Journal of Computer Science and Electrical Engineering
Print ISSN: 2663-1938
Online ISSN: 2663-1946
DOI: https://doi.org/10.61784/jcsee3136

DEEP LEARNING-BASED MALICIOUS TRAFFIC ANALYSIS: A COMPREHENSIVE SURVEY

WenCai He[1], ZhiJie Peng[2], MingShen Zhang[1], He Zhu[1*]
[1]*College of Cyber Security, Tarim University, Aral 843300, Xinjiang, China.*
[2]*School of Physics and Electronics, Changsha University of Science and Technology, Changsha 410114, Hunan, China.*
[*]*Corresponding Authors: He Zhu*

Abstract: Malicious traffic analysis has become increasingly challenging due to encryption-by-default communication, evolving attack behaviors, and distribution shifts across heterogeneous environments. At present, many studies still do not share consistent pipeline designs, traffic representations, benchmark datasets, or assessment methods. Based on this problem, this paper systematically examines deep-learning-based malicious-traffic-analysis technologies. Existing works are organized into four main parts: data acquisition and preprocessing, traffic representation, learning models, and performance evaluation methods. In particular, we compare representative methods, including traditional machine learning, deep learning, and hybrid forms, in terms of detection accuracy, computational cost, cross-scenario generalization ability, and suitable metrics. We also identify key problems in practical applications, such as encrypted traffic monitoring capability, biased datasets, and adversarial attacks, and summarize related research paths. This paper introduces the research area and provides a reference for reproducible literature-based evaluation design.
Keywords: Malicious traffic analysis; Deep learning; Encrypted traffic detection; Traffic representation learning; Network security

1 INTRODUCTION

With the rapid development of cloud computing, the Internet of Things (IoT), and 5G/6G mobile networks, cyberspace has evolved into a highly open, heterogeneous, and dynamic environment [1-2]. Meanwhile, cyber attacks have become increasingly large-scale, stealthy, and automated [3-4]. Malicious network traffic, which represents the communication-level manifestation of cyber attacks, plays a central role in intrusion detection systems (IDS), network situational awareness, and threat hunting platforms [5-6]. From early coarse-grained attacks such as port scanning and worm propagation to denial-of-service attacks [7-8], and then to today's sophisticated threats relying on encrypted channels, botnets, and advanced persistent threats (APTs), malicious traffic has exhibited growing complexity in communication patterns, behavioral characteristics, and adversarial strategies. As a result, traditional detection paradigms based on rule matching or deep packet inspection (DPI) are facing fundamental limitations [9-11]. In recent years, the widespread deployment of encryption protocols such as Hypertext Transfer Protocol Secure (HTTPS), Transport Layer Security (TLS) 1.3 [12], and Quick UDP Internet Connections (QUIC) has significantly enhanced communication privacy and security [13], but at the same time has drastically reduced the visibility of network payloads. Attackers can hide malicious activities within encrypted tunnels or deliberately mimic the statistical properties of benign traffic, making malicious and benign flows highly similar at the metadata level [14-16]. Consequently, detection approaches that rely on payload inspection or handcrafted signatures are becoming increasingly ineffective. Under this background, how to accurately identify malicious traffic without decrypting communication contents, using only flow metadata, protocol interaction patterns, and behavioral features, has become a critical research problem in network security [17-20].

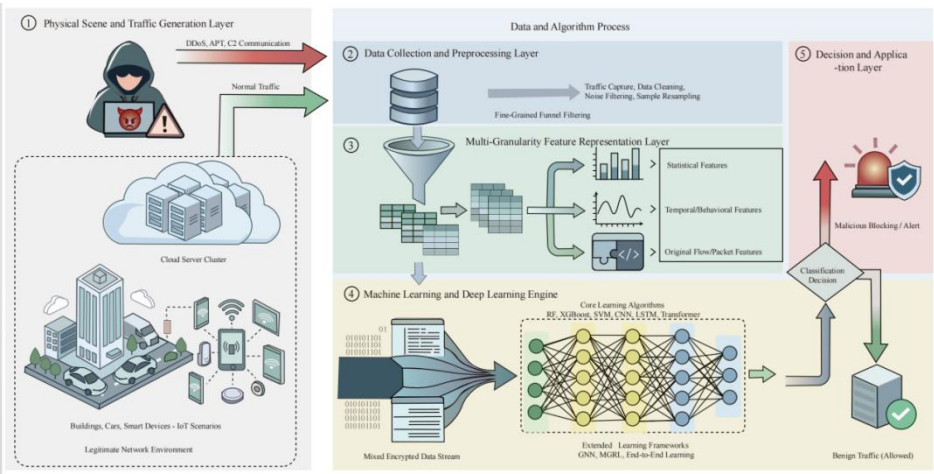

Figure 1 Overall Framework of Deep Learning-Based Malicious Traffic Analysis

Machine learning (ML) [21], especially deep learning (DL) [22], provides a promising paradigm to address these challenges. Compared with traditional methods that heavily depend on expert-designed features, ML-based approaches are capable of automatically learning discriminative representations from large-scale traffic data, enabling a shift from manual feature engineering to data-driven and end-to-end representation learning [23-25]. In recent years, substantial progress has been made in tasks such as encrypted traffic classification [26-27], malware communication detection [28], botnet identification [29-30], and network anomaly detection [31-34]. As illustrated in Figure 1, these studies have gradually formed a technical ecosystem covering data acquisition and preprocessing, traffic representation and feature modeling, learning model design and optimization, as well as system deployment and evaluation.

Despite these advances, deploying ML-based malicious traffic analysis systems in real-world networks still faces several fundamental challenges. On the one hand, public datasets and real-world traffic often differ significantly in protocol distributions, attack patterns, and background noise, which leads to severe performance degradation when models are transferred across scenarios [35-36]. On the other hand, although deep models exhibit strong representation capabilities, they often suffer from high computational overhead, limited interpretability, and insufficient robustness against adversarial manipulation [37-38]. In particular, when facing adversarially crafted traffic designed to evade detection, the security and stability of existing models remain far from satisfactory. Moreover, current studies differ substantially in data processing pipelines, feature representations, and evaluation protocols, which hinders fair comparison and reproducibility, and further complicates practical deployment.

Several survey papers have reviewed machine learning techniques for network traffic classification or intrusion detection [23-31, 33-37, 39]. However, existing reviews predominantly adopt a literature-listing approach, simply categorizing studies by the underlying algorithms—such as detailing which models employ Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), or Transformers—rather than critically evaluating them through a problem-oriented lens. For instance, aiming at the distribution shift problem in encrypted traffic analysis, current studies often propose complex deep learning frameworks, yet these frequently exhibit severe generalization defects across heterogeneous environments; similarly, regarding adversarial attacks, existing literature heavily focuses on algorithmic robustness but largely ignores the strict latency and resource constraints of real-world deployment. To bridge this gap, this survey transcends mere algorithmic categorization. Through a critical comparative analysis, we reveal the fundamental trade-offs these methods entail in practical deployment, systematically examining the tensions between accuracy, robustness, interpretability, and computational cost.

