AttentionChain: Self-Attention DL Model with Blockchain Logging for Intrusion Detection in IoMT

Subroto Kumar Ghosh, Mohtasin Golam, Sium Bin Noor, Jae-Min Lee, and Dong-Seong Kim
Networked Systems Laboratory, Department of IT Convergence Engineering,
Kumoh National Institute of Technology, Gumi, South Korea.
(subroto, gmoh248, siumbinmoor, ljmpaul, and dskim)@kumoh.ac.kr

***Abstract*—Internet of Medical Things (IoMT) devices require real-time intrusion detection systems that provide both high accuracy and verifiable audit trails for regulatory compliance. Traditional centralized approaches lack transparency and immutability guarantees. This paper proposes AttentionChain, a self-attention neural network integrated with blockchain for IoMT intrusion detection. AttentionChain employs a custom FeatureAttention layer that learns weighted representations of network traffic features, achieving 99.89% accuracy on the CIC-IoMT 2024 dataset (5.4M samples, 46 features, 19 attack classes) with real-time inference at 0.0233 ms per sample, and 0.017% false positive rate. Detected intrusions are automatically logged to Pure Chain, a permissioned blockchain using the Proof of Authority and Association (PoA^2) consensus mechanism [1], [2]. Pure Chain logging guarantees immutable and cryptographically verifiable audit trails. Comprehensive benchmarking on Pure Chain demonstrates 65.33 TPS throughput with 0.9 second latency, enabling practical deployment for healthcare environments. AttentionChain provides healthcare institutions with verifiable, transparent, and tamper-proof intrusion detection capabilities for securing critical IoMT networks.**

***Index Terms*—Deep Learning, IoMT, Intrusion Detection, Pure Chain, Self-Attention.**

I. Introduction

IoMT has revolutionized healthcare delivery by enabling remote patient monitoring, real-time clinical data collection, and automated diagnostic systems across hospital networks and connected medical infrastructures [3]. However, the expansion of interconnected medical devices has introduced new security vulnerabilities that directly threaten patient safety and data privacy [4], [5], [6]. IoMT devices operate in resource-constrained environments with limited computational capabilities, making them inherently susceptible to sophisticated network attacks including DDoS attacks, packet injection, unauthorized access, and malware propagation. Unlike traditional IT systems where security breaches result in data loss or financial damage, compromised IoMT devices can directly endanger patient lives and compromise critical medical operations [7], [8]. Healthcare organizations face strict regulatory compliance requirements including HIPAA, GDPR [9], and healthcare-specific standards that mandate comprehensive audit trails, forensic evidence retention, and immediate threat detection for all security incidents. The challenge of balancing real-time intrusion detection requirements with regulatory compliance and audit trail preservation remains a critical gap in healthcare cybersecurity [10].

Current intrusion detection systems for IoMT networks typically rely on signature-based methods, machine learning, or deep learning, each exhibiting significant limitations [11]. Signature-based systems are limited to detecting only previously known attack patterns, requiring frequent manual updates to address emerging threats, and often generating numerous false positives in the dynamic and complex healthcare environment [12]. Machine learning techniques improve upon this by enabling detection based on learned traffic patterns rather than rigid signatures [11]. Deep learning enhances detection capability by automatically extracting hierarchical and complex features from raw data, generally achieving higher accuracy [13]. However, deep learning models can be affected by irrelevant or noisy features that may distract the learning process, resulting in performance below expectations [11]. In addition, standalone deep learning systems introduce critical vulnerabilities. Detection decisions lack cryptographic verification, logged alerts can be modified or deleted in centralized databases without trace, audit trails are vulnerable to tampering by administrators or attackers [14]. It could result in no immutable forensic evidence for regulatory compliance or incident investigation. Existing systems cannot simultaneously satisfy the essential requirements like high-accuracy attack detection, real-time processing capability, and verifiable and tamper-proof event logging [15]. Recent IoMT intrusion detection and blockchain-based security studies primarily focus on centralized deep models or conceptual logging schemes that do not provide per-prediction, automatically generated audit trails on a deployed permissioned blockchain. This gap creates a fundamental weakness in healthcare IoMT security architecture where accurate threat detection cannot be verified or forensically validated.

