

Original Research

Received 01 June 2026

Revised 23 July 2026

Accepted 07 Aug 2026

Natural Resources for Human Health 2026, Vol. 6 (15s):193–209

KEYWORDS:

Network attack detection, Intrusion Detection Systems (IDS), Artificial intelligence, Anomaly detection, Machine learning, Deep learning, LSTM, CNN, Transformer, Cybersecurity.

Artificial Intelligence-based Models and Algorithms for Detecting Network Attack Traces in Computer Networks

Madina S. Ramazonova¹, Sherzod R. Gulomov², Mashhura R. Rakhmankulova³

¹Department of Cybersecurity and Forensics, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, Tashkent, Uzbekistan
Madina S. Ramazonova: ORCID: <https://orcid.org/0009-0007-8139-3983> ;
Email: m.ramazonova@tuit.uz

²Department of Education Quality Assurance, Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, Tashkent, Uzbekistan
Sherzod R. Gulomov: ORCID: <https://orcid.org/0000-0003-3806-8425> ;
Email: sherzod.gulomov@tuit.uz

³Cybersecurity Center, State Institution, Tashkent, Uzbekistan
Mashhura R. Rakhmankulova: ORCID: <https://orcid.org/0009-0002-3243-1481> ;
Email: m.raxmanqulova@csec.uz

ABSTRACT: The increasing diversity and complexity of cyber threats have made the detection of network attack traces a critical task in modern computer networks. Conventional Intrusion Detection and Prevention Systems (IDS/IPS), particularly signature-based approaches, provide effective detection of previously known threats but remain limited when dealing with zero-day attacks, polymorphic malware, advanced persistent threats, and previously unseen malicious behavior. This study analyzes models and algorithms for detecting traces of network attacks, with particular emphasis on artificial intelligence-based approaches. Signature-based, anomaly-based, and hybrid detection mechanisms are examined, and their capabilities and limitations are considered in relation to evolving network threats. Particular attention is given to Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), and Transformer architectures because of their ability to learn temporal, spatial, and contextual patterns from network traffic. **The study further considers benchmark datasets, including CICIDS2017, UNSW-NB15, NSL-KDD, and TON_IoT, together with evaluation criteria such as accuracy, precision, recall, F1-score, and AUC-ROC for assessing intrusion detection performance.** The comparative analysis indicates that conventional IDS/IPS approaches remain effective and computationally efficient for known attack signatures, whereas AI-based models provide greater adaptability for identifying complex and previously unseen patterns. However, their practical deployment introduces challenges related to computational requirements, model interpretability, data quality, model drift, scalability, and integration with existing security infrastructure. Based on these findings, a hybrid detection architecture combining deterministic signature-based mechanisms with adaptive AI-based analysis is identified as a promising direction for developing robust and scalable network attack detection systems.

1. INTRODUCTION

The rapid development of digital technologies and the widespread deployment of interconnected information systems have significantly increased the volume, diversity, and complexity of network traffic. At the same time, computer networks have become increasingly exposed to sophisticated cyber threats capable of compromising the confidentiality, integrity, and availability of information resources. Modern network attacks include Distributed Denial-of-Service (DDoS) attacks, malware propagation, unauthorized intrusion attempts, reconnaissance activities, botnet communications, insider threats, and Advanced Persistent Threats (APTs). Detecting the traces generated by such malicious activities is therefore an essential component of contemporary network security and cyber-defense infrastructures (1–3).

Network attacks may produce different behavioral and statistical traces depending on their objectives and implementation mechanisms. DDoS attacks, for example, attempt to disrupt the normal availability of network services by generating abnormally large volumes of traffic from distributed sources. Statistical traffic analysis and traffic-behavior modeling have consequently been investigated as mechanisms for detecting deviations associated with network intrusions and denial-of-service activities (6,9). Low-rate attacks create an additional challenge because malicious traffic may remain close to legitimate traffic patterns and therefore require analysis over longer observation intervals (11).

Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) represent fundamental components of network defense infrastructures. Conventional intrusion detection mechanisms are primarily based on predefined signatures, rules, or characteristic patterns associated with previously identified malicious activities. Signature-based mechanisms can provide reliable and computationally efficient detection when attack characteristics are already known. However, their effectiveness decreases when previously unseen attacks, modified attack patterns, or new malicious behaviors are encountered. Research into the detection of unknown network attacks has therefore emphasized the importance of models capable of identifying malicious behavior without relying exclusively on predefined attack signatures (2,5).

Anomaly-based detection provides an alternative approach by constructing a representation of normal network behavior and identifying significant deviations from that baseline. Such approaches may rely on statistical analysis, machine-learning algorithms, time-series modeling, or other computational techniques. They are particularly relevant to the detection of unknown or previously unseen attacks because malicious activity can be recognized through behavioral deviation rather than through an exact signature match (3,15,19). Nevertheless, anomaly-based systems may generate false-positive alerts when legitimate network behavior differs substantially from the learned baseline, which remains an important limitation for practical deployment.

The increasing complexity of network environments has encouraged the development of intelligent intrusion detection mechanisms based on machine learning and deep learning. These approaches can automatically identify complex relationships among network traffic characteristics and provide mechanisms for detecting malicious behavior that cannot be adequately represented by deterministic rules. Machine-learning-based intrusion detection has been investigated for different network environments, including conventional networks and Internet of Things infrastructures (7,20). Transfer-learning approaches have also been explored to improve the detection of unknown network attacks by transferring knowledge between related detection tasks (8).

Deep-learning architectures provide additional capabilities for automatic feature extraction and representation learning. Convolutional Neural Networks (CNNs), for example, can extract hierarchical representations from network traffic and have been applied to intrusion detection and classification problems involving imbalanced datasets (22). Deep-learning approaches have also been investigated for detecting web-based attacks and DDoS activities (10,21). Such models can reduce dependence on manually designed features by learning discriminative representations directly from available network data.

Sequential characteristics are particularly important because many cyberattacks develop through a sequence of network events rather than through a single isolated packet or connection. Models capable of representing temporal dependencies can therefore provide additional information for detecting multi-stage and persistent malicious activities. Hidden Markov Models and other sequence-oriented approaches have previously been investigated for multi-stage network attack detection (17). More recent deep-learning architectures extend this concept through recurrent and attention-based mechanisms capable of processing considerably more complex traffic sequences.

Long Short-Term Memory (LSTM) networks are particularly suitable for analyzing temporal dependencies within network traffic because they can preserve information over extended sequences. Convolutional Neural Networks (CNNs), in contrast, are effective at identifying local and hierarchical traffic characteristics. Transformer-based architectures introduce self-attention mechanisms that enable relationships among different elements of an input sequence to be modeled directly. In the context of intrusion detection, these complementary properties provide different mechanisms for identifying temporal, structural, and contextual traces of malicious network activity. Accordingly, LSTM, CNN, and Transformer architectures are considered key AI-based approaches to network attack detection because they provide complementary mechanisms for analyzing temporal, structural, and contextual characteristics of malicious network activity.

