# Deep Neural Network (DNN): Intrusion Detection System (IDS)

|  |  |  |
| --- | --- | --- |
| **Name** | **BITS ID** | **Contribution** |
| SUBHRANSU MISHRA | 2023AC05489 | 100% |
| JAWAHARLAL RAJAN S | 2023AC05504 | 100% |
| LAKSHMISRINIVAS PERAKAM | 2023AC05540 | 100% |
| SHAILESH KUMAR SINGH | 2023AC05475 | 100% |

## Selected Application Area: Intrusion Detection System (IDS)

An Intrusion Detection System (IDS) monitors network traffic to spot unusual or harmful activity. It alerts administrators when something suspicious is detected, helping protect systems from unauthorized access and cyber-attacks. The IDS can identify and report unusual patterns (anomalies), and some advanced systems can even take immediate action to block harmful traffic. These tools work by learning the difference between normal (good) and malicious (bad) connections to keep networks secure.

## Research Papers / Journals Explored:

|  |  |  |
| --- | --- | --- |
| Paper1 | ***Enhancing network intrusion detection systems with combined network and host traffic features using deep learning: deep learning and IoT perspective - CNN*** | [*Access Link*](https://link.springer.com/article/10.1007/s10791-024-09480-3) |
| Paper2 | ***IoT‑based blockchain intrusion detection using optimized recurrent neural network - RNN*** | [*Access Link*](https://link.springer.com/article/10.1007/s11042-023-16662-6) |
| Paper3 | ***Intrusion Detection Using Transformer in Controller Area Network - Transformer*** | [*Access Link*](https://ieeexplore.ieee.org/document/10659904) |