To address these limitations, this paper introduces AttentionChain, a self-attention deep learning framework utilizing a FeatureAttention layer with immutable blockchain logging. This design enables the model to both detect new or emerging attacks in IoMT traffic. Self-attention mechanisms in deep learning enable the model to identify which specific traffic characteristics are most indicative of attacks. Unlike traditional deep learning models that treat all features equally, self-attention layers compute adaptive weights over input features. This targeted feature weighting enhances accuracy, robustness, and efficiency of intrusion detection in diverse IoMT settings, making FeatureAttention a crucial component in

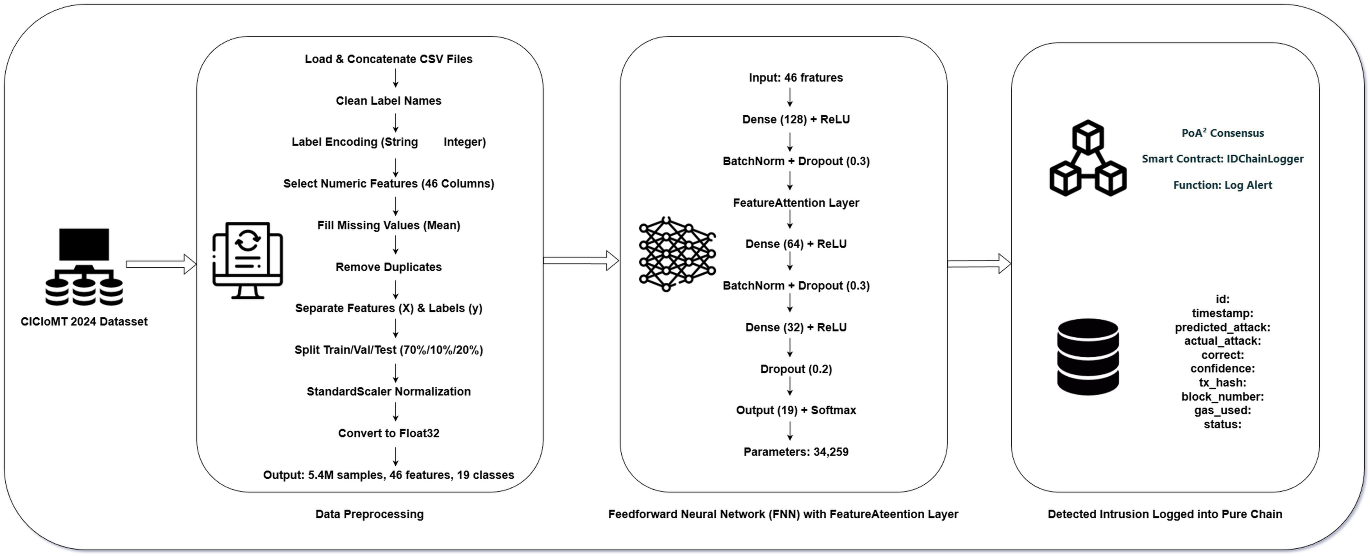

CICIoMT 2024 Dataset
Load & Concatenate CSV Files
Clean Label Names
Label Encoding (String Integer)
Select Numeric Features (46 Columns)
Fill Missing Values (Mean)
Remove Duplicates
Separate Features (X) & Labels (y)
Split Train/Val/Test (70%/10%/20%)
StandardScaler Normalization
Convert to Float32
Output: 5.4M samples, 46 features, 19 classes
Data Preprocessing
Input: 46 fratures
Dense (128) + ReLU
BatchNorm + Dropout (0.3)
FeatureAttention Layer
Dense (64) + ReLU
BatchNorm + Dropout (0.3)
Dense (32) + ReLU
Dropout (0.2)
Output (19) + Softmax
Parameters: 34,259
Feedforward Neural Network (FNN) with FeatureAteention Layer
PoA² Consensus
Smart Contract: IDChainLogger
Function: Log Alert
id:
timestamp:
predicted_attack:
actual_attack:
correct:
confidence:
tx_hash:
block_number:
gas_used:
status:
Detected Intrusion Logged into Pure Chain

Fig. 1: AttentionChain Framework Architecture

improving deep learning-based security solutions. Blockchain ensures verifiable and immutable event logging. Pure Chain is suitable for real-time healthcare applications because of its high throughput and low latency. By automatically logging self-attention deep learning predictions to Pure Chain, the framework achieves accurate and real-time intrusion detection, cryptographically verifiable audit trails, and transparent forensic analysis that enables rigorous investigation of security incidents. In contrast to existing approaches, AttentionChain combines a FeatureAttention-based self-attention architecture with an implemented Web3.py–based logging component and a smart contract on Pure Chain, so that every detection decision is immutably recorded on-chain during inference in real time.