In this paper, we present a comprehensive survey of recent advances in malicious traffic analysis based on machine learning. Instead of proposing a new detection model, we aim to organize existing research efforts into a coherent framework and provide a systematic understanding of the design space. Specifically, the main contributions of this survey are summarized as follows:

• Unified and structured analytical framework. We use a unified end-to-end framework for machine learning-based malicious traffic analysis, covering data acquisition, preprocessing, traffic representation, model design, and system deployment. This framework provides a systematic perspective to organize fragmented research efforts and serves as the backbone for analyzing the full design pipeline.

• Problem-oriented review of traffic representation and data processing. Instead of simply listing existing methods, we systematically review data acquisition, preprocessing, and traffic representation techniques from a problem-driven perspective, including flow-level statistics, temporal patterns, and raw traffic representations. We further analyze how different design choices affect model performance under practical constraints such as encrypted traffic and limited observability.

• Comprehensive comparison of learning models with deployment considerations. We provide a structured comparison of traditional machine learning and deep learning models in malicious traffic analysis, evaluating them not only in terms of detection performance, but also computational complexity, scalability, and deployment cost. This enables a clearer understanding of the trade-offs involved in model selection for real-world systems.

• Critical analysis of key challenges and trade-offs. We identify and analyze fundamental challenges in practical deployment, including encrypted traffic analysis, cross-scenario generalization, and adversarial robustness. More importantly, we highlight the inherent trade-offs among accuracy, robustness, interpretability, and efficiency, which are often overlooked in existing studies.

• Forward-looking research directions. Based on the above analysis, we outline several promising research directions, including generalization-centric learning, robust detection, and efficient deployment, aiming to bridge the gap between academic research and real-world applications.

The objective of this survey is to provide a structured, comparative, and critical overview of the field from the perspectives of data, representation, models, and security constraints, thereby offering useful guidance for future research on model design, dataset construction, and real-world system deployment.

The remainder of this paper is organized as follows. Section II introduces the problem definition and the general processing pipeline of malicious traffic analysis. Section III reviews data acquisition and preprocessing methods. Section IV summarizes traffic representation and feature modeling techniques. Section V surveys different categories of learning models. Section VI outlines open issues and future research directions. Finally, Section VII concludes the paper.

2 GENERAL FRAMEWORK OF MACHINE LEARNING-BASED MALICIOUS TRAFFIC ANALYSIS

Machine learning-based malicious traffic analysis is generally formulated as a data-driven classification or detection problem, where network traffic collected from monitored environments is processed, represented, and fed into learning models to determine whether a given traffic instance is benign or malicious. Although existing studies differ in data sources, feature representations, and model architectures, most of them follow a similar processing pipeline. Establishing a unified framework is therefore essential for organizing the research landscape and for clarifying the relationships among different technical components. Recent surveys and empirical studies have shown that detection performance is jointly determined by data quality, feature representation, and model design, rather than by the learning algorithm alone [8, 40-43].

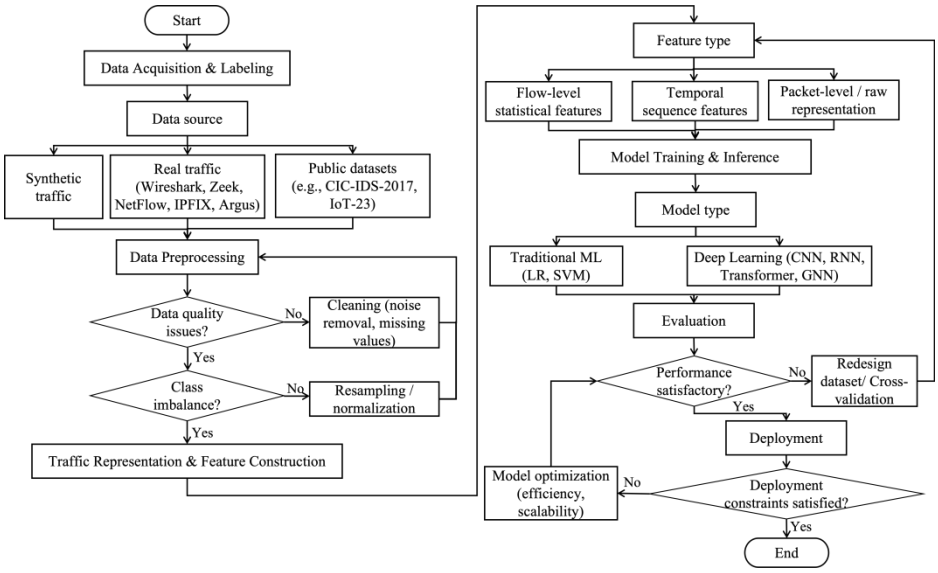

Figure 2 Processing Pipeline of ML-Based Malicious Traffic Analysis

2.1 Problem Definition

In a typical setting, malicious traffic analysis aims to infer the security status of network communications based on observable traffic data. Depending on the application scenario, the analysis granularity may range from packet-level and flow-level samples to session-level or host-level behaviors. Formally, given a set of traffic samples collected from real networks, testbeds, or public datasets, each sample is associated with a label indicating its class, such as benign or malicious, or a more fine-grained attack category. The learning objective is to train a model that can generalize to unseen traffic while maintaining acceptable computational cost and stability in realistic environments.

In practice, several factors significantly complicate this task. First, the increasing adoption of encryption protocols forces many detection systems to rely mainly on metadata and statistical characteristics rather than payload content, which has been widely studied in encrypted traffic classification and malware traffic detection [12, 15-16, 19]. Second, real-world traffic data are typically highly imbalanced, with malicious samples being much rarer than benign ones, which has been shown to degrade the reliability of conventional evaluation and training strategies [5]. Third, network environments and attack patterns evolve over time, leading to performance degradation when models trained on one dataset are deployed in different scenarios, as discussed in recent measurement-driven studies [6, 21].

2.2 Overall Processing Pipeline

Figure 2 outlines a typical processing pipeline of machine learning-based malicious traffic analysis. Although specific implementations vary across different studies, the overall workflow can be generally divided into five main stages: data acquisition and labeling, data preprocessing, traffic representation and feature construction, model training and inference, and evaluation and deployment.