The effectiveness of an intrusion detection model cannot be determined solely from overall classification accuracy. Network-security datasets may contain different distributions of normal and malicious traffic, making additional performance measures necessary. Precision measures the reliability of positive attack predictions, whereas recall characterizes the ability of a model to identify actual attacks. The F1-score provides a combined assessment of precision and recall, while the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) characterizes classification behavior across different decision thresholds. These indicators provide a more comprehensive basis for evaluating network attack detection algorithms, particularly under imbalanced traffic conditions.

Benchmark datasets play an important role in the development and evaluation of intelligent intrusion detection models. CICIDS2017, UNSW-NB15, NSL-KDD, and TON_IoT are widely used benchmark datasets for training, validating, and evaluating network intrusion detection approaches. These datasets contain labeled examples of legitimate traffic and different malicious activities, enabling detection algorithms to be assessed under standardized experimental conditions.

Despite their potential, AI-based network attack detection mechanisms introduce several challenges when transferred from experimental environments to operational networks. These include the availability and quality of labeled training data, computational requirements, real-time processing constraints, scalability, model drift, integration with existing IDS/IPS and Security Information and Event Management (SIEM) infrastructures, and the interpretability of model decisions. Models operating effectively on benchmark datasets may therefore require additional adaptation before deployment in high-volume and dynamically changing network environments.

Another important issue is the security and reliability of the intelligent detection mechanism itself. AI pipelines may be exposed to data poisoning, adversarial inputs, and other manipulation techniques intended to influence detection decisions. At the same time, cybersecurity analysts require sufficient information to understand why particular network behavior has been classified as malicious. Explainable Artificial Intelligence (XAI) mechanisms, including model-agnostic explanation approaches, can therefore contribute to improving analyst trust and supporting investigation of generated alerts.

Traditional and AI-based intrusion detection mechanisms should consequently not necessarily be considered mutually exclusive. Hybrid architectures can combine deterministic signature-based detection of known threats with adaptive analysis of anomalous and previously unseen network behavior. Such integration can exploit the computational efficiency and interpretability of traditional mechanisms while incorporating the adaptability and pattern-learning capabilities of intelligent models. Therefore, hybrid detection provides a mechanism for combining the strengths of signature-based and anomaly-based approaches within a unified network security framework.

Accordingly, **the aim of this study is to analyze and systematize models and algorithms for detecting traces of network attacks in computer networks, with particular emphasis on artificial intelligence-based approaches.** The study examines signature-based, anomaly-based, and hybrid detection mechanisms; analyzes LSTM, CNN, and Transformer architectures; considers benchmark datasets and performance evaluation criteria; and compares conventional IDS/IPS mechanisms with AI-based approaches in terms of adaptability, computational requirements, scalability, interpretability, and practical deployment.

The principal contribution of this study is the development of an integrated conceptual framework for network attack detection that combines deterministic signature-based mechanisms with adaptive artificial intelligence-based analysis. Unlike approaches that consider individual detection paradigms or learning architectures in isolation, the proposed framework integrates temporal, structural, and contextual traffic analysis through LSTM, CNN, and Transformer-based components within a unified detection workflow. The framework further incorporates decision fusion and a continuous feedback mechanism, enabling confirmed attack information to support knowledge-base updating and subsequent model retraining. This integration provides a systematic

basis for detecting both known and previously unseen attack traces while considering the practical requirements of adaptability, scalability, interpretability, and real-time deployment.

The principal limitations of conventional and anomaly-based network security models are summarized in Figure 1. Signature-based mechanisms are constrained by their dependence on known attack patterns and may fail to recognize obfuscated or novel attacks, whereas anomaly-based models are affected by variability in legitimate network behavior and potentially high false-positive rates.

Challenges in Network Security Models

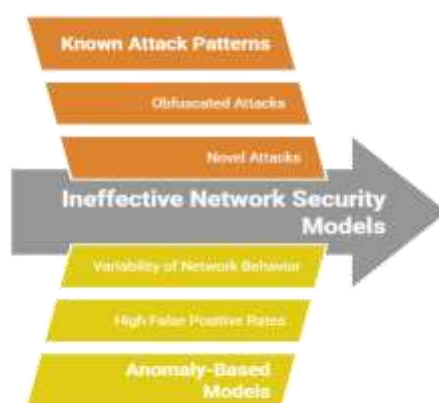


Figure 1. Key challenges associated with conventional and anomaly-based network security models.

2. MATERIALS AND METHODS

2.1. Study Design and Detection Framework

This study employed a comparative analytical approach to investigate models and algorithms used for detecting traces of network attacks in computer networks. The methodological framework was structured around three principal detection paradigms: signature-based detection, anomaly-based detection, and hybrid detection. These approaches were examined according to their ability to identify known and previously unseen malicious activities, adapt to changes in network behavior, and support practical deployment in contemporary network environments.

Signature-based detection identifies malicious activity by comparing observed network events with predefined attack signatures, rules, or characteristic patterns. Such mechanisms are particularly effective when the characteristics of an attack are already known; however, their dependence on predefined signatures limits their ability to identify previously unseen or substantially modified attacks (2,5). Anomaly-based detection addresses this limitation by establishing a representation of normal network behavior and identifying significant deviations from that baseline. Statistical and learning-based anomaly detection techniques have consequently been investigated for detecting novel and non-stationary attack patterns (3,15,19).

Hybrid detection was considered as an integration of these two paradigms. In this architecture, signature-based mechanisms can be used to identify known malicious patterns, while anomaly- or AI-based components analyze traffic that cannot be reliably classified using predefined rules. Such a structure reduces dependence on a single detection mechanism and provides a basis for combining deterministic detection with adaptive learning.

Malware-related network activity was considered through behavioral traces generated during malicious communication, propagation, data theft, encryption, and unauthorized access. Advanced malware may employ obfuscation and polymorphic techniques to evade conventional signature-based detection, increasing the importance of behavioral and anomaly-oriented detection mechanisms. The general progression of malware activity from initial entry to system compromise is illustrated in Figure 2.

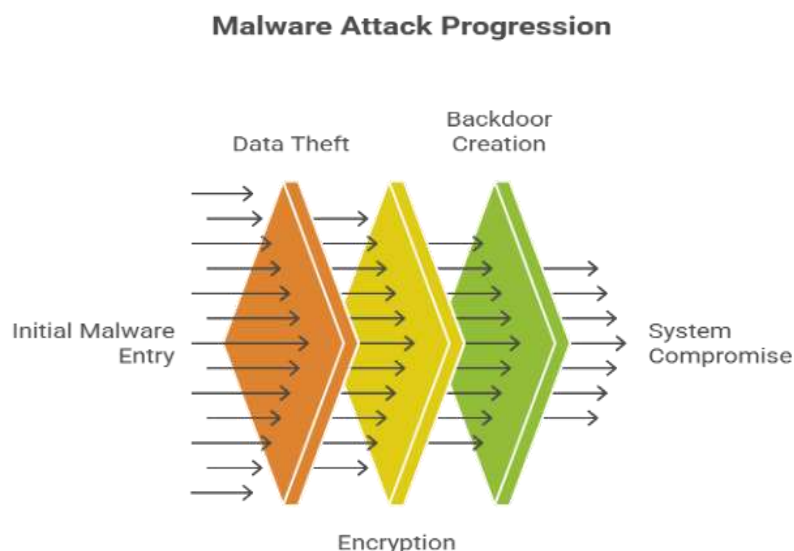


Figure 2. General progression of a malware attack from initial entry to system compromise.