The contributions of this work are as follows:

- Presented a self-attention deep learning framework, achieving 99.89% accuracy with real-time inference 0.0233 ms per sample, and 0.017% false positive rate.
- Developed FeatureAttention layer that computes adaptive importance weights over network traffic features, enabling transparent identification of which characteristics are most indicative of attacks.
- Demonstrated Pure Chain integration achieving cryptographically verifiable, tamper-proof intrusion logging at 65.33 transactions per second throughput, and 0.9 second latency.

II. Methodology

This section presents the complete methodology of the AttentionChain framework for blockchain-verifiable IoMT intrusion detection. The approach consists of four integrated stages: (1) comprehensive data preprocessing and normalization of network traffic features from the CIC-IoMT 2024 dataset [16], (2) design and training of the AttentionChain neural network incorporating a FeatureAttention mechanism for threat detection, (3) inference and performance evaluation across 1,081,536 test samples, and (4) automatic logging of detected intrusions to Pure Chain for cryptographically verifiable audit trail maintenance. Each stage is designed to address critical requirements: accurate attack classification, real-time processing capability, and tamper-proof forensic evidence retention. The following subsections detail the implementation of each stage.

A. Framework Overview

Figure. 1 illustrates the flow of data of the proposed AttentionChain framework. The framework integrates three key steps: (1) data preprocessing of IoMT network traffic, (2) self-attention deep learning for intrusion detection, and (3) blockchain integration for immutable audit trail logging. The framework processes raw network traffic features through feature normalization, trains an attention-based neural network to classify 19 attack types with feature importance weights, and automatically logs detected intrusions to Pure Chain with cryptographic verification.

B. Dataset and Data Preprocessing

The CIC-IoMT 2024 dataset [16] comprises network traffic from healthcare IoMT devices with 5,407,680 samples containing 46 numeric network flow features and 19 attack class labels. Attack type labels are cleaned by removing suffix markers (“_train”, “_test”) using regular expression matching: label = regex_match($r‘(.*[A-Za-z])(\underset{.}{+})?‘$). Labels are encoded to integers 0–18 using scikit-learn’s LabelEncoder, creating 19 discrete attack classes.

Numeric features are selected from the raw data, resulting in 46 features per sample. Missing values within numeric features are imputed with column-wise means to handle incomplete records. Duplicate rows are removed to eliminate redundant samples. Data is partitioned using stratified random sampling into training (70%, 3,785,376 samples), validation (10%, 540,768 samples), and test (20%, 1,081,536 samples) sets. Stratification ensures class distribution is maintained across all sets, preventing class imbalance issues.

Feature normalization is applied using StandardScaler:

$$\hat{X} = \frac{X - \mu}{\sigma} \tag{1}$$

where μ and σ are computed from training data only. The training scaler is then applied to validation and test data to prevent data leakage. All feature matrices are converted to float32 precision for efficient GPU computation during training and inference.

C. FeatureAttention Layer

The custom FeatureAttention layer enables adaptive feature weighting for intrusion detection by learning importance scores across all input features through a matrix-based attention mechanism. This allows the model to focus on the most relevant network traffic characteristics when distinguishing between benign and different types of attack.

Given input features $x \in \mathbb{R}^{B\times128}$, where B is the batch size and 128 is the number of input features, the FeatureAttention layer applies a trainable linear transformation parameterized by a weight matrix and bias:

$$e = \tanh(xW_{\text{att}} + b_{\text{att}}) \tag{2}$$

where $W_{\text{att}} \in \mathbb{R}^{128\times128}$ is the weight matrix and $b_{\text{att}} \in \mathbb{R}^{128}$ is a learned bias vector, both initialized using Glorot uniform and zeros, respectively.