At the data acquisition stage, traffic is collected from real networks, simulated environments, or public datasets. Commonly used tools and frameworks, such as Wireshark, Zeek, NetFlow, Internet Protocol Flow5 Information Export (IPFIX), and Argus, have been widely adopted to extract traffic records and metadata for subsequent analysis [24-27], [30]. In addition to real traffic collection, synthetic traffic generation and publicly available datasets, such as CIC-IDS-2017, IoT-23, and CIRA-CIC-DoHBrw-2020, are frequently used to support controlled experiments and benchmarking [19, 22].

After data collection, preprocessing is usually required to address issues such as noise, redundancy, missing values, and class imbalance. Recent studies have shown that appropriate sampling, normalization, and resampling strategies can significantly improve the stability and reliability of learning-based detectors, especially under highly imbalanced data distributions.

The next stage is traffic representation and feature construction, which determines how network communications are described and provided as input to learning models. Existing works have explored flow-level statistical features,

temporal features derived from packet sequences, as well as packet-level or raw traffic representations for end-to-end learning [21]. The choice of representation directly affects the trade-off between information richness, computational cost, and robustness to encryption.

Based on the constructed representations, various learning models can be trained to perform detection or classification. Traditional machine learning models, such as logistic regression and support vector machines, have been widely used due to their efficiency and interpretability, especially in metadata-based and resource-constrained scenarios [36-37]. More recently, deep learning models, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), gated recurrent units (GRUs), the Transformer architecture, and graph neural networks (GNNs), have demonstrated strong capabilities in automatic feature learning and complex pattern modeling, particularly for encrypted and large-scale traffic analysis [44-46].

Finally, trained models are evaluated using appropriate performance metrics and validation protocols. Prior studies have pointed out that evaluation results can be strongly biased by dataset composition, class imbalance, and experimental settings, which calls for careful design of benchmarking methodologies and cross-scenario testing [10, 12-14, 22, 24]. In addition, practical deployment must consider computational efficiency, scalability, and robustness against evasion or adversarial behaviors, which remain open challenges in real-world systems [46-48].

The above framework provides a unified view of machine learning-based malicious traffic analysis from data collection to model deployment. It also serves as the organizational backbone of this survey. The details of each component are presented in the following sections.

3 DATA ACQUISITION AND PREPROCESSING FOR MALICIOUS TRAFFIC ANALYSIS

High-quality data constitute the foundation of machine learning-based malicious traffic analysis, as the reliability, generalization capability, and practical value of detection models are largely constrained by the representativeness and quality of the training data. Existing studies typically construct datasets through three complementary approaches, namely real-world traffic collection, synthetic traffic generation, and publicly available benchmark datasets.

3.1 Real-World Traffic Collection

Real-world traffic collection aims to capture network behaviors in operational environments and thus provides data with high realism and practical relevance. A variety of tools have been developed to support traffic capture and feature extraction at different levels of the protocol stack.

Low-level packet capture tools, such as Tcpdump and Wireshark, are widely used to obtain raw packet traces, enabling fine-grained inspection of protocol fields and payload structures. Bhuyan et al. [8] pointed out that such tools provide essential data support for payload-based anomaly detection and protocol analysis. In the context of encrypted traffic, metadata-oriented monitoring systems become more important. Zeek generates structured logs (e.g., conn.log and http.log) to describe connection states and application layer behaviors. Prior work on encrypted traffic analysis and metadata-based monitoring highlights the value of session logs and TLS-related handshake features for classification and forensics without payload decryption [12, 14].

At the flow level, NetFlow and its standardized successor IPFIX convert packet streams into compact flow records, offering improved scalability and flexibility for large-scale network monitoring [14, 27]. Similarly, Argus focuses on flow feature extraction and supports customizable feature extensions. The UNSW-NB15 benchmark dataset, widely used in intrusion detection research, is constructed from Argus derived flow records and has been adopted in numerous feature-learning and evaluation studies [24]. Overall, real-world traffic collection provides highly realistic data; however, it is often constrained by privacy concerns, labeling costs, and limited coverage of rare or emerging attack patterns.

3.2 Synthetic Traffic Generation

To alleviate the scarcity of labeled malicious traffic and to improve scenario coverage, many studies rely on synthetic traffic generation. Existing approaches can be roughly categorized into script-based generation, traffic generator-based synthesis, and virtual network simulation frameworks.

The widely used CIC-IDS-2017 dataset was introduced with script-based traffic generation to simulate diverse benign and malicious behaviors in a controlled environment [52]. The CIRA-CIC-DoHBrw-2020 dataset [19], in contrast, focuses on DNS-over-HTTPS (DoH) scenarios and employs dedicated collection tools to reproduce encrypted tunneling traffic patterns. Complementary studies also discuss large-scale traffic generation, hybrid deep learning pipelines for intrusion detection, and broader surveys of learning-based cybersecurity methods that motivate standardized benchmarking [15, 20-22].

Despite their flexibility and scalability, synthetic datasets inevitably suffer from intrinsic limitations. In particular, discrepancies between synthetic and real-world traffic distributions may lead to feature shift and degraded model generalization. Moreover, overly simplified generation rules may introduce unrealistic or biased patterns, thereby reducing the credibility of evaluation results [49-51]. Consequently, synthetic traffic is often used as a complement rather than a replacement for real-world data.

3.3 Public Benchmark Datasets

Publicly available datasets play a crucial role in enabling fair comparison and reproducible evaluation of malicious traffic detection methods. Among existing benchmarks, CIC-IDS-2017, IoT-23, and CIRACIC-DoHBrw-2020 are three representative datasets that have been extensively adopted in recent studies [52-54].

CIC-IDS-2017 provides comprehensive coverage of multiple attack types and is suitable for general-purpose intrusion detection evaluation [52]. However, its protocol versions are relatively outdated, which may cause feature drift when models are deployed in modern encrypted environments. IoT-23 is specifically designed for IoT scenarios and offers realistic device-generated traffic traces [22, 53], yet its applicability is limited by the relatively narrow diversity of device types. CIRA-CIC-DoHBrw-2020 focuses on DoH traffic and achieves fine-grained protocol-level characterization [19, 54], but its severe class imbalance constrains cross-scenario generalization performance. The comparison of public datasets is shown in Table 1.

These observations indicate that dataset quality directly determines the upper bound of achievable detection performance. Therefore, researchers are encouraged to carefully balance dataset scale, class distribution, and protocol coverage, and to incorporate data augmentation and cross-domain transfer techniques to mitigate the inherent limitations of individual datasets [26, 52-53].

