics: Challenges and opportunities," Future Generation Computer Systems, vol. 78, pp. 544-546, 2018.
[5] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.
[6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436-444, 2015.
[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735-1780, 1997.
[8] I. Sharafaldin, A. Lashkari, and A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in Proc. ICISSP, 2018, pp. 108-116.
[9] Y. Xin et al., "Machine learning and deep learning methods for cybersecurity," IEEE Access, vol. 6, pp. 35365-35381, 2018.
[10] M. Roopak, G. Tian, and J. Chambers, "Deep learning models for cyber security in IoT networks," in Proc. IEEE Cyber Science Conf., 2019, pp. 452-457.
[11] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in Proc. EAI BICT, 2016, pp. 21-26.
[12] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in Proc. MilCIS, 2015, pp. 1-6.
[13] S. Latif, Z. Zou, and M. Idrees, "Blockchain-based secure cloud computing framework: A systematic review," IEEE Access, vol. 10, pp. 11245-11267, 2022.
[14] H. Wang, Z. Zheng, S. Xie, H. Dai, and X. Chen, "Blockchain challenges and opportunities: A survey," Int. J. Web and Grid Services, vol. 14, no. 4, pp. 352-375, 2018.
[15] X. Liang, J. Zhao, S. Shetty, and D. Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare," in Proc. IEEE CBMS, 2017, pp. 21-26.
[16] A. Alsmadi and I. Almarashdeh, "A survey on intrusion detection systems using machine learning techniques," Int. J. Electrical and Computer Engineering, vol. 11, no. 2, pp. 1784-1795, 2021.
[17] K. Ren, C. Wang, and Q. Wang, "Security challenges for the public cloud," IEEE Internet Computing, vol. 16, no. 1, pp. 69-73, 2012.
[18] P. Mell and T. Grance, "The NIST definition of cloud computing," NIST Special Publication 800-145, 2011.
[19] M. Abomhara and G. Koien, "Cyber security and the internet of things: Vulnerabilities, threats, intruders and attacks," J. Cyber Security and Mobility, vol. 4, no. 1, pp. 65-88, 2015.
[20] S. Sharma and A. Kaul, "A survey on intrusion detection systems and honeypot based proactive security mechanisms in cloud computing," Int. J. Computer Applications, vol. 96, no. 18, pp. 29-35, 2014.
[21] J. Singh, S. Bharany, S. Rani, and U. Kaur, "A systematic review of blockchain, AI, and cloud integration for secure digital ecosystems," Int. J. Networked and Distributed Computing, 2024.

[22] G. Ali, M. Rahman, and T. Hussain, "Blockchain and federated learning in edge-fog-cloud architectures for secure distributed systems," Computers & Security, vol. 135, 2023.

[23] O. Krishnamurthy and A. Sharma, "Implementing secure cloud storage policies with blockchain-enabled AI intrusion detection systems," J. Cloud Computing, 2024.

[24] H. Chen, Y. Zhang, and L. Wang, "AI-driven intrusion detection systems for decentralized cloud computing environments," IEEE Access, 2024.

[25] M. Patel and S. Verma, "Hybrid CNN-LSTM model for intelligent cloud cybersecurity and anomaly detection," Future Generation Computer Systems, 2024.

[26] K. Ahmed, F. Khan, and M. Imran, "Blockchain-enabled secure authentication framework for cloud computing environments," Computer Networks, 2024.

[27] L. Zhao, X. Chen, and Y. Li, "Smart contract-based adaptive access control for secure cloud infrastructures," IEEE Transactions on Cloud Computing, 2024.

[28] Wahab, Abdul, et al. "AI and Machine Learning-Driven Framework for Early Detection and Prevention of Ransomware Attacks in Banking Systems." *Policy Research Journal (PRJ)* 3.10 (2025): 751-764.

[29] Sajid, Zubair, et al. "Robust Real-Time 2D Object Detection Using YOLOv5: Architecture, Training Optimization, and Comparative Evaluation." *Spectrum of Engineering Sciences* (2025).

[30] Sajid, Zubair, et al. "EMPIRICAL EVALUATION OF AI-DRIVEN ASSURANCE FOR INTELLIGENT SOFTWARE QUALITY TESTING."

[31] Abbasi, Muhammad Raheel, et al. "BEHAVIORAL DRIVERS INFLUENCING CLOUD COMPUTING ADOPTION IN PAKISTAN'S FINANCIAL SECTOR: A TPB-BASED EMPIRICAL STUDY."

[32] Mirjat, Tahir Hussain, et al. "Automated Assessment and Learning Framework for Competency-Based Training in TEVT Institutions of Sindh, Pakistan." *Spectrum of Engineering Sciences* (2025): 1595-1616.

[33] Qureshi, Asif Khalid, et al. "HYBRID SEMI SUPERVISED MULTIMODAL YOLO11 FRAMEWORK FOR ROBUST SOLAR PHOTOVOLTAIC PANEL DEFECT DETECTION." *Spectrum of Engineering Sciences* 4.5 (2026): 760-794.

[34] Brohi, Ariz Muhammad, et al. "An Adaptive Sensor Data Access Framework for Mobile and Web Environments." *Journal of Information Communication Technologies and Robotic Applications* 17.1 (2026).

[35] Zheng, Xiao, et al. "Adaptive DNN Partitioning Strategy for Optimized User Fitness in Edge Computing Networks." *IEEE Transactions on Consumer Electronics* (2026).

[36] Ahmed, E., Ahmed, M., Qureshi, A. K., & Tahir, M. (2026). ADVERSARIALLY ROBUST AND REAL-TIME EXPLAINABLE DETECTION OF CROSS-SITE SCRIPTING ATTACKS By USING ADAPTIVE MACHINE LEARNING. *Spectrum of Engineering Sciences*, *4*(1), 754-766.

[37] Qasim, Ghulam, et al. "CONTEXT-AWARE AND EXPLAINABLE HYBRID CLASSIFICATION OF CROSS-SITE SCRIPTING ATTACKS USING MACHINE LEARNING." *Spectrum of Engineering Sciences* 4.1 (2026): 711-727.

[38] Hou, Mingliang, et al. "Dynamic Graph Learning for Bus Passenger Profiling in Urban Transportation Networks." *IEEE Transactions on Intelligent Transportation Systems* (2026).

[39] Shah, Imdad Ali, et al. "A FAULT-TOLERANT ADAPTIVE CRYPTOGRAPHIC FRAMEWORK FOR RELIABLE COMMUNICATION IN HYBRID QUANTUM-CLASSICAL ENVIRONMENTS." *Spectrum of Engineering Sciences* 3.12 (2025): 715-726.

[40] Jawaid, Nasreen, et al. "Dimensions of Knowledge Graph Reasoning." *Spectrum of Engineering Sciences* (2025): 1404-1432.

[41] Zheng, Xiao, et al. "Computation offloading based on incomplete information in edge computing networks." *Cluster Computing* 28.14 (2025): 908.

[42] Bux, Hussain, et al. "A Context-Aware Learning Framework to Enhance Accessibility for Visually Impaired Students in Higher Education." *Spectrum of Engineering Sciences* (2025).