The attention energies e are normalized using the softmax function across the feature dimension for each sample to produce attention weights:

$$a = \text{softmax}(e) \tag{3}$$

resulting in $a \in \mathbb{R}^{B\times128}$, with attention weights for each sample that sum to 1 across all features.

The output of the FeatureAttention layer is then computed by scaling each input feature by its corresponding attention weight:

$$\text{output} = x \odot a \tag{4}$$

where $\odot$ denotes element-wise multiplication over the feature dimension for each sample.

This matrix-based attention mechanism allows the model to adaptively amplify the most discriminative features and suppress irrelevant ones, thus improving the accuracy and robustness of intrusion detection. The computed attention weights can be extracted for further analysis or secure logging, supporting transparency and auditability of model decisions. Unlike standard self-attention mechanisms that model pairwise interactions between tokens in a sequence, the proposed FeatureAttention layer operates directly over the feature dimension of tabular IoMT traffic, using a single linear projection and softmax normalization to obtain per-feature importance scores with $O(d)$ complexity for d input features. This design is tailored to high-dimensional flow-based intrusion detection, where emphasizing discriminative network statistics is more critical than modeling long-range temporal dependencies.

D. AttentionChain Model Architecture

AttentionChain is a feedforward neural network integrating dense layers, batch normalization, dropout regularization, and the custom FeatureAttention layer. The architecture accepts 46 normalized network traffic features as input. The first dense layer maps inputs to 128 units with ReLU activation, followed by batch normalization and dropout (rate = 0.3) for regularization. The FeatureAttention layer then computes adaptive importance weights over these 128 features.

Following FeatureAttention, a dense layer reduces dimensionality to 64 units with ReLU activation, batch normalization, and dropout (0.3). A third dense layer further reduces to 32 units with ReLU activation and dropout (0.2). Finally, the output layer contains 19 units with softmax activation, producing probability distributions over the 19 attack classes. The complete model contains 34,259 trainable parameters distributed across dense layer weights and biases, batch normalization parameters, and the FeatureAttention weight matrix. This lightweight architecture enables real-time inference while maintaining sufficient expressive capacity for complex attack pattern detection in diverse IoMT network traffic.

E. Model Training

Model training minimizes Categorical Crossentropy loss:

$$L = -\sum_{k=1}^{19} y_k \log(\hat{y}_k) \tag{5}$$

where y_k is the one-hot encoded ground truth label and $\hat{y}_k$ is the predicted probability for class k. Training data labels are converted to one-hot encoded format: $y_{\text{onehot}} =$ to_categorical$(y, 19)$, creating a 19-dimensional binary vector for each sample.

Optimization is performed using Adam optimizer with learning rate $\alpha = 0.001$:

$$\theta_{t+1} = \theta_t - \alpha \frac{m_t}{\sqrt{v_t} + \epsilon} \tag{6}$$

where m_t is the exponential moving average of gradients (first moment) and v_t is the exponential moving average of squared gradients (second moment). The training configuration uses batch size 128, processing 3,785,376 training samples across 50 epochs. Early stopping with patience 10 is implemented, monitoring validation loss to prevent overfitting. Training is terminated when validation loss does not improve for 10 consecutive epochs. Total training time was 38.60 minutes on the complete training set.

F. Inference and Evaluation

Inference performs forward propagation through all trained layers for each test sample, generating probability scores across 19 attack classes. Predicted class is extracted as:

$$\hat{y}_{\text{pred}} = \arg\max_{k} \hat{y}_k \tag{7}$$

Performance is evaluated using scikit-learn metrics: overall accuracy, weighted precision, weighted recall, and weighted F1-score. To quantify the impact of incorrect alarms in IoMT settings, the false positive rate (FPR) is computed as:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{8}$$

Per-class metrics are computed for all 19 attack types from the classification report. Confusion matrix analysis reveals classification performance across all attack class pairs.

G. Blockchain Integration and Alert Logging

Detected intrusions are automatically logged to the Pure Chain blockchain to create immutable, cryptographically verifiable audit trails during inference. The framework connects to Pure Chain via Web3.py using the RPC endpoint and interacts with the IDChainLogger smart contract (address 0xf0fB8e25191d9bb6B95e7d997fB71b838FB12042), whose `logAlert` function accepts the attack type (string), confidence score (uint256), top features (string array), feature importance values (uint256 array), and a correctness flag (uint256).