3.4 Data Preprocessing and Imbalance Handling

Preprocessing plays a pivotal role in improving data quality and stabilizing model training. Typical preprocessing steps include noise removal, normalization, feature scaling, and redundancy reduction, which aim to enhance data consistency and reduce unnecessary computational overhead. Donkol et al. demonstrated that hybrid optimization-based preprocessing pipelines can significantly improve detection performance by jointly refining data quality and feature representation [4]. Along this line, a multistage preprocessing strategy, including data cleaning, length normalization, and feature dimensionality reduction, has been widely adopted in recent studies. For instance, Yang and Shami studied transfer learning with optimized CNNs for vehicular intrusion detection, where careful sample filtering and representation learning are critical under noisy telemetry [25]. Separately, feature analyses on CICIDS-2017 and related benchmarks show that information gain–based ranking combined with dimensionality reduction can substantially shrink feature sets while preserving detection accuracy [19, 26].

Table 1 Comparison of Representative Public Datasets for Encrypted Malicious Traffic Detection

Category / Metric	CIC-IDS-2017	IoT-23	CIRA-CIC-DoHBrw-2020
Basic Information			
1. Release year	2017	2020	2020
2. Primary focus	General intrusion detection	IoT botnet and malicious device traffic	DNS-over-HTTPS traffic analysis
3. Traffic source	Script-generated mixed attack scenarios	Realistic IoT device network traffic	Protocol-specific DoH browsing flows
Traffic Characteristics			
1. Encrypted traffic	Yes	Partial	Yes
2. Class balance	Imbalanced	Imbalanced	Imbalanced
3. Attack diversity	High①	Moderate	Low to moderate
4. Scenario specificity	General-purpose	IoT-oriented②	DoH-specific③
Strengths and Limitations			
1. Main strength	Broad attack coverage①	Realistic IoT environment②	High relevance to encrypted DoH analysis ③
2. Main limitation	Protocol aging and limited realism in some flows	Limited device and attack-type variety	Narrow application scope and strong class imbalance
Recommended Usage			
1. Best suited for	Benchmarking general encrypted threat detection	Evaluating IoT security monitoring models	Studying DoH-aware detection and classification
2. Research value	Widely used baseline④	Strong domain realism for IoT	High value for protocol-specific encrypted traffic research

Note: ① Broad attack diversity; ② strong IoT scenario realism; ③ high relevance to encrypted traffic analysis; ④ widely adopted benchmark value

Class imbalance is another persistent challenge in malicious traffic datasets, where benign samples usually dominate and minority attack classes are severely underrepresented. Abdelkhalek et al. addressed this issue by adopting hybrid resampling strategies and achieved notable improvements in minority-class recognition performance [5]. Existing solutions generally follow two complementary directions, namely datalevel augmentation and algorithm-level optimization. On the data side, the Synthetic Minority Oversampling Technique (SMOTE) and its variants improve

class distribution by synthesizing minority samples [27], whereas generative models, such as generative adversarial networks (GANs) and diffusion models, further enhance data diversity by learning deep feature representations and producing more realistic synthetic samples [28]. On the algorithmic side, focal loss and cost-sensitive learning alleviate classification bias by modifying the loss function and penalty mechanisms [29]. Hybrid machine–deep learning pipelines have also been shown to improve robustness under strong class imbalance on contemporary intrusion detection benchmarks [22], while systematic IDS surveys further emphasize careful metric selection when positives are rare [29].

In addition to algorithmic performance, the reliability of evaluation under imbalanced settings has also attracted increasing attention. Karatas et al. [9] showed that conventional evaluation metrics may become misleading when class distributions are highly skewed, highlighting the necessity of carefully designed preprocessing pipelines and evaluation protocols. Furthermore, Chen et al. analyzed distributed denial of service (DDoS) detection in software-defined networking (SDN)-based cloud settings and highlighted how dataset construction and class definitions affect reported accuracy [30], underscoring the need for standardized benchmarking. Data acquisition strategies and preprocessing techniques jointly shape the effectiveness and reliability of machine learning-based malicious traffic analysis. A systematic understanding of their strengths and limitations is therefore indispensable for both fair benchmarking and practical deployment.

4 TRAFFIC REPRESENTATION AND FEATURE MODELING FOR MALICIOUS TRAFFIC ANALYSIS

Traffic representation and feature modeling form the critical interface between raw network traffic and learning algorithms. The effectiveness of malicious traffic analysis systems largely depends on how traffic is abstracted into discriminative representations. According to existing studies, current approaches can be broadly categorized into statistical feature-based representations, temporal and behavioral features, raw traffic representations, and frequency-domain or multi-granularity feature modeling schemes [13].

4.1 Statistical Feature-Based Representations

Statistical features have been widely adopted in malicious traffic analysis due to their high computational efficiency and strong interpretability, especially under encrypted traffic scenarios where payload contents are inaccessible. These features typically summarize flows using metrics such as packet length statistics, flow duration, byte counts, and protocol field distributions. Gamage and Samarabandu surveyed deep learning methods for network intrusion detection and discussed how statistical and engineered features interact with modern classifiers [31]. Moreover, Pacheco et al. established the dominant role of statistical features in machine learning-based traffic classification workflows [3], demonstrating their practical effectiveness in large-scale detection systems. Anderson and McGrew further validated that statistical features remain effective for encrypted traffic classification when combined with appropriate learning models [6]. Despite their efficiency and interpretability, statistical features usually rely on handcrafted design and domain expertise. They are also sensitive to protocol evolution and traffic distribution shifts, which may lead to performance degradation in cross-scenario deployments.

4.2 Temporal and Behavioral Feature Modeling

To capture the dynamic characteristics of network traffic, temporal and behavioral feature modeling has been introduced to complement static statistical summaries. These methods preserve sequential information, such as packet arrival patterns, burst behaviors, and temporal correlations, which are particularly useful for detecting stealthy or low-rate attacks. Surveys on deep learning for intrusion detection highlight temporal modeling—including recurrent architectures and sequence-level representations—as a key complement to static flow statistics. In addition, the work in [6] demonstrated that temporal features, when combined with statistical features, can significantly enhance the robustness of encrypted traffic classification. These results indicate that temporal and behavioral representations provide richer contextual information than purely static features. However, such methods usually incur higher computational and storage overhead, and their performance may be sensitive to traffic truncation or sampling strategies adopted in real-world monitoring systems.

4.3 Raw Traffic Representations

With the rapid development of deep learning, an increasing number of studies have shifted from handcrafted feature engineering to raw traffic-based representations, enabling end-to-end learning frameworks. In this paradigm, packet byte streams or packet sequences are directly fed into neural networks, which automatically learn hierarchical feature representations. Ferrag et al. systematically compared deep learning approaches for intrusion detection [28], highlighting representation learning from complex traffic artifacts. Vinayakumar et al. demonstrated strong performance using deep architectures for intelligent intrusion detection [32], while Alsaedi et al. introduced large-scale IoT telemetry resources that are widely used to train and evaluate learning-based detectors [34]. End-to-end learning paradigms further reduce reliance on manual feature engineering in encrypted traffic settings [6]. Although raw traffic representations benefit from powerful representation learning capabilities, they usually require higher computational

resources and large-scale labeled datasets, and their interpretability remains limited, which poses challenges for practical deployment in high-speed networks.