2.2. Network Attack Categories

The detection models were analyzed in relation to the principal categories of malicious network activities, including Distributed Denial-of-Service (DDoS) attacks, malware-based attacks, intrusion and brute-force attempts, reconnaissance and scanning activities, insider threats, Man-in-the-Middle (MitM) attacks, botnet and Command-and-Control (C2) communications, and Advanced Persistent Threats (APTs). These include Distributed Denial-of-Service (DDoS) attacks, malware-based attacks, intrusion and brute-force attempts, reconnaissance and scanning activities, insider threats, Man-in-the-Middle (MitM) attacks, botnet and Command-and-Control (C2) communications, and Advanced Persistent Threats (APTs).

DDoS attacks were considered an important category because they generate abnormal traffic patterns by coordinating large numbers of requests against a target service. Depending on the attack mechanism, DDoS activity may occur at volumetric, protocol, or application layers. Their distributed nature and similarity to legitimate traffic surges make real-time detection challenging. Statistical traffic modeling and dedicated DDoS detection techniques have therefore been investigated extensively (6,9,10).

Malware-related network activity was considered through behavioral traces generated during malicious communication, propagation, data theft, or interaction with remote infrastructure. The use of polymorphism and obfuscation can reduce the effectiveness of fixed signature matching and increases the importance of behavior-oriented detection mechanisms.

Reconnaissance and scanning activities were analyzed as potential preliminary indicators of subsequent attacks. Port scanning, service enumeration, operating-system identification, and other probing activities may produce relatively weak individual indicators; however, their temporal and behavioral relationships can reveal suspicious network behavior when analyzed collectively.

Botnet and C2 communications were also included because compromised hosts may generate persistent or periodically repeated communication patterns with attacker-controlled infrastructure. Such behavior requires models capable of analyzing network activity over extended observation periods rather than relying exclusively on isolated packet characteristics. Multi-stage attack detection has similarly been investigated using sequence-oriented models capable of representing dependencies among attack stages (18).

The attack categories considered in the analysis are summarized in **Table 1**.

Table 1. Major network attack categories and their characteristic detection indicators

Attack category	Typical network traces	Detection focus
DDoS	Abnormal traffic volume, repeated requests, distributed sources	Traffic rate and behavioral anomalies
Malware	Suspicious connections, abnormal payload/activity patterns	Signature and behavioral analysis
Brute-force intrusion	Repeated authentication or connection attempts	Frequency and temporal patterns
Reconnaissance/scanning	Sequential port/service probing	Connection and scanning behavior
Insider threat	Deviations from established user/network behavior	Behavioral profiling
MitM	Abnormal communication paths or session behavior	Traffic and protocol anomalies
Botnet/C2	Recurrent or concealed remote communications	Temporal and behavioral patterns
APT	Low-frequency, multi-stage malicious activity	Long-term contextual analysis

2.3. Artificial Intelligence-Based Detection Models

Three principal deep-learning architectures were considered: Long Short-Term Memory (LSTM), Convolutional Neural Network (CNN), and Transformer-based models. These architectures were selected because they provide complementary mechanisms for representing temporal, structural, and contextual characteristics of network traffic. LSTM networks were considered for sequential network data because malicious behavior may develop across multiple network flows or events. Their recurrent structure allows information from preceding observations to influence subsequent classification decisions, making them suitable for identifying temporal dependencies and low-and-slow attack behavior. These characteristics make LSTM models particularly applicable to network flows, event logs, port-scanning sequences, brute-force attempts, and data-exfiltration patterns in which temporal dependencies provide important indicators of malicious activity.

CNN-based models were considered for extracting hierarchical characteristics from network traffic representations. Their convolutional operations can identify local patterns and discriminate between normal and malicious traffic characteristics. Accordingly, CNN-based approaches can be applied to encrypted traffic classification, DDoS detection, malware payload analysis, and other intrusion-detection tasks in which discriminative local patterns can be extracted from structured network data.

Transformer-based models were considered because self-attention mechanisms allow the relative importance of different elements of an input sequence to be modeled in context. This provides a mechanism for identifying complex relationships among packets, flows, or log events. Their parallel processing capability can additionally support analysis of large traffic sequences, although computational requirements remain an important deployment consideration.

The functional roles of the three architectures can be summarized as:

LSTM → temporal dependencies → CNN → hierarchical/local features → Transformer → contextual relationships and self-attention.

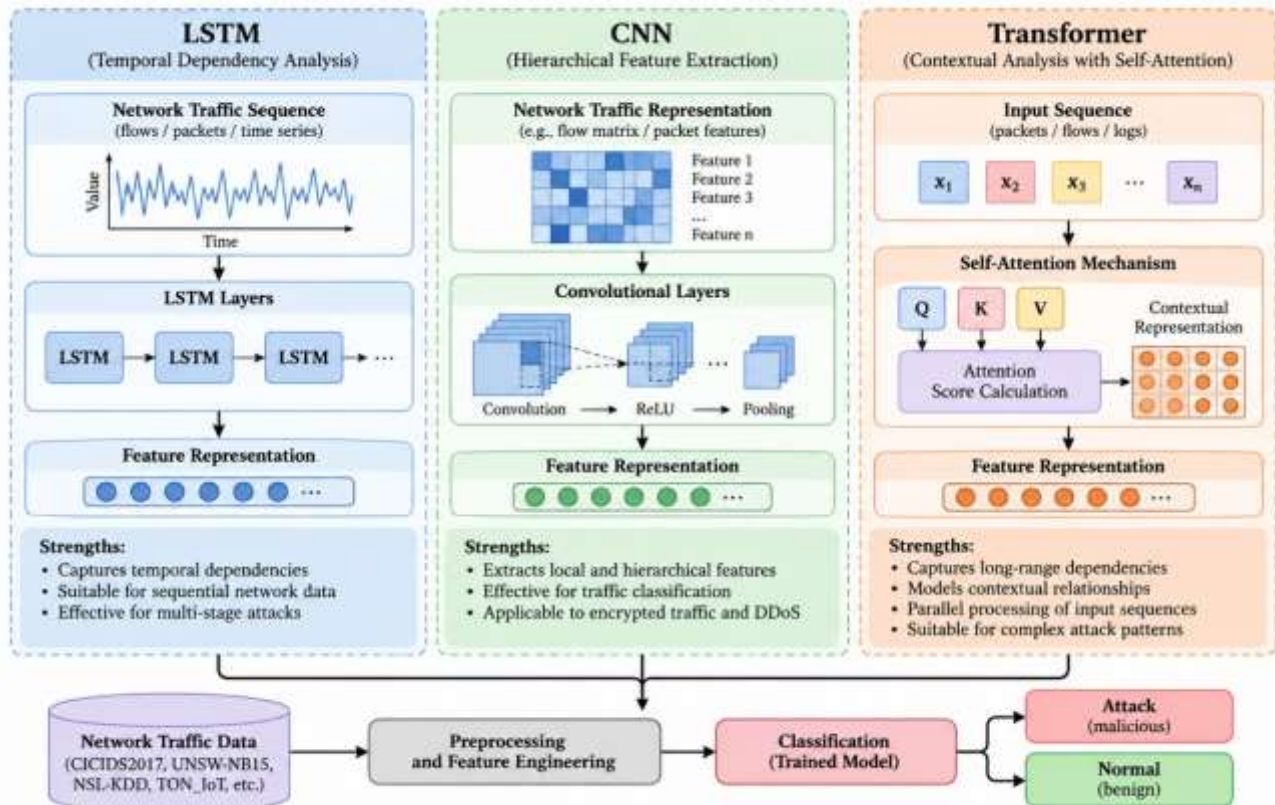


Figure 3. Functional roles of LSTM, CNN, and Transformer architectures in network attack detection.