For each prediction sample, Algorithm 1 extracts `attack_type`, confidence, and correctness, builds a `logAlert(...)` transaction, signs it with the private key, and submits it to Pure Chain. The client then waits for the confirmation receipt, records the transaction hash, block number, and gas used, and appends this metadata to a JSON log, ensuring that every detection decision is permanently linked to a verifiable on-chain record without manual intervention.

Algorithm 1 Blockchain Logging

```
Require: AttentionChain predictions, Pure Chain RPC URL, contract address
Ensure: Confirmed transactions on blockchain
Initialize Web3 connection to Pure Chain
Load IDChainLogger smart contract
for each prediction sample $i$ do
    Extract attack_type, confidence, correctness
    nonce ← w3.eth.get_transaction_count(...)
    Build tx ← logAlert(attack_type, confidence, top_features, importance_values, correctness)
    Configure gas parameters
    Sign tx with private key
    Send raw tx to blockchain
    Wait for confirmation receipt
    Record tx_hash, block_number, gas_used
end for
Save logs to JSON file
return Confirmed transactions array
```

H. Blockchain Performance Benchmarking

Pure Chain performance is evaluated to ensure suitability for real-time IoMT monitoring. Throughput (TPS) is calculated as:

$$\text{TPS} = \frac{\text{Number of confirmed transactions}}{\text{Total elapsed time in seconds}} \tag{9}$$

Transaction confirmation latency is measured as time from submission to block inclusion.

III. Performance Evaluation

A. Model Accuracy and Metrics

AttentionChain achieved 99.89% accuracy on the CIC-IoMT 2024 test set containing 1,081,536 network traffic samples. The model correctly classified 1,081,305 intrusions out of 1,081,536 test instances, corresponding to an overall false positive rate of 0.017%. The confusion matrix in figure 2 shows that the test precision, recall, and F1-score metrics all reached 99.89%.

TABLE I: AttentionChain Test Performance

Metric	Value
Test Accuracy	99.89%
Test Precision	99.89%
Test Recall	99.89%
Test F1-Score	99.89%
False Positive Rate	0.017%
Total Test Samples	1,081,536
Correct Classifications	1,081,305

B. Inference Performance

The model processed each network traffic sample in 0.0233 milliseconds. This resulted in a throughput of 42,960 samples per second. The complete test set of 1,081,536 samples was processed in 25.30 seconds. The model architecture contains 34,259 trainable parameters.

TABLE II: Inference Performance Results

Metric	Value
Inference Time per Sample	0.0233 milliseconds
Throughput	42,960 samples/second
Total Test Samples Processed	1,081,536
Model Parameters	34,259

C. Pure Chain Performance

Figure 3 shows that AttentionChain predictions were automatically logged to Pure Chain. Benchmarking results demonstrate 65.33 TPS throughput with mean latency 0.9 seconds. Block generation time on Pure Chain averaged 1.4–1.5 seconds, with approximately 5.2 transactions per block on average. Each blockchain transaction consumed an average of 238,737 gas. The median gas consumption was 241,665 gas.