4.4 Frequency-Domain and Multi-Granularity Feature Modeling

Beyond time-domain and raw representations, frequency-domain features and multi-granularity modeling strategies have also been explored to enhance robustness and generalization. Frequency-domain features transform traffic signals using Fourier analysis to capture periodic patterns and hidden spectral characteristics. Foundational ensemble methods such as random forests remain widely used baselines for noisy, heterogeneous security datasets [35], while systematic surveys summarize a broad spectrum of machine learning and deep learning techniques applied to cybersecurity problems [36]. Moreover, recent studies suggest that combining multiple feature types, such as statistical, temporal, and raw traffic features, can effectively improve detection robustness against traffic distribution shifts and adversarial behaviors [25, 36]. This multi-granularity and multi-feature fusion strategy provides a promising direction for building more resilient malicious traffic analysis systems. Nevertheless, such hybrid representations inevitably increase model complexity and computational overhead, making efficient feature fusion and adaptive representation learning important open research problems.

Table 2 Comparison of Representative Feature Selection Methods in Malicious Traffic Analysis

Category	Ref.	Core idea	Dataset(s)	Key result	Typical application scenarios
Filter-based	[36] (MI- RFFI)	Rank features via Mutual Information+RF importance	CIC-IDS-2017; custom DDoS	Acc.99.0% (DDoS); redundancy reduced	Fast feature screening; resource-constrained IDS
Wrapper-based	[37] (ML/DL survey)	Wrapper-style selection discussed within broader ML/DL cybersecurity pipelines	Mixed IDS benchmarks (survey scope)	Summarizes common practices across tasks	Small/medium feature sets; accuracy-priority settings
Embedded	[32] (XGBoost-CNN)	Joint selection during model training (hybrid XGBoost+CNN)	UNSW-NB15; CIC-IDS-2017	Acc.98.6%	End-to-end IDS pipelines; model-coupled selection
Hybrid	[24] (IGRF-RFE)	Two-stage: Information Gain → RF-RFE	UNSW- NB15	45.0% feature reduction; +2.0% acc. vs baseline	Cross-scenario use; balance efficiency & accuracy
Hybrid (other)	[26] (PCA+IG)	principal component analysis (PCA) reduction+ Information Gain ranking	CIC-IDS- 2017	~60.0% feature reduction; acc. >98.0%	General IDS preprocessing; dimension reduction first

4.5 Feature Selection

Feature selection plays an important role in reducing feature redundancy and improving the efficiency of machine learning-based malicious traffic analysis. Existing studies generally categorize feature selection techniques into three classes: filter-based, wrapper-based, and embedded methods. Filter-based methods rely on statistical criteria to rank and select features without involving any specific classifier. Ahmad et al. combined mutual information with random forest (RF) feature importance (MI-RFFI) and achieved a detection accuracy of 99% in DDoS detection [36], demonstrating that statistical filtering can effectively preserve discriminative features while reducing redundancy. Wrapper-based methods directly associate the feature selection process with classifier performance by searching for feature subsets that optimize detection accuracy. Xin et al. reviewed machine learning and deep learning methods for cybersecurity [37], including wrapper-style selection strategies coupled with classifiers in intrusion detection pipelines. Embedded methods integrate feature selection into the model training process, enabling simultaneous learning of feature importance and model parameters. Vinayakumar et al. proposed an Extreme Gradient Boosting (XGBoost)–CNN hybrid model and reported a detection accuracy of 98.55% [32], illustrating the effectiveness of embedded feature selection in complex learning frameworks.

A performance comparison of representative feature selection methods is summarized in Table 2.

While these approaches have achieved promising results, relying on a single selection paradigm may be suboptimal. Yin et al. proposed a hybrid IGRF-RFE method that combines filter-based and wrapper based strategies [24], achieving

a 45% reduction in feature dimensionality together with a 2% improvement in accuracy on the UNSW-NB15 dataset. This suggests that hybrid feature selection schemes provide a practical trade-off between efficiency and performance. As summarized in Table 2, different feature selection paradigms exhibit distinct trade-offs among computational efficiency, detection performance, and interpretability. Hybrid methods such as IGRF-RFE [24] effectively combine the strengths of filter and wrapper approaches, achieving substantial dimensionality reduction while maintaining or improving accuracy, making them particularly suitable for deployment in evolving network environments.

The existing traffic representation methods can be categorized into statistical features, temporal and behavioral features, raw traffic representations, and frequency-domain or multi-granularity features. Statistical features offer efficiency and interpretability, temporal features enhance contextual awareness, raw traffic representations enable end-to-end learning, and hybrid modeling strategies aim to balance expressiveness and robustness. The comparative analysis of representative feature extraction techniques is summarized in Table 3, which provides a structured overview of different representation paradigms and their core characteristics.

Table 3 Comparative Analysis of Feature Extraction Techniques

Feature Type	Ref.	Granularity	Core Technique	Key Advantages	Typical Application Scenarios
Handcrafted Statistical Features	[31], [3]	Flow or Session-level	Surveyed or taxonomized statistical vs. learned features	Efficient and interpretable; widely deployed lines	Enterprise network security monitoring, interpretable encrypted traffic analysis
Temporal and Behavioral Feature	[31]	Millisecond or Second-level	Sequence or temporal DL models (e.g., RNN/attention-style pipelines)	Captures dynamics; useful for stealthy behaviors	Real-time intrusion detection; encrypted traffic with sequential structure
Raw Traffic Representation	[32]	Packet or Byte-level	End-to-end learning architecture	Eliminates manual feature engineering; adapts to encrypted scenarios	Large-scale encrypted traffic analysis, adaptive unmanned feature design
Frequency Domain and Multi-Granularity Feature	[36], [26]	Frequency Point or Band- level	Multi-representation learning and anomaly works (survey + dynamic DL)	Improves robustness via richer signal views	Challenging detection settings with distribution shift

5 LEARNING MODELS FOR MALICIOUS TRAFFIC ANALYSIS

Learning models constitute the analytical core of malicious traffic analysis systems. After traffic representation and feature construction, the learning algorithm determines detection accuracy, robustness, scalability, and deployment feasibility. Existing studies demonstrate a clear evolution from traditional machine learning classifiers to deep neural architectures, followed by hybrid frameworks and emerging adaptive paradigms designed for privacy-preserving and dynamic adversarial environments.