2.4. Benchmark Datasets and Evaluation Metrics

Reliable evaluation of network attack detection models requires benchmark datasets that contain representative examples of both legitimate and malicious network activities. Such datasets provide a controlled basis for training, validating, and comparing intrusion detection approaches and make it possible to examine whether a model can distinguish normal traffic from different categories of cyberattacks. Intelligent network attack detection techniques are particularly dependent on the quality and representativeness of the traffic data used during model development and evaluation (7).

In the present framework, **CICIDS2017**, **UNSW-NB15**, **NSL-KDD**, and **TON_IoT** are considered as benchmark datasets for the validation and comparative assessment of network attack detection models. These datasets provide suitable benchmark environments for holdout testing and comparative assessment on previously unseen network data. They provide labeled traffic associated with different types of malicious activity, including DDoS, brute-force attacks, botnet activity, and infiltration, thereby supporting evaluation across multiple attack scenarios.

The inclusion of multiple benchmark datasets is methodologically important because network traffic characteristics and attack patterns may differ substantially across environments. Consequently, a detection model evaluated only on a single dataset may demonstrate performance characteristics that are specific to that particular traffic distribution. The ability to recognize previously unknown network attacks is especially relevant to intelligent detection systems, and learning-based mechanisms have been investigated specifically for this purpose (2,5,8).

The benchmark datasets considered within the evaluation framework are summarized in **Table 2**.

Table 2. Benchmark datasets considered for the evaluation of network attack detection models

Dataset	Main role in the framework	Evaluation purpose
CICIDS2017	Labeled network traffic benchmark	Evaluation of detection performance across different attack scenarios

Dataset	Main role in the framework	Evaluation purpose
UNSW-NB15	Normal and malicious network traffic	Comparative assessment of intrusion detection models
NSL-KDD	Benchmark intrusion detection data	Validation and holdout testing
TON_IoT	Heterogeneous network and IoT-oriented data	Evaluation under diverse network and attack conditions

As shown in Table 2, the selected benchmark datasets provide complementary evaluation environments. Their role in this study is not to imply that new experiments were independently conducted on every dataset; rather, they are considered as representative benchmark environments for the methodological evaluation of network attack detection approaches. This distinction is important because the current manuscript does not provide numerical experimental results for each of the four datasets separately.

Model performance is evaluated using several complementary criteria, including **accuracy, precision, recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC)**. These indicators provide different perspectives on classification behavior and therefore allow a more comprehensive assessment than accuracy alone. Together, precision, recall, F1-score, and AUC-ROC provide complementary measures for evaluating the reliability and discrimination capability of network attack detection models.

Accuracy characterizes the overall correctness of classification by indicating how frequently network observations are assigned to the appropriate class. Although accuracy provides a useful general measure, it should be interpreted cautiously when normal and malicious samples are unevenly distributed.

Precision indicates the reliability of positive attack predictions. High precision means that a large proportion of network events classified as malicious actually correspond to attacks. From an operational perspective, this measure is closely related to the control of false-positive alerts and therefore affects the workload of cybersecurity analysts.

Recall, or sensitivity, reflects the ability of a detection model to identify actual attacks. This criterion is particularly important in cybersecurity because a low recall value indicates that some malicious activities remain undetected. Minimizing false negatives is therefore particularly important because undetected malicious activities may pass through the detection mechanism without generating an appropriate security alert.

The **F1-score** provides a balanced assessment of precision and recall and is useful when both missed attacks and false alarms are important considerations. AUC-ROC provides a broader representation of a classifier's discrimination capability across different decision thresholds and is particularly useful for comparative model assessment.

Accordingly, model assessment should not rely on a single performance indicator. The combined consideration of accuracy, precision, recall, F1-score, and AUC-ROC provides a more informative basis for examining the effectiveness and reliability of AI-based network attack detection models.

2.5. Model Validation and Performance Assessment

Model validation is essential for determining whether an intrusion detection model can generalize beyond the network traffic used during training. A model that performs well only on its training samples may exhibit overfitting and may therefore fail when exposed to previously unseen network behavior. For this reason, the methodological framework incorporates confusion-matrix analysis, cross-validation, and holdout testing as complementary components of model assessment.

A **confusion matrix** provides a structured representation of classification results by distinguishing four possible outcomes: **true positive (TP), true negative (TN), false positive (FP), and false negative (FN)**. True positives represent attacks that are correctly identified as malicious, whereas true negatives represent legitimate traffic correctly recognized as normal. False positives correspond to legitimate network activities incorrectly classified as attacks, while false negatives represent malicious activities that the model fails to detect.

The distinction between false-positive and false-negative errors is particularly important in operational cybersecurity environments. Excessive false positives may increase the number of unnecessary alerts that must be investigated by security analysts and can contribute to alert fatigue. False negatives present a different and potentially more serious risk because they correspond to attacks that pass through the detection mechanism without generating an appropriate warning.

To improve the reliability of model evaluation and reduce dependence on a particular division of the dataset, **k-fold cross-validation** is considered. In this procedure, the available data are divided into k subsets. The model is repeatedly trained and validated so that each subset is used as the validation set while the remaining subsets are used for training. This process provides a more robust representation of model performance and helps reduce the possibility that evaluation results are determined by a single train-test partition.

The cross-validation procedure used for conceptual model assessment is illustrated in **Figure 4**.