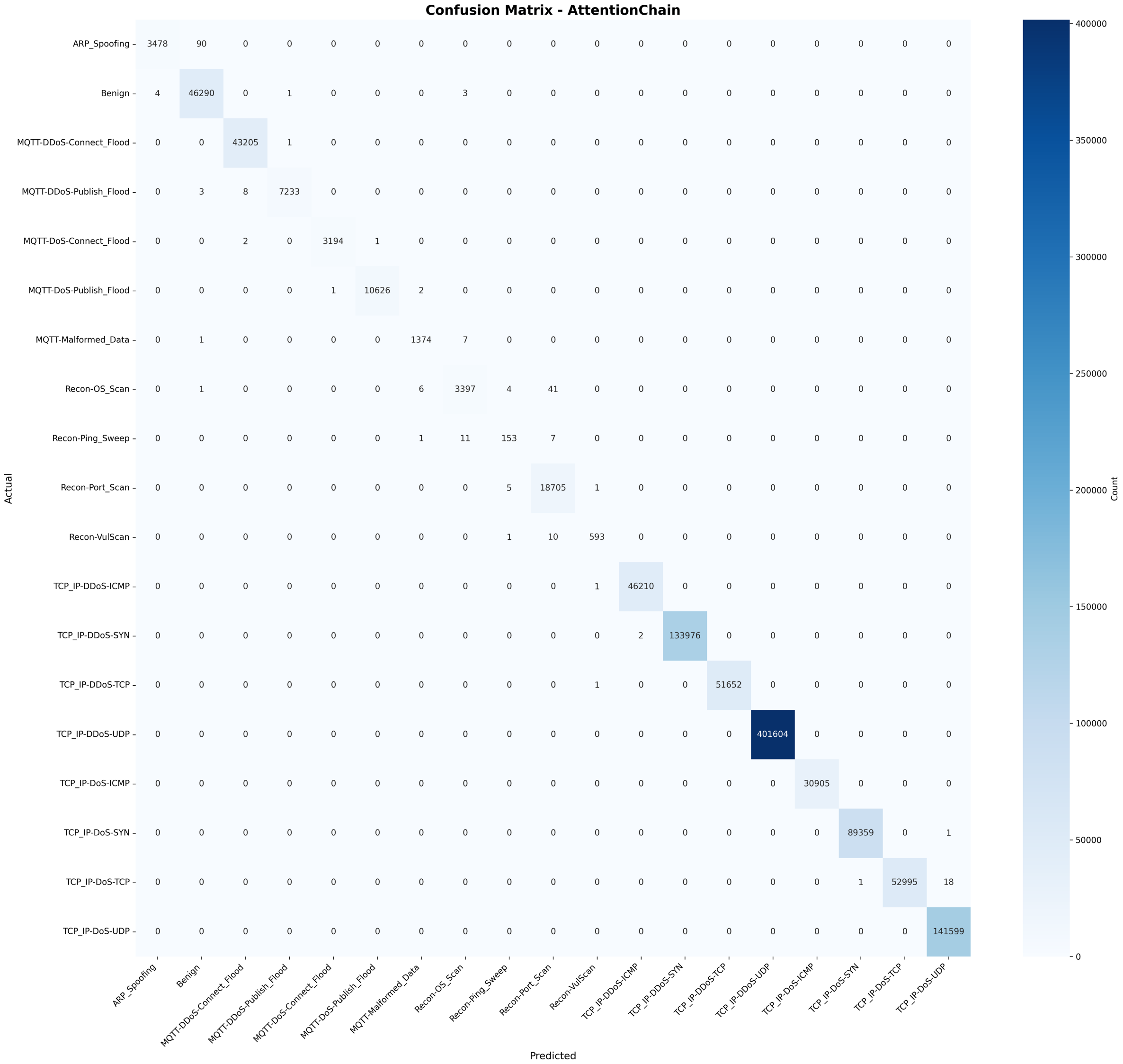
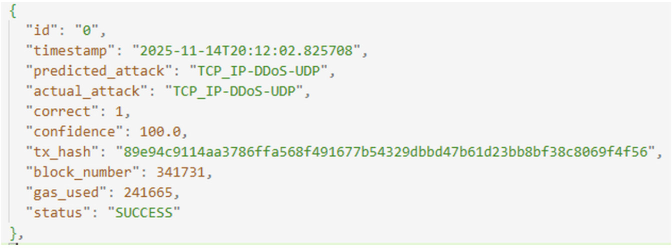


Fig. 2: Confusion Matrix of the Self-Attention Feedforward Neural Network Model



```
{
  "id": "0",
  "timestamp": "2025-11-14T20:12:02.825708",
  "predicted_attack": "TCP_IP-DDoS-UDP",
  "actual_attack": "TCP_IP-DDoS-UDP",
  "correct": 1,
  "confidence": 100.0,
  "tx_hash": "89e94c9114aa3786ffa568f491677b54329dbbd47b61d23bb8bf38c8069f4f56",
  "block_number": 341731,
  "gas_used": 241665,
  "status": "SUCCESS"
},
```

