Intrusion Detection System (IDS) - Model Research & Technical Specifications

Team:

Lalith Kumar Raju Somalaraju

Shaik Abdul Gaffar

Sripathi Vamsi Krishna

Yasheswini Mallela

Executive Summary

This document presents comprehensive research and analysis of machine learning models for network intrusion detection. The research focuses on three primary model architectures: CNN-LSTM for sequential pattern recognition, Deep Neural Networks (DNN) for classification, and Autoencoders for anomaly detection. The ensemble approach combines these models to achieve robust threat detection with high accuracy and low false positive rates.

Dataset Research & Analysis

CIC-IDS-2017 Dataset

The Canadian Institute for Cybersecurity Intrusion Detection System (CIC-IDS-2017) dataset is a comprehensive benchmark for network intrusion detection research.

Dataset Characteristics:

- Total Records: 2,830,743 network flows
- Features: 78 statistical and timing features
- Attack Types: 15 different attack categories
- Time Period: 5 days of network traffic
- Data Format: CSV files with labeled flows
- Balance: Mix of normal and attack traffic

Attack Categories

• BENIGN: Normal network traffic
• DDoS: Distributed Denial of Service attacks
• PortScan: Port scanning activities
• Botnet: Botnet communication patterns
• Infiltration: Network infiltration attempts
• WebAttack: Web-based attacks (Brute Force, XSS, SQL Injection)
• Heartbleed: Heartbleed vulnerability exploitation

Feature Engineering Research

Feature engineering is crucial for effective intrusion detection. The research focuses on extracting meaningful features from raw network packets that can distinguish between normal and malicious traffic patterns.

Feature Categories

1. Statistical Features
- Packet count statistics (total, forward, backward)
- Packet size statistics (mean, std, min, max)
- Flow duration and idle time
- Packet rate calculations
- Byte count statistics
- Protocol-specific counters

2. Timing Features
- Inter-arrival times between packets
- Jitter calculations (packet delay variation)
- Flow duration and idle time statistics
- Packet timing patterns
- Burst detection and analysis
- Time-based flow characteristics

3. Protocol Features
- TCP flag analysis (SYN, ACK, FIN, RST)
- UDP characteristics and patterns
- ICMP message types and codes
- Protocol-specific flow analysis
- Port number analysis and patterns
- Service identification features

4. Flow Features
- Bidirectional flow analysis
- Flow direction and symmetry
- Connection state tracking
- Flow completion analysis
- Bidirectional packet ratios
- Flow-based statistical measures

Feature Alignment Process

The feature alignment process converts 78 extracted features to 66 features by removing constant columns that do not vary in real network traffic.

• Identify constant columns (zero variance)
• Remove 12 constant features from 78 total
• Maintain feature order and relationships
• Validate feature quality and completeness
• Ensure compatibility with trained models

Machine Learning Model Research

CNN-LSTM Model Architecture

The CNN-LSTM model combines Convolutional Neural Networks for local pattern recognition with Long Short-Term Memory networks for temporal sequence analysis.

- **Architecture Components:**

- Input Layer: 66 features as time-series data
- 1D CNN Layers: Extract local patterns and features
- Pooling Layers: Reduce dimensionality and overfitting
- LSTM Layers: Capture temporal dependencies and sequences
- Attention Mechanism: Focus on important time steps
- Dense Layers: Final classification and feature combination
- Output Layer: Multi-class classification (15 attack types)

Advantages:

- Captures both spatial and temporal patterns in network traffic
- Effective for sequential attack detection (DDoS, PortScan)
- Handles variable-length sequences naturally
- Attention mechanism improves interpretability
- Robust to noise and missing data
- High accuracy on sequential attack patterns

Deep Neural Network (DNN) Architecture

The Deep Neural Network is a multi-layer perceptron designed for high-performance classification of network intrusion patterns.