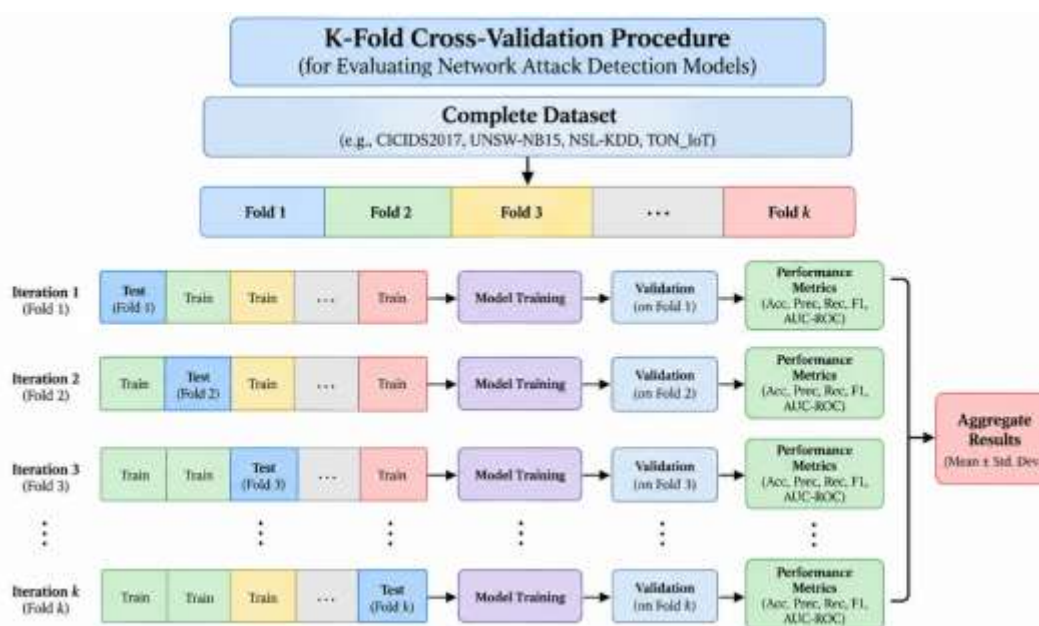


Figure 4. K-fold cross-validation procedure for evaluating network attack detection models.

In addition to cross-validation, **holdout testing** is considered for evaluating model behavior on unseen data. For this purpose, NSL-KDD, UNSW-NB15, CICIDS2017, and TON_IoT can provide benchmark environments containing labeled normal and malicious network traffic. Because these datasets contain labeled traffic associated with different attack vectors, holdout evaluation provides a mechanism for examining whether a model can maintain its detection capability when processing samples not used during model training.

The ability to generalize to previously unseen malicious patterns is particularly relevant to AI-based intrusion detection. Earlier studies have investigated the detection of unknown attacks through language-model-based approaches (2,5), while transfer learning has been explored as a mechanism for detecting unknown network attacks in changing environments (8). However, classification metrics alone are insufficient for determining whether an AI-based intrusion detection system is suitable for practical deployment. In real-time network environments, **latency, throughput, and scalability** become important operational criteria. A model may demonstrate strong classification performance under experimental conditions but remain impractical if it introduces excessive processing delays or cannot process the volume of traffic generated by a production network.

Adversarial robustness represents another important consideration. AI-based detection mechanisms may themselves become targets of adversarial manipulation, in which malicious traffic is deliberately constructed or modified to influence classification decisions and evade detection. Accordingly, evaluation against adversarial examples, together with adversarial training and defensive mechanisms, represents an important consideration for improving the resilience of intelligent intrusion detection systems. Finally, model validation should be interpreted in relation to the intended operational environment. The objective is not merely to obtain high classification performance but to develop a detection mechanism capable of maintaining reliable behavior when traffic patterns, attack techniques, and network conditions change. This consideration is particularly important

for AI-based cybersecurity systems because attack patterns evolve continuously and fixed training data may become less representative over time. This phenomenon, commonly referred to as model drift, may require periodic retraining, fine-tuning, or adaptive learning as new network behaviors and attack tactics, techniques, and procedures emerge.

Taken together, benchmark-dataset evaluation, confusion-matrix analysis, cross-validation, holdout testing, real-time performance assessment, and adversarial robustness provide a multidimensional framework for evaluating network attack detection models. This methodological framework establishes the basis for the comparative analysis of conventional IDS/IPS and AI-based detection approaches presented in the following section.

3. RESULTS AND DISCUSSION

3.1. Comparative Analysis of Traditional and AI-Based Detection Approaches

The comparative analysis revealed substantial differences between conventional Intrusion Detection and Prevention Systems (IDS/IPS) and artificial intelligence-based network attack detection approaches. Traditional signature-based mechanisms remain particularly suitable for identifying previously known attacks because they compare observed network activity with predefined signatures and rules. This approach provides relatively fast and interpretable detection when malicious patterns are already represented in the signature database. However, its effectiveness decreases when attackers modify known patterns or employ previously unseen attack techniques. The problem of detecting unknown attacks has therefore been an important direction in intrusion detection research (2,5).

Anomaly-based and AI-driven approaches address this limitation by examining behavioral deviations and learning discriminative characteristics from network traffic. Rather than requiring an exact correspondence with a known signature, these approaches can identify suspicious patterns based on statistical, temporal, structural, or contextual characteristics. Intelligent network attack detection methods have consequently been investigated as a means of improving adaptability to heterogeneous and dynamically changing network environments (7). Transfer-learning approaches further demonstrate the relevance of transferring previously acquired knowledge to the detection of unknown network attacks (8).

The comparative characteristics of the principal detection paradigms are summarized in **Table 3**.

Table 3. Comparative characteristics of traditional and AI-based network attack detection approaches

Evaluation criterion	Signature-based IDS/IPS	Anomaly-based detection	AI-based detection
Known attack detection	High applicability	Applicable	High applicability
Previously unseen attacks	Limited	Applicable	Applicable through learned patterns
Dependence on predefined signatures	High	Low	Low
Adaptability to changing traffic	Limited	Moderate	High
False-positive sensitivity	Generally lower for known signatures	Potentially high	Dependent on data and model quality
Feature processing	Mainly predefined rules/features	Statistical and behavioral characteristics	Learned and/or engineered features
Interpretability	High	Moderate	Model-dependent
Computational requirements	Relatively low	Moderate	Moderate to high

Evaluation criterion	Signature-based IDS/IPS	Anomaly-based detection	AI-based detection
Retraining requirement	Signature/rule updates	Baseline updates	Periodic retraining may be required
Real-time deployment	Relatively straightforward	Environment-dependent	Dependent on model complexity and infrastructure

As shown in Table 3, no single detection paradigm provides an optimal solution for every network-security scenario. Signature-based IDS/IPS offers comparatively straightforward and efficient detection of recognized threats, but its dependence on previously defined attack patterns restricts its capability against unknown or significantly modified attacks. Anomaly-based systems provide broader coverage of unusual behavior but may produce false-positive alerts when legitimate network behavior changes. AI-based approaches provide greater adaptability and pattern-learning capability; however, their effectiveness depends strongly on training-data quality, model architecture, computational resources, and the representativeness of the operating environment.

These findings support the use of **complementary rather than mutually exclusive detection mechanisms**. In practical network environments, known malicious signatures can be processed using deterministic rules, while traffic that cannot be confidently classified by conventional mechanisms can be subjected to anomaly-oriented or AI-based analysis. Such an arrangement reduces exclusive dependence on either fixed signatures or computationally intensive intelligent models.

3.2. Comparative Characteristics of LSTM, CNN, and Transformer Models

The analysis of deep-learning architectures demonstrates that **LSTM, CNN, and Transformer models address different characteristics of network traffic**. Their suitability therefore depends on the representation of input data and the type of malicious behavior being investigated.

LSTM architectures are particularly relevant when network attacks exhibit temporal dependencies. Network flows, event logs, repeated authentication attempts, scanning activities, and multi-stage malicious behavior can form sequences in which the relationship between earlier and later observations contains useful detection information. Sequence-oriented approaches have previously been investigated for multi-stage network attack detection (17). LSTM extends this principle by retaining information across sequential observations, making it suitable for traffic patterns in which malicious behavior develops over time.