5.1 Traditional Machine Learning Models

Traditional machine learning models have long served as foundational tools for malicious traffic detection due to their structural simplicity, computational efficiency, and interpretability [55]. These methods rely heavily on handcrafted statistical and temporal features, such as packet length distributions, flow duration, inter-arrival times, and byte rates [33]. Consequently, they are particularly suitable for structured flow analysis, edge environments, and scenarios requiring explainable decision-making [34].

Among them, Logistic Regression (LR) remains a common baseline classifier. Its linear decision boundary and transparent probabilistic formulation enable rapid validation of feature effectiveness [35]. Early work on the KDD99 dataset reported detection accuracies around 89% for DoS attacks [8]. However, the linear modeling assumption limits its ability to capture nonlinear and encrypted traffic behaviors, leading to performance degradation in complex scenarios [36, 37].

Support Vector Machines (SVM) extend modeling capability via kernel-based nonlinear mapping [38]. Their ability to operate effectively on metadata without inspecting encrypted payloads makes them attractive for encrypted traffic classification. Anderson et al. demonstrated strong performance using RBF-kernel SVMs [6]. Subsequent

improvements focused on optimizing kernel parameters and reducing computational overhead in large-scale datasets such as CIC-IDS2017 and IoT environments [39, 40]. Nevertheless, SVM training complexity scales poorly with dataset size, restricting its applicability in high-throughput real-time systems.

Random Forest (RF) represents a widely adopted ensemble approach [41]. By aggregating multiple decorrelated decision trees, RF significantly improves generalization and robustness to noisy or incomplete traffic data. It also provides intrinsic feature importance scores, assisting security analysts in identifying key indicators of compromise. High detection accuracy has been reported for DDoS scenarios and early-stage APT behavior identification [23, 42]. However, inference latency grows with the number of trees, which may constrain deployment in millisecond-level response systems [43, 44].

Gradient Boosting Trees, including XGBoost, LightGBM, and CatBoost, further enhance predictive performance by iteratively minimizing residual errors [45]. Comparative studies across multiple intrusion detection datasets confirm their superiority in handling class imbalance and complex feature interactions [46]. Lightweight real-time detection frameworks based on XGBoost have also been proposed [47]. Despite their high accuracy, these models are sensitive to hyperparameter tuning and outliers, requiring careful validation and regularization to ensure stability [48].

Overall, traditional models remain competitive in structured and resource-constrained environments but struggle to generalize to highly nonlinear, encrypted, or evolving attack patterns.

5.2 Deep Learning Models

The increasing prevalence of encrypted communication, polymorphic malware, and multi-stage attacks has driven the adoption of deep learning models in malicious traffic analysis [49]. Unlike traditional approaches, deep architectures learn hierarchical feature representations directly from raw traffic inputs, enabling end-to-end detection without extensive manual feature engineering [50].

1) Convolutional Neural Networks (CNN): To address the severe feature obfuscation problem in encrypted environments where payload inspection fails, existing research primarily leverages CNNs to automatically extract local spatial and sequential patterns directly from raw byte streams or TLS handshake metadata [51, 52]. While this approach successfully bypasses manual feature engineering and achieves high accuracy on standardized benchmarks like CIC-IDS-2017 [31], it presents critical architectural defects when confronting sophisticated evasion tactics. Specifically, targeting stealthy command-and-control (C2) communications and low-rate attacks, the localized receptive fields of CNNs struggle to capture long range temporal dependencies, leading to high false-negative rates [54]. Furthermore, addressing the strict resource limitations of IoT security deployments, recent studies have shifted towards lightweight CNN variants [53], yet this often compromises the model's capacity to recognize complex, multi-stage threat behaviors. Consequently, deploying CNN-based architectures necessitates a strict trade-off between local feature extraction efficiency and the detection robustness of temporally dispersed adversarial traffic.

2) Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU): Aiming at the temporal evasion techniques employed by modern botnets and Advanced Persistent Threats (APTs), researchers have increasingly utilized RNN-based sequence modeling to capture the dynamic communication cycles and burst patterns of malicious flows [31]. To solve the long-term dependency decay inherent in standard RNNs, current methodologies primarily adopt LSTM and GRU architectures, utilizing their gating mechanisms to maintain memory over extended traffic sequences [53]. However, this temporal modeling capability introduces significant deployment bottlenecks. When defending against high-throughput network attacks, the inherently sequential computation graph of these models prevents effective parallelization, resulting in unacceptable inference latency and high training costs that hinder real-time edge deployment [53]. Although recent advancements have attempted to integrate attention mechanisms or adopt unsupervised autoencoder-style frameworks to improve adaptability [13, 54-58], these solutions still struggle to balance robust sequential memory with the strict computational throughput demanded by operational security centers.

3) Autoencoders (AE): To overcome the pervasive scarcity of labeled attack data and the continuous emergence of zero-day threats, researchers frequently deploy Autoencoders (AEs) and their advanced variants (e.g., Stacked AEs, Variational AEs) to establish unsupervised anomaly detection baselines [4, 51, 56]. By exclusively learning the latent representations of benign network behaviors, these models attempt to isolate malicious deviations through elevated reconstruction errors, sometimes integrating hybrid frameworks like Isolation Forests to boost baseline robustness [57]. Despite their theoretical appeal for identifying unknown attacks without prior signatures, AEs encounter severe operational bottlenecks in dynamic deployment environments. The highly variable nature of real-world benign traffic makes defining a reliable reconstruction error threshold exceptionally difficult, often resulting in unmanageable false positive alerts. More critically, due to their powerful representational capacity, overly expressive autoencoders frequently suffer from an "identity mapping" flaw, where they inadvertently reconstruct stealthy malicious traffic as normal behavior, thereby completely blinding the detection system to sophisticated intrusions [58].

4) Transformer Models: Aiming to resolve the critical challenge of capturing global dependencies across long-session encrypted traffic [1], researchers have increasingly turned to Transformer architectures driven by self-attention mechanisms [59-62]. By leveraging large-scale pretraining paradigms like BERT (e.g., NetBERT), these models excel at extracting highly generalizable traffic embeddings across diverse network environments. However, when transitioning from theoretical accuracy to practical deployment, Transformers encounter a severe algorithmic bottleneck. Their core self-attention mechanism incurs a quadratic computational complexity with respect to the traffic sequence

length, which drastically limits their scalability, throughput, and real-time applicability in ultra-high-speed networks [28]. Although recent efforts have utilized knowledge distillation to compress these models for lightweight edge deployment [11, 60-61], achieving an optimal balance between global contextual understanding and line-rate processing speeds remains a massive, unresolved deployment hurdle.