Fig. 3: AttentionChain predictions automatically logged to Pure Chain

D. Comparison with Recent Works

Table IV presents a comparative analysis between the proposed AttentionChain and recent intrusion detection frameworks. Compared to recent IDS approaches, the AttentionChain framework demonstrates the highest accuracy and F1-score with the lowest inference time on the CIC-IoMT2024 dataset, while uniquely providing permissioned blockchain auditability. Hybrid-DNN [17] and HIDS-RPL [18] offer strong detection performance on CIC-DDoS2019, but lack both auditability and real-time healthcare focus. The Explainable Transformer [19] achieves lower multiclass accuracy and does not support audit trails. This highlights that AttentionChain stands out by achieving state-of-the-art detection, real-time inference, and secure compliance for IoMT networks.

TABLE III: Pure Chain Performance

Metric	Value
Transactions Per Second (TPS)	65.33
Mean Latency	0.9 seconds
Block Generation Time	1.4–1.5 seconds
Average Transactions per Block	5.2

TABLE IV: Performance comparison with recent intrusion detection systems on CIC-IoMT2024 and CIC-DDoS2019 datasets.

System	Dataset	Method	Accuracy%	F1-Score%	Inference Time (ms)	Usage of Blockchain
Hybrid-DNN [17]	CIC-DDoS2019	XGBoost FS + CNN-LSTM	99.50	99.46	0.179	No
HIDS-RPL [18]	CIC-DDoS2019	CNN + LSTM Hybrid	99.87	98.54	Not Specified	No
Explainable Transformer [19]	CIC-IoMT2024	Transformer + XAI (LIME/SHAP)	97.4	97	Not Specified	No
AttentionChain (Proposed)	CIC-IoMT2024	Self-attention DL + Blockchain	99.89	99.89	0.0233	Permissioned

IV. Conclusion

AttentionChain successfully integrates a FeatureAttention-based self-attention deep learning model with Pure Chain blockchain to provide real-time IoMT intrusion detection with verifiable audit trails. On the CIC-IoMT 2024 dataset, the framework achieved 99.89% test accuracy with a 0.017% false positive rate, while processing 42,960 samples per second at 0.0233 ms per sample and completing training in 38.60 minutes. All detected intrusions were automatically logged to Pure Chain via the IDChainLogger smart contract, and benchmarking showed 65.33 transactions per second with mean latency of 0.9 seconds, indicating that immutable on-chain logging can be integrated without violating typical IoMT real-time monitoring requirements. These results demonstrate that AttentionChain can secure critical IoMT networks while simultaneously providing cryptographically verifiable audit trails for regulatory compliance and forensic investigations. Future work will explore federated learning techniques to distribute model training across multiple healthcare organizations while preserving data privacy and enabling interoperability across multiple permissioned blockchains.

Acknowledgment

This work was partly supported by Innovative Human Resource Development for Local Intellectualization program through the IITP grant funded by the Korea government (MSIT) (IITP-2025-RS-2020-II201612, 25%) and by Priority Research Centers Program through the NRF funded by the MEST (2018R1A6A1A03024003, 25%) and by the MSIT, Korea, under the ITRC support program (IITP-2025-RS-2024-00438430, 25%) and by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(RS-2025-25431637, 25%)

References

[1] D.-S. Kim and R. Syamsul, “Integrating Machine Learning with Proof-of-Authority-and-Association for Dynamic Signer Selection in Blockchain Networks,” *ICT Express*, 2024.

[2] D.-S. Kim, I. S. Igboanusi, L. A. C. Ahakonye, and G. O. Anyanwu, “Proof-of-Authority-and-Association Consensus Algorithm for IoT Blockchain Networks,” in *The 43rd IEEE International Conference on Consumer Electronics (ICCE 2025)*, 2025.

[3] N. Sengupta, R. Chinnasamy, and M. Subramanian, “Enhanced intrusion detection system for iomt devices using improved human evolutionary optimization algorithm and tabular transformers,” in *2024 International Conference on IT Innovation and Knowledge Discovery (ITIKD)*. IEEE, 2025, pp. 1–7.

[4] K. Deepthika, G. Shobana, K. V. Reddy, B. Kumar, S. Upadhyay *et al.*, “Blockchain-integrated deep learning for secure health data sharing and consent management,” in *2024 Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI)*. IEEE, 2024, pp. 101–106.