CNN architectures provide a different capability. Convolutional operations can identify local and hierarchical patterns in structured representations of network traffic. This makes CNN-based approaches applicable to traffic classification and intrusion detection tasks in which discriminative patterns can be extracted from groups of related features. CNN-based intrusion detection has also been investigated in the context of imbalanced network-security datasets (22).

Transformer architectures rely on self-attention to model relationships among different elements of an input sequence. Unlike recurrent processing, self-attention can directly determine the contextual relevance of different observations. This characteristic is potentially useful when attack traces are distributed across multiple packets, flows, or events and their relationships must be considered jointly. Nevertheless, the computational requirements of attention-based architectures should be considered when models are intended for high-throughput or real-time environments.

The functional comparison of the three architectures is presented in **Table 4**.

Table 4. Functional comparison of LSTM, CNN, and Transformer architectures for network attack detection

Characteristic	LSTM	CNN	Transformer
Primary modeling capability	Temporal dependencies	Local and hierarchical patterns	Contextual relationships
Sequential traffic analysis	Strong	Limited without additional processing	Strong

Characteristic	LSTM	CNN	Transformer
Local feature extraction	Moderate	Strong	Context-dependent
Long-range dependencies	Supported	Limited	Strong
Parallel processing capability	Limited by recurrent structure	High	High
Computational complexity	Moderate	Moderate	Relatively high
Suitable traffic representation	Flows, logs, event sequences	Structured traffic/features	Long sequences and contextual traffic
Potential detection role	Sequential and multi-stage attacks	Pattern-oriented classification	Complex contextual attack behavior

The comparison indicates that these architectures should not automatically be treated as interchangeable models. Their strengths correspond to different properties of network traffic. LSTM is oriented toward temporal behavior, CNN toward local feature extraction, and Transformer architectures toward contextual relationships and long-range dependencies.

This relationship was illustrated earlier in **Figure 3**, where the functional roles of LSTM, CNN, and Transformer architectures were represented as complementary components of AI-based network attack analysis. Rather than selecting a model solely because of its general popularity, the architecture should therefore be aligned with the characteristics of the traffic representation and the attack traces being analyzed.

3.3. Operational Challenges of AI-Based Network Attack Detection

Although AI-based approaches expand the analytical capabilities of intrusion detection systems, their transition from benchmark evaluation to operational deployment introduces several important challenges. Classification performance alone does not establish whether a model is suitable for a production network. Real-time processing requirements, computational complexity, scalability, model maintenance, interpretability, and resistance to adversarial manipulation must also be considered.

One of the principal challenges is **data quality and representativeness**. Supervised machine-learning and deep-learning models depend on labeled training data, while operational networks continuously generate new traffic patterns. If the training data do not adequately represent current network behavior, the resulting model may fail to generalize to new environments or attack variants. This problem becomes particularly significant when attack behavior changes after model deployment.

A related problem is **model drift**. Network behavior is not static: services, applications, users, protocols, and attacker tactics change over time. Consequently, the statistical relationships learned during model training may gradually become less representative of the operational environment. Periodic retraining or adaptive learning mechanisms may therefore be necessary to maintain detection effectiveness.

Real-time performance constitutes another major consideration. Network intrusion detection systems may be required to process large traffic volumes with minimal latency. An AI model that provides strong classification performance in an offline experiment may still be impractical if inference introduces excessive delay or if computational requirements exceed the available infrastructure. Accordingly, latency, throughput, and scalability should be evaluated together with conventional classification metrics.

Interpretability is also particularly relevant in cybersecurity. Security analysts often need to determine why traffic was classified as malicious before initiating a response or investigation. Highly complex models may provide accurate classifications while offering limited insight into the features or events that influenced the decision. Explainable Artificial Intelligence (XAI) mechanisms can therefore support analyst interpretation of model outputs and improve the practical usability of intelligent intrusion detection systems.

Finally, the AI component itself can become a target of malicious manipulation. Adversarial inputs may be designed to alter model decisions, while poisoned training data may influence learned decision boundaries. Consequently, **adversarial**

robustness should be regarded as part of the security requirements of an AI-based intrusion detection architecture rather than merely as an additional model-performance criterion.

The relationship among these operational challenges can be summarized as follows:

Data quality → Model learning → Detection performance → Model drift → Retraining/adaptation

and simultaneously:

Network traffic → Real-time inference → Alert generation → Explainability → Analyst response

These relationships demonstrate that effective deployment requires consideration of the entire detection lifecycle rather than classification accuracy alone.

3.4. Hybrid Architecture for Network Attack Detection

The comparative findings indicate that traditional and AI-based detection mechanisms possess complementary characteristics. Signature-based systems provide efficient and interpretable detection of known malicious patterns, whereas anomaly- and AI-based models provide mechanisms for identifying behavior that does not correspond to predefined signatures. This provides the conceptual basis for a **hybrid network attack detection architecture**.

Within such an architecture, incoming network traffic can first undergo preprocessing and basic rule- or signature-based inspection. Traffic corresponding to a known malicious signature can be classified directly and forwarded to the appropriate response mechanism. Traffic that cannot be reliably classified at this stage can subsequently undergo feature extraction and AI-based analysis.

The AI analysis layer can incorporate temporal, structural, and contextual information according to the selected model architecture. LSTM-based components may process sequential dependencies, CNN-based components may identify local and hierarchical patterns, and Transformer-based components may analyze broader contextual relationships. The resulting classification can then be combined with conventional detection information to support a final decision.

The conceptual workflow can therefore be represented as:

Network Traffic → Preprocessing → Signature-Based Detection → AI-Based Analysis → Decision Fusion → Attack Classification → Alert/Response

This hybrid structure does not imply that every traffic sample must be processed by every available model. Instead, the architecture can route traffic according to detection confidence, computational requirements, and operational policies. Known threats may be resolved using less computationally expensive mechanisms, whereas uncertain or anomalous traffic can be forwarded to more sophisticated analysis.

The proposed conceptual architecture is illustrated in **Figure 5**.

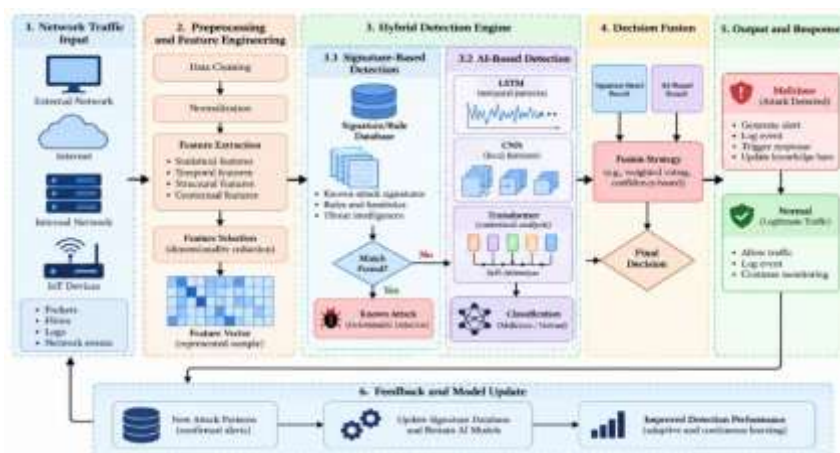


Figure 5. Hybrid architecture integrating signature-based and artificial intelligence-based mechanisms for network attack detection.