## Comparative Analysis:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Paper 1** | **Paper 2** | **Paper 3** |
| **Title of the Paper** | Enhancing Network Intrusion Detection Systems with Combined Network and Host Traffic Features Using Deep Learning: Deep Learning and IoT Perspective | IoT-based Blockchain Intrusion Detection Using Optimized Recurrent Neural Network | Intrusion Detection Using Transformer in Controller Area Network |
| **Authors** | Estabraq Saleem Abduljabbar Alars, Sefer Kurnaz | V. Saravanan, M. Madiajagan, Shaik Mohammad Rafee, P. Sanju, Tasneem Bano Rehman, Balachandra Pattanaik | Hyunjun Jo and Deok-Hwan Kim |
| **Year of Publication** | Nov-2024 | Sep-2023 | Aug-2024 |
| **Architecture of Deep Learning** | **Model Name**: **Hybrid CNN-LSTM**  The model combines **Convolutional Neural Networks (CNNs)** and **Long Short-Term Memory (LSTM)** networks for network intrusion detection (NIDS).  Key features include: **Layers**:  - multi-layer CNNs for spatial feature extraction.  - LSTMs for capturing temporal dependencies in traffic data.  - Dropout layer with a 0.5 dropout rate to prevent overfitting. **Activation**:  - CNN layers use **ReLU (Rectified Linear Unit)** for efficient non-linear transformations.  - LSTM layers use **Sigmoid** activation to model sequential patterns. **Unique Features**:  **-** Combines spatial and temporal feature extraction for a holistic intrusion detection system.  **-** Dimensionality reduction techniques like **PCA (Principal Component Analysis)** and **RFE (Recursive Feature Elimination)** improve computational efficiency.  **-** Designed to reduce false positives and detect both known and zero-day attacks effectively.reducing computational complexity | **Model Name**: Blockchain-based African Buffalo Optimization with RNN (Recurrent Neural Network) (*BbAB-RNN*).  Key features include:  **Layers**:  - Input layer for data preprocessing (features of normal and malicious traffic).  - Hidden layers using **Recurrent Neural Network (RNN)** for sequential pattern detection.  - Output layer for intrusion classification.  **Activation**:  Sigmoid activation function in hidden layers for learning sequential dependencies.  **Optimization**: African Buffalo Optimization **(ABO)** for enhancing the detection accuracy and reducing overfitting.  **Unique Features**:  - Integration with **blockchain technology** to securely store data.  - Use of **Identity-Based Encryption (IBE)** for encrypting and securing datasets in the blockchain. | **Model Name**: Transformer-based Intrusion Detection System for CAN (Controller Area Network).  **Layers**:  - Embedding layer and positional embedding for token representation.  - Stacked transformer blocks for spatial and temporal feature extraction.  - **Feedforward network** (FFN) within transformer blocks for feature compatibility.  - Linear layer for final classification (N-dimensional vector output).  **Activation**:  - ReLU (Rectified Linear Unit) in the feedforward layers for non-linear transformations.  - Multi-head self-attention mechanism for understanding token relationships.  **Unique Features**:  - Simultaneous analysis of CAN ID sequences (temporal) and payload sequences (spatial).  - Masking technique to prevent target CAN ID disclosure during training.  - Incorporation of unsupervised learning for adaptability to unpredictable attack scenarios. |
| **How the Network Helps in overall task (Feature Eng/Regression/Classification or All)** | The hybrid CNN-LSTM model helps by:   * Extracting spatial (topological) and temporal features for better classification. * Improving intrusion detection accuracy (98.5%) and reducing false positives. * Enhancing feature engineering by combining host and network traffic data. * Adapting to dynamic and evolving attack patterns by learning complex relationships between network events, making it suitable for zero-day attack detection. * Reducing computational overhead by applying feature selection techniques like PCA and RFE, which streamline data processing without compromising accuracy. | The BbAB-RNN model helps by:   * Detects intrusion by continuously monitoring network traffic using RNN. * Secures sensitive data with blockchain integration. * Reduces overfitting and improves accuracy through ABO. * Enhances feature engineering by combining IoT-based datasets with cryptographic encryption. * Mitigates computational complexity and improves intrusion detection rates with threshold-based attack identification. | The Transformer-based Intrusion Detection System for CAN helps by:   * Detects anomalies in both temporal (CAN ID sequences) and spatial (payload data) patterns. * Learns without labelled data using unsupervised learning, reducing dependency on manual annotation. * Identifies predictable and unpredictable attack patterns like flooding, fuzzy, and malfunction attacks. * Processes short window sizes efficiently, enabling real-time detection in resource-constrained environments. * Uses multi-head self-attention for robust feature extraction and context understanding. |
| **Training Procedures** | **Dataset Split**: 70% training, 15% validation, 15% testing.  **Optimizer**: Adam (Adaptive Momentum Estimation)  **Learning Rate**: 0.001  **Batch Size**: 128  **Epochs**: 50  **Regularization**: Dropout rate of 0.5  **Loss Function**: Categorical Cross-Entropy  **Validation**: 10-fold cross-validation | **Dataset Split:** 80% training, 20% testing.  **Optimizer:** African Buffalo Optimization (ABO).  **Learning Rate:** Adaptive, as tuned by ABO.  **Batch Size:** Not Available  **Regularization:** Enforced by ABO to prevent overfitting.  **Loss Function:** Not specified in paper | **Dataset Split**: Normal data for training; attack datasets (flooding, fuzzy, malfunction) for evaluation. 80% training 20% testing.  **Optimizer**: Adam optimizer for stable convergence.  **Learning Rate**: Tuned based on the dataset.  **Batch Size**: Not explicitly mentioned in paper.  **Regularization**: Attention-based mechanisms mitigate overfitting. |
| **Evaluation/Performance Metric Used** | * Accuracy * Precision * Recall * F1-Score * ROC-AUC | * Accuracy * Precision * Recall * F1-Score * Execution Time | * Precision * Recall * F1-Score * ROC Curve |
| **Name of Datasets** | Public Kaggle Dataset :  [Network Intrusion Detection](https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection) | Public Kaggle Dataset : [Malware Executable Detection](https://www.kaggle.com/datasets/piyushrumao/malware-executable-detection?select=uci_malware_detection.csv) | In-Vehicle Network Intrusion Detection Dataset shared by the Hacking and Countermeasures Research Lab in Korea.  URL: *No explicit link found in paper due to proprietary content.* |

## Conclusion:

Each paper addresses unique aspects of intrusion detection. The **CNN-LSTM model** provides balanced accuracy for general networks. The **blockchain-optimized RNN** adds security for IoT systems, while the **transformer-based IDS** is highly efficient for automotive cybersecurity. Together, they demonstrate the versatility of deep learning in tackling diverse intrusion detection challenges.