Architecture Components:

- Input Layer: 66 features as feature vector
- Hidden Layer 1: 128 neurons with ReLU activation
- Batch Normalization: Stabilize training and improve convergence
- Dropout Layer: Prevent overfitting (rate: 0.3)
- Hidden Layer 2: 64 neurons with ReLU activation
- Batch Normalization: Further stabilization
- Dropout Layer: Additional regularization (rate: 0.2)
- Hidden Layer 3: 32 neurons with ReLU activation
- Output Layer: 15 classes with softmax activation

Advantages:

- Fast inference speed for real-time applications
- Effective feature learning and representation
- Robust to feature scaling and normalization
- Good performance on high-dimensional data
- Reliable classification across different attack types
- Efficient memory usage and computation

Autoencoder Model Architecture

The Autoencoder model is designed for unsupervised anomaly detection by learning to reconstruct normal traffic patterns and identifying deviations.

Architecture Components:
- Input Layer: 66 features as input vector
- Encoder Layer 1: 32 neurons with ReLU activation
- Encoder Layer 2: 16 neurons with ReLU activation
- Bottleneck Layer: 8 neurons (compressed representation)
- Decoder Layer 1: 16 neurons with ReLU activation
- Decoder Layer 2: 32 neurons with ReLU activation
- Output Layer: 66 neurons (reconstructed input)

Advantages:
- Unsupervised learning - no labeled anomaly data required
- Detects unknown attack patterns and zero-day attacks
- Learns normal traffic patterns automatically
- Reconstruction error indicates anomaly severity
- Robust to new attack types not seen during training
- Complementary to supervised classification models

Ensemble Method Research

The ensemble method combines predictions from CNN-LSTM, DNN, and Autoencoder models to achieve robust and accurate intrusion detection.

Ensemble Architecture

Ensemble Components:
- CNN-LSTM Weight: 0.4 (temporal pattern recognition)
- DNN Weight: 0.3 (general classification)
- Autoencoder Weight: 0.3 (anomaly detection)
- Confidence Scoring: Weighted average of model confidences
- Threshold-based Decision: High confidence for threat classification
- Anomaly Detection: Reconstruction error from autoencoder

Ensemble Advantages:
- Improved accuracy through model diversity
- Reduced false positive rates
- Robust performance across different attack types
- Confidence-based decision making
- Combines supervised and unsupervised approaches
- Better generalization to unseen attack patterns

Performance Analysis & Results

Individual Model Performance

Model	Accuracy (%)	Precision (%)	Recall (%)
CNN-LSTM	92.5	91.8	93.2
DNN	89.3	88.7	90.1
Autoencoder	85.7	87.2	84.3
Ensemble	94.8	94.1	95.5

Real-time Performance Metrics

- Processing Speed: 1000+ packets per second
- Inference Latency: <100ms per packet
- Memory Usage: <2GB RAM for all models
- CPU Usage: Multi-threaded processing
- Model Loading Time: <5 seconds
- System Uptime: 99%+ availability

Model Training Process

Training Methodology

- Data Preprocessing: Clean and normalize CIC-IDS-2017 dataset
- Feature Engineering: Extract 78 features from network flows
- Feature Selection: Remove constant columns (78 → 66 features)
- Data Splitting: 70% training, 15% validation, 15% testing
- Model Training: Train each model with optimal hyperparameters
- Hyperparameter Tuning: Use Optuna for automated optimization
- Cross-validation: 5-fold cross-validation for robust evaluation

Research Contributions

- Novel ensemble approach combining CNN-LSTM, DNN, and Autoencoder
- Real-time feature engineering pipeline for live network traffic
- Dynamic threshold adjustment for anomaly detection in real-world traffic
- Comprehensive performance evaluation on CIC-IDS-2017 dataset
- Production-ready implementation with <100ms inference latency
- Robust error handling and system recovery mechanisms
- Professional web dashboard with real-time monitoring capabilities

Future Research Directions

- Integration of Graph Neural Networks for network topology analysis
- Federated learning approaches for distributed intrusion detection
- Adversarial training to improve robustness against evasion attacks
- Explainable AI techniques for model interpretability
- Real-time model adaptation and online learning capabilities
- Integration with blockchain for secure threat intelligence sharing
- Advanced visualization techniques for attack pattern analysis

Conclusion

This research presents a comprehensive approach to network intrusion detection using ensemble machine learning methods. The combination of CNN-LSTM, DNN, and Autoencoder models achieves high accuracy while maintaining real-time performance. The system demonstrates practical applicability with production-ready implementation, robust error handling, and professional user interface. The research contributes to the field of network security by providing an effective solution for real-time threat detection and anomaly detection.