Such an architecture also provides a mechanism for incorporating feedback. Confirmed alerts and newly identified attack patterns can be added to the knowledge base or used during subsequent model retraining. This creates a detection lifecycle in which conventional rules and adaptive models can evolve together as the threat environment changes.

3.5. Discussion

The results of the comparative analysis indicate that the principal advantage of AI-based network attack detection lies in its capacity to analyze patterns that cannot be completely represented through static signatures. This is particularly relevant to unknown, obfuscated, multi-stage, and behaviorally complex attacks. Research on unknown attack detection (2,5), intelligent network attack detection (7), and transfer-learning-based detection of previously unseen attacks (8) supports the importance of adaptive mechanisms in modern intrusion detection.

At the same time, increased model complexity introduces additional operational requirements. Deep-learning architectures require appropriate data representation, sufficient computational resources, systematic validation, and continuous monitoring after deployment. Therefore, superior representational capability should not automatically be interpreted as superior operational suitability. The selection of a detection model should take into account not only its classification capability but also latency, throughput, scalability, interpretability, maintainability, and resistance to adversarial manipulation.

The comparison of LSTM, CNN, and Transformer architectures further demonstrates that model selection should be related to the characteristics of attack traces. Sequential malicious behavior favors architectures capable of representing temporal dependencies; locally structured traffic patterns may benefit from convolutional processing; and complex relationships across longer sequences can motivate the use of attention-based architectures. Consequently, the choice of model should follow the characteristics of the detection problem rather than assuming the universal superiority of a single architecture.

The analysis also highlights the importance of the validation framework described in Section 2.5. Cross-validation and holdout testing provide mechanisms for evaluating generalization, while confusion-matrix-derived indicators distinguish between different types of classification errors. For operational cybersecurity, this distinction is essential because false positives and false negatives have different consequences: the former increase analyst workload, whereas the latter represent malicious activity that may remain undetected.

Overall, the findings support a **hybrid and adaptive detection framework** in which signature-based mechanisms, anomaly analysis, and AI models perform complementary roles. Such integration provides a more comprehensive conceptual approach to network attack detection than exclusive reliance on a single detection paradigm. However, the present analysis is comparative and methodological; the current manuscript does **not** provide a new controlled experiment demonstrating that one of the examined LSTM, CNN, or Transformer models statistically outperforms the others. Consequently, the comparative conclusions should be interpreted in terms of architectural characteristics, reported applicability, and deployment considerations rather than as a new numerical ranking of model performance.

4. CONCLUSION

This study presented a systematic comparative analysis of traditional and artificial intelligence-based models and algorithms for detecting network attack traces in computer networks. The analysis considered signature-based, anomaly-based, and hybrid detection paradigms together with LSTM, CNN, and Transformer architectures, benchmark datasets, model-validation procedures, and operational requirements for intelligent intrusion detection systems.

The comparative analysis showed that conventional signature-based IDS/IPS mechanisms remain particularly applicable to previously identified attacks because they provide deterministic, interpretable, and relatively computationally efficient detection. However, their dependence on predefined signatures limits their applicability to previously unseen, obfuscated, or substantially modified malicious behavior. Research on unknown network attack detection and adaptive learning approaches demonstrates the importance of mechanisms that can operate beyond fixed attack signatures (2,5,8).

Anomaly-based and AI-based approaches address this limitation by analyzing deviations from normal network behavior and learning discriminative traffic characteristics. Nevertheless, their effectiveness is strongly influenced by the quality and representativeness of training data, class distribution, model architecture, and changes in the operational environment. Therefore, the performance of intelligent detection systems should be evaluated using complementary criteria, including accuracy, precision, recall, F1-score, and AUC-ROC, rather than relying on accuracy alone.

The comparison of deep-learning architectures further demonstrated that **LSTM, CNN, and Transformer models provide complementary analytical capabilities**. LSTM architectures are particularly relevant to temporal dependencies and sequential attack behavior; CNN models are suitable for extracting local and hierarchical traffic characteristics; and Transformer architectures provide mechanisms for modeling broader contextual relationships through self-attention. Consequently, model selection should be determined by the characteristics of the network traffic and attack traces rather than by assuming the universal superiority of a particular architecture.

The analysis also demonstrated that successful deployment of AI-based intrusion detection requires consideration beyond classification performance. **Latency, throughput, scalability, model drift, interpretability, adversarial robustness, and integration with existing security infrastructure** represent important operational requirements. A model that demonstrates strong performance on benchmark data may still require adaptation before deployment in dynamic, high-volume network environments.

Based on the comparative findings, a **hybrid detection architecture** integrating signature-based and AI-based mechanisms provides a suitable conceptual direction for network attack detection. In this architecture, known malicious patterns can be processed through deterministic signature and rule-based mechanisms, while uncertain, anomalous, or previously unseen traffic can be subjected to AI-based analysis. Decision fusion and continuous feedback can subsequently support alert generation, knowledge-base updating, and model retraining.

Overall, the study demonstrates that effective detection of modern network attack traces requires the integration of **deterministic detection, adaptive learning, systematic validation, and continuous model updating**. Such an integrated approach provides a methodological basis for developing more adaptive, scalable, and operationally applicable intrusion detection systems in dynamically evolving network environments.

Key limitation

A limitation of the present study is that the comparison is primarily **analytical and methodological**. Although benchmark datasets and standard evaluation criteria are considered, the current study does not report a unified controlled experiment in which all examined LSTM, CNN, and Transformer architectures are trained and tested under identical experimental conditions. Therefore, the comparative findings characterize their functional capabilities and deployment considerations rather than establishing a numerical performance ranking among the models.

ACKNOWLEDGEMENTS

The authors acknowledge the institutional and academic support provided by the Tashkent University of Information Technologies named after Muhammad al-Khwarizmi and the Cybersecurity Center, State Institution, Uzbekistan, during the preparation of this study.

AUTHOR CONTRIBUTIONS

Madina S. Ramazonova: Conceptualization, Methodology, Investigation, Literature Review, Formal Analysis, Visualization, Writing—Original Draft.

Sherzod R. Gulomov: Conceptualization, Methodology, Supervision, Validation, Writing—Review & Editing.

Mashhura R. Rakhmankulova: Investigation, Literature Review, Formal Analysis, Validation, Writing—Review & Editing.

All authors reviewed and approved the final version of the manuscript.

CONFLICT OF INTEREST

The authors declare that there are no conflicts of interest regarding the publication of this study.

ETHICS STATEMENT

Ethical approval was not required for this study because the research did not involve human participants, animals, personal data, clinical interventions, or biological samples. The study was based on the comparative and methodological analysis of network attack detection approaches and publicly described benchmark datasets.