5) Graph Neural Networks (GNN): Targeting the profound difficulty in tracking coordinated threats like distributed botnets and the lateral movement of Advanced Persistent Threats (APTs), researchers have introduced Graph Neural Networks (GNNs) to explicitly model global topological relationships [63]. By abstracting network entities (hosts or flows) as nodes and their interactions as edges, GNNs provide a powerful structural lens for dynamic attack graph modeling and coordinated traffic anomaly detection [64-79], with advanced frameworks further exploring heterogeneous graph representations [65]. Nevertheless, practical deployment of GNNs reveals severe operational constraints. Most critically, their detection performance is extremely fragile to incomplete observability; it relies heavily on the absolute completeness and reliability of the underlying network topology information [63]. Furthermore, when applied to the massive, dynamic node populations of real-world enterprise networks, GNNs face an acute scalability bottleneck, struggling to achieve efficient large-graph representation learning without incurring prohibitive computational latency [66-67]. Collectively, deep learning models significantly enhance nonlinear representation capability but introduce higher computational and data requirements.

5.3 Hybrid Models and Emerging Paradigms

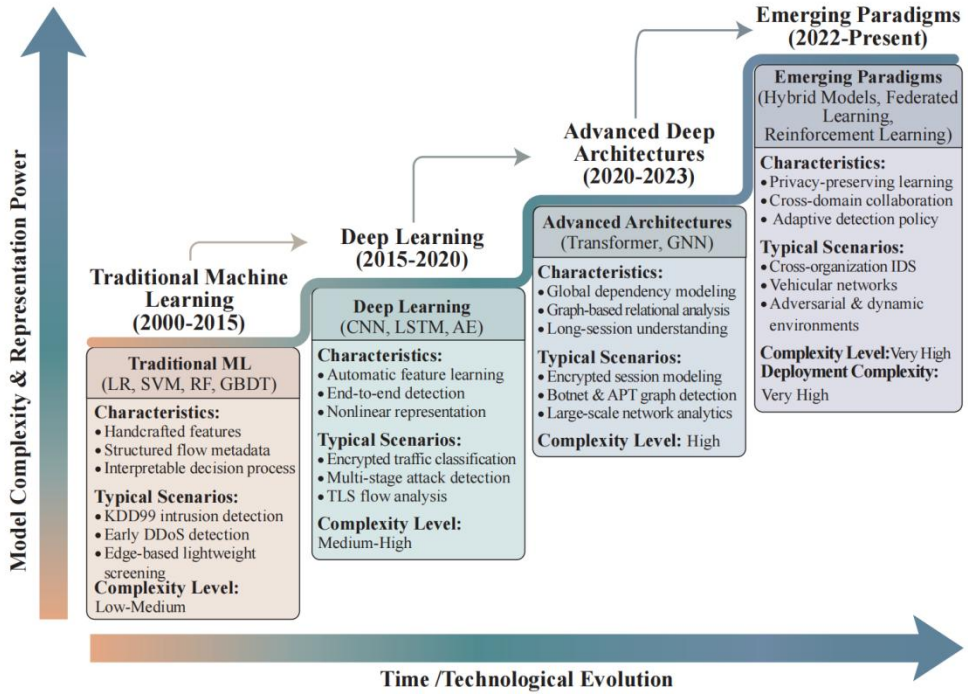

Figure 3 Processing Pipeline of ML-Based Malicious Traffic Analysis

To shatter the performance ceilings of individual architectures when confronting complex, multi-stage attacks, researchers increasingly deploy hybrid models that fuse complementary paradigms [28]. For instance, pairing PSO with LSTM-RNNs attempts to resolve feature redundancy alongside temporal modeling [4], while CNN-GRU pipelines aim to simultaneously decode spatial obfuscation and sequential patterns in encrypted traffic [34]. However, this brute-force integration imposes an exponential surge in computational overhead and architectural complexity, severely violating the latency constraints of high-speed networks, thus forcing a heavy reliance on aggressive pruning or attention mechanisms to salvage deployability [22].

Beyond model fusion, confronting the strict data privacy walls between organizations and the rapid mutation of adversarial tactics requires fundamentally new paradigms. Federated Learning (FL) was introduced to bypass isolated data silos, enabling cross-domain collaborative intrusion detection without sharing raw, sensitive traffic [68-69]. Yet, in operational deployments, FL is frequently crippled by massive communication bottlenecks, severe performance degradation caused by non-independent and identically distributed (non-IID) enterprise traffic, and a high susceptibility to malicious model poisoning [70-71].

Similarly, Reinforcement Learning (RL) formulates malicious traffic analysis as a Markov decision process to achieve dynamic, self-adapting defense policies against evolving threats [26, 72-74]. Nevertheless, relying on RL for real-time security introduces fatal operational vulnerabilities: its notoriously slow convergence, intricate reward design, and unpredictable exploration behaviors pose unacceptable security risks during the early stages of online deployment.

The evolutionary trajectory of learning paradigms in malicious traffic analysis is illustrated in Figure 3. The progression reflects increasing model capacity and robustness in response to encrypted traffic proliferation, large-scale network dynamics, and adversarial threats, albeit at the cost of higher computational and deployment complexity.

6 OPEN ISSUES AND FUTURE RESEARCH DIRECTIONS

Despite achieving near-perfect accuracy on isolated benchmark datasets, current deep learning (DL) techniques exhibit fundamental structural defects when transitioned to operational deployment [80-84]. Real-world intrusion detection systems are continually challenged by the encryption-by-default Internet architecture, severe cross-domain distribution shifts, and dynamic adversarial manipulations. As evidenced by the performance degradation of online anomaly detectors like Kitsune in realistic environments [51], closing the chasm between laboratory settings and practical security operations is paramount. Consequently, future research must decisively abandon the practice of merely ”chasing state-of-the-art scores on closed datasets.” Instead, the field requires a paradigm-level transformation across the following core dimensions to ensure that learning-based traffic intelligence systems are genuinely robust and deployable.

6.1 Behavior-Centric Intelligence under Encrypted Traffic

As end-to-end encryption permanently blinds traditional payload-inspection mechanisms, future research must fundamentally pivot from packet-level content matching to behavior-centric intelligence. Merely applying deep learning to encrypted byte streams is insufficient due to severe feature obfuscation. Instead, models must learn to extract robust, invariant representations from multi-granularity metadata—encompassing flow-level statistics, session dynamics, and cross-layer host interaction patterns. To achieve this without relying on scarce, manually annotated threat data, self-supervised pretraining paradigms are poised to become indispensable for mining intrinsic behavioral signatures from massive volumes of unlabeled encrypted traffic [84]. Furthermore, to counteract the extreme scarcity of zero-day attack labels under limited observability, integrating protocol semantics with advanced generative modeling is crucial to synthesize rich, realistic adversarial behaviors for proactive defense [88].