[5] S. K. Ghosh, M. Golam, M. S. Khaliq, M. M. H. Somrat, L. A. C. Ahakonye, J.-M. Lee, and D.-S. Kim, “Purechain for healthcare data sovereignty: Managing patient consent with smart contracts,” , pp. 1462–1463, 2025.

[6] P. Radoglou-Grammatikis, P. Sarigiannidis, G. Efstathopoulos, T. Lagkas, G. Fragulis, and A. Sarigiannidis, “A self-learning approach for detecting intrusions in healthcare systems,” in *ICC 2021-IEEE International Conference on Communications*. IEEE, 2021, pp. 1–6.

[7] B. Shanmugam and S. Azam, “Risk assessment of heterogeneous iomt devices: a review,” *Technologies*, vol. 11, no. 1, p. 31, 2023.

[8] J. Domenech, I. V. Martin-Faus, S. Mhiri, and J. Pegueroles, “Ensuring patient safety in iomt: A systematic literature review of behavior-based intrusion detection systems,” *Internet of Things*, vol. 28, p. 101420, 2024.

[9] A. Rai, M. Naik, and I. Seraphim B, “Leveraging blockchain technology for secure and efficient storage of medical data,” in *2024 IEEE 16th International Conference on Computational Intelligence and Communication Networks (CICN)*, 2024, pp. 652–656.

[10] S. R. Hassan, M. U. Tanveer, S. Prajapat, and M. Shabaz, “A comprehensive survey on intrusion detection in internet of medical things: Datasets, federated learning, blockchain, and future research directions,” *ICT Express*, 2025.

[11] J. A. Alzubi, O. A. Alzubi, I. Qiqieh, and A. Singh, “A blended deep learning intrusion detection framework for consumable edge-centric iomt industry,” *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 2049–2057, 2024.

[12] T. A. Alhaj, S. M. Abdulla, M. A. E. Iderss, A. A. A. Ali, F. A. Elhaj, M. A. Remli, and L. A. Gabralla, “A survey: To govern, protect, and detect security principles on internet of medical things (iomt),” *IEEE Access*, vol. 10, pp. 124 777–124 791, 2022.

[13] A. Mohammadi, H. Ghahramani, S. A. Asghari, and M. Aminian, “Securing healthcare with deep learning: A cnn-based model for medical iot threat detection,” in *2024 19th Iranian Conference on Intelligent Systems (ICIS)*. IEEE, 2024, pp. 168–173.

[14] N. Nezhadsistani, N. S. Moayedian, and B. Stiller, “Blockchain-enabled federated learning in healthcare: Survey and state-of-the-art,” *IEEE Access*, 2025.

[15] U. Zukaib, X. Cui, C. Zheng, M. Hassan, and Z. Shen, “Meta-ids: Meta-learning-based smart intrusion detection system for internet of medical things (iomt) network,” *IEEE Internet of Things Journal*, vol. 11, no. 13, pp. 23 080–23 095, 2024.

[16] S. Dadkhah, E. C. P. Neto, R. Ferreira, R. C. Molokwu, S. Sadeghi, and A. A. Ghorbani, “Ciciomt2024: A benchmark dataset for multi-protocol security assessment in iomt,” *Internet of Things*, vol. 28, p. 101351, 2024.

[17] A. Zainudin, L. A. C. Ahakonye, R. Akter, D.-S. Kim, and J.-M. Lee, “An efficient hybrid-dnn for ddos detection and classification in software-defined iiot networks,” *IEEE Internet of Things Journal*, vol. 10, no. 10, pp. 8491–8504, 2023.

[18] A. Berguiga, A. Harchay, and A. Massaoudi, “Hids-rpl: A hybrid deep learning-based intrusion detection system for rpl in internet of medical things network,” *IEEE Access*, vol. 13, pp. 38 404–38 429, 2025.

[19] R. Kalakoti, S. Nõmm, and H. Bahsi, “Explainable transformer-based intrusion detection in internet of medical things (iomt) networks,” in *2024 International Conference on Machine Learning and Applications (ICMLA)*, 2024, pp. 1164–1169.