DATA AVAILABILITY STATEMENT

No new primary dataset was generated or collected specifically for this study. The analysis considered publicly established benchmark datasets, including CICIDS2017, UNSW-NB15, NSL-KDD, and TON_IoT, within the methodological framework for evaluating network attack detection approaches. Information concerning these datasets is available through their respective original sources and associated publications.

REFERENCES

- [1] Baba, T., Matsuda, S., 2002. Tracing network attacks to their sources. *IEEE Internet Computing* 6(2), 20–26. <https://doi.org/10.1109/4236.991439>.
- [2] Rieck, K., Laskov, P., 2007. Language models for detection of unknown attacks in network traffic. *Journal in Computer Virology* 2(4), 243–256. <https://doi.org/10.1007/s11416-006-0030-0>.
- [3] Mahoney, M.V., Chan, P.K., 2002. Learning nonstationary models of normal network traffic for detecting novel attacks. In: *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 376–385. <https://doi.org/10.1145/775047.775102>
- [4] Camtepe, S.A., Yener, B., 2007. Modeling and detection of complex attacks. In: *Third International Conference on Security and Privacy in Communications Networks and the Workshops (SecureComm 2007)*, pp. 234–243. IEEE. <https://doi.org/10.1109/SECCOM.2007.4550338>.
- [5] Rieck, K., Laskov, P., 2006. Detecting unknown network attacks using language models. In: *International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment*, pp. 74–90. Springer, Berlin, Heidelberg.
- [6] Cabrera, J.B.D., Ravichandran, B., Mehra, R.K., 2000. Statistical traffic modeling for network intrusion detection. In: *Proceedings of the 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, pp. 466–473. IEEE.
- [7] Aljabri, M., Aljameel, S.S., Mohammad, R.M.A., Almotiri, S.H., Mirza, S., Anis, F.M., Aboulmour, M., Alomari, D.M., Alhamed, D.H., Altamimi, H.S., 2021. Intelligent techniques for detecting network attacks: Review and research directions. *Sensors* 21(21), 7070. <https://doi.org/10.3390/s21217070>
- [8] Zhao, J., Shetty, S., Pan, J.W., Kamhoua, C., Kwiat, K., 2019. Transfer learning for detecting unknown network attacks. *EURASIP Journal on Information Security* 2019, Article 1. <https://doi.org/10.1186/s13635-019-0084-4>
- [9] Prasad, K.M., Reddy, A.R.M., Rao, K.V., 2014. DoS and DDoS attacks: Defense, detection and traceback mechanisms—A survey. *Global Journal of Computer Science and Technology* 14(7), 15–32.
- [10] Elsayed, M.S., Le-Khac, N.A., Dev, S., Jurcut, A.D., 2020. DDoSNet: A deep-learning model for detecting network attacks. In: *2020 IEEE 21st International Symposium on “A World of Wireless, Mobile and Multimedia Networks” (WoWMoM)*, pp. 391–396. IEEE.
- [11] Thatte, G., Mitra, U., Heidemann, J., 2008. Detection of low-rate attacks in computer networks. In: *Proceedings of the 11th IEEE Global Internet Symposium*, pp. 1–6. <https://doi.org/10.1109/INFOCOM.2008.4544638>
- [12] Yan, Q., Wang, M., Huang, W., Luo, X., Yu, F.R., 2019. Automatically synthesizing DoS attack traces using generative adversarial networks. *International Journal of Machine Learning and Cybernetics* 10(12), 3387–3396.
- [13] Nadeem, A., Howarth, M.P., 2013. A survey of MANET intrusion detection & prevention approaches for network layer attacks. *IEEE Communications Surveys & Tutorials* 15(4), 2027–2045.
- [14] Wu, Y., Wei, D., Feng, J., 2020. Network attacks detection methods based on deep learning techniques: A survey. *Security and Communication Networks* 2020, 8872923. <https://doi.org/10.1155/2020/8872923>
- [15] Tartakovsky, A.G., Polunchenko, A.S., Sokolov, G., 2013. Efficient computer network anomaly detection by changepoint detection methods. *IEEE Journal of Selected Topics in Signal Processing* 7(1), 4–11. <https://doi.org/10.1109/JSTSP.2012.2233713>.

- [16] Maheshwari, R., Gao, J., Das, S.R., 2007. Detecting wormhole attacks in wireless networks using connectivity information. In: *IEEE INFOCOM 2007—26th IEEE International Conference on Computer Communications*, pp. 107–115. IEEE.
- [17] Ourston, D., Matzner, S., Stump, W., Hopkins, B., 2003. Applications of hidden Markov models to detecting multi-stage network attacks. In: *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*, pp. 10. IEEE.
- [18] Nezhad, S.M.T., Nazari, M., Gharavol, E.A., 2016. A novel DoS and DDoS attacks detection algorithm using ARIMA time series model and chaotic system in computer networks. *IEEE Communications Letters* 20(4), 700–703.
- [19] Patcha, A., Park, J.M., 2007. An overview of anomaly detection techniques: Existing solutions and latest technological trends. *Computer Networks* 51(12), 3448–3470.
- [20] Saheed, Y.K., Abiodun, A.I., Misra, S., Holone, M.K., Colomo-Palacios, R., 2022. A machine learning-based intrusion detection for detecting Internet of Things network attacks. *Alexandria Engineering Journal* 61(12), 9395–9409. <https://doi.org/10.1016/j.aej.2022.02.063>
- [21] Pan, Y., Sun, F., Teng, Z., White, J., Schmidt, D.C., Staples, J., Krause, L., 2019. Detecting web attacks with end-to-end deep learning. *Journal of Internet Services and Applications* 10, 16. <https://doi.org/10.1186/s13174-019-0115-x>
- [22] Zhang, H., Huang, L., Wu, C.Q., Li, Z., 2020. An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset. *Computer Networks* 177, 107315. <https://doi.org/10.1016/j.comnet.2020.107315>
- [23] Ahmad, I., Ul Haq, Q.E., Imran, M., Alassafi, M.O., AlGhamdi, R.A., 2022. An efficient network intrusion detection and classification system. *Mathematics* 10(3), 530. <https://doi.org/10.3390/math10030530>
- [24] Tartakovsky, A.G., 2014. Rapid detection of attacks in computer networks by quickest changepoint detection methods. In: *Data Analysis for Network Cyber-Security*, pp. 33–70.
- [25] Bowman, B., Laprade, C., Ji, Y., Huang, H.H., 2020. Detecting lateral movement in enterprise computer networks with unsupervised graph AI. In: *23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020)*, pp. 257–268.
- [26] Injadat, M., Salo, F., Nassif, A.B., Essex, A., Shami, A., 2018. Bayesian optimization with machine learning algorithms towards anomaly detection. In: *2018 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6. IEEE.
- [27] Stabili, D., Marchetti, M., Colajanni, M., 2017. Detecting attacks to internal vehicle networks through Hamming distance. In: *2017 AEIT International Annual Conference*, pp. 1–6. IEEE.
- [28] Feinstein, L., Schnackenberg, D., Balupari, R., Kindred, D., 2003. Statistical approaches to DDoS attack detection and response. In: *Proceedings of the DARPA Information Survivability Conference and Exposition*, Vol. 1.