6.2 Domain-Generalizable and Transferable Learning Frameworks

The most fatal Achilles’ heel of contemporary deep learning-based detection systems is their severe fragility across heterogeneous network environments. Models highly optimized for a single, closed world dataset routinely experience catastrophic performance degradation when deployed in novel domains due to profound distribution shifts [82]. Consequently, future research must aggressively pivot from maximizing single-scenario accuracy to engineering inherently domain-generalizable and transferable learning frameworks. To achieve this, domain-adversarial training [83] and invariant representation learning offer critical pathways by explicitly forcing models to disentangle and extract environmentagnostic threat signatures [85-86]. This paradigm shift also strongly motivates a two-stage deployment strategy: large-scale, domain-agnostic pretraining followed by agile, lightweight adaptation at the network edge [87-90]. Furthermore, because real-world network traffic undergoes relentless conceptual drift, integrating continual learning mechanisms that support incremental knowledge acquisition without catastrophic forgetting is no longer optional, but an absolute prerequisite for sustaining long-term detection efficacy [91].

6.3 Robust and Adversarially-Aware Detection Models

Adversarial manipulation represents a catastrophic threat to the integrity of learning-based traffic analysis, as deep models exhibit inherent structural vulnerabilities that allow attackers to bypass detection through imperceptible perturbations [92]. Systematic evidence confirms that these evasion tactics are particularly potent in security-oriented settings, where static defense boundaries are easily overstepped by strategically crafted traffic [93-94]. Consequently, future research must pivot from treating robustness as a post hoc patch to establishing it as a first-class design objective, integrating adversarial training and robust optimization directly into the model synthesis process under realistic network constraints [95-100]. To support high-assurance deployment, the field must move beyond empirical defenses toward certified robustness and formal verification techniques that can provide rigorous reliability guarantees [96-97]. Furthermore, to improve diagnostic trust within Security Operations Center (SOC) workflows, it is essential to co-deploy adversarial perturbation detectors with practical explainability tools, ensuring that model decisions are both resilient and interpretable [89, 98]. Ultimately, addressing the broader privacy and security concerns in deep architectures requires a holistic optimization approach [99], aiming to learn stable representations that remain resilient to the inherent tension between high accuracy and adversarial reliability [100].

6.4 Scalable and Deployment-Oriented Model Design

A fundamental disconnect persists between the escalating complexity of deep learning architectures and the unforgiving latency and throughput constraints of production network environments. While high capacity models achieve impressive scores in offline laboratory benchmarks, they frequently fail to sustain line-rate processing in high-speed or resource-constrained infrastructures, rendering them operationally ineffective. To bridge this gap, future malicious

traffic analysis systems must undergo a decisive paradigm shift, prioritizing the joint optimization of detection accuracy and hardware-aware computational efficiency. This necessitates the aggressive integration of lightweight model architectures, model compression, and knowledge distillation to strip away structural redundancy while preserving security integrity. Furthermore, addressing the volatility of real-time traffic requires a transition toward online and streaming deep-learning paradigms that can adapt continuously and maintain stability within sub-millisecond latency budgets [90]. Ultimately, to achieve real-time performance at an enterprise scale, the research community should gravitate toward hierarchically distributed edge-cloud collaborative frameworks, where detection tasks are strategically partitioned to balance localized response speeds with centralized analytical power.

7 CONCLUSIONS

This survey provides a structured synthesis of malicious traffic analysis, covering the full pipeline from data construction to deployment. The main takeaway is that model performance is fundamentally system level: improvements depend on the joint design of dataset realism, traffic representation, learning architecture, and evaluation protocol, rather than any isolated algorithmic component. From this perspective, we analyze the practical operating conditions of traditional machine learning methods, deep neural networks, and hybrid pipelines, and summarize their trade-offs in terms of detection performance, interpretability, robustness, and computational cost. For practical deployment, three aspects are particularly important. First, encrypted traffic analysis should rely on behavior-centric and cross-layer information, rather than assumptions based on payload visibility. Second, cross-domain generalization should be explicitly considered in both model design and evaluation, through transfer-aware learning strategies and consistent benchmarking protocols. Third, robustness and efficiency should be jointly addressed, since high offline accuracy alone is insufficient without adversarial resilience and runtime feasibility.

Future research should focus on reproducible evaluation frameworks, representation learning with improved domain generalization capability, and lightweight robust models suitable for edge–cloud deployment. Overall, this field is gradually shifting from model-centric design toward system-level and deployment-oriented methodologies.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

FUNDING

This work was supported in part by the Poplar Talent Startup Funds of Tarim University in China(TDZKBS202571, TDZKBS202656).

REFERENCES

[1] Hong Y, Li Q, Yang Y, et al. Graph based encrypted malicious traffic detection with hybrid analysis of multi-view features. Information Sciences, 2023, 644: 119229.

[2] Hindy H, Brosset D, Bayne E, et al. A Taxonomy of Network Threats and the Effect of Current Datasets on Intrusion Detection Systems. IEEE Access, 2020, 8: 104650–104675. DOI: 10.1109/ACCESS.2020.3000179.

[3] Pacheco F, Exposito E, Gineste M, et al. Towards the Deployment of Machine Learning Solutions in Network Traffic Classification: A Systematic Survey. IEEE Communications Surveys & Tutorials, 2019, 21(2): 1988–2014. DOI: 10.1109/COMST.2018.2883147.

[4] Donkol A A E, Hafez A G, Hussein A I, et al. Optimization of intrusion detection using likely point PSO and enhanced LSTM-RNN hybrid technique in communication networks. IEEE Access, 2023, 11: 9469-9482.

[5] Abdelkhalek A, Mashaly M. Addressing the class imbalance problem in network intrusion detection systems using data resampling. Journal of Big Data, 2023, 10(1): 1-20.

[6] Anderson B, McGrew D. Machine Learning for Encrypted Malware Traffic Classification: Accounting for Noisy Labels and NonStationarity. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017: 1723–1732.

[7] Erhan L, Ndubuaku M, Di Mauro M, et al. Smart anomaly detection in sensor systems: A multi-perspective review. Information Fusion, 2021, 67: 64-79.

[8] Bhuyan M H, Bhattacharyya D K, Kalita J K. Network Anomaly Detection: Methods, Systems and Tools. IEEE Communications Surveys & Tutorials, 2014, 16(1): 303-336.

[9] Karatas G, Demir O, Sahingoz O K. Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset. Security and Communication Networks, 2020, 2020: 1–14.

[10] Milenkoski A, Vieira M, Kounev S, et al. Evaluating computer intrusion detection systems: A survey of common practices. ACM Computing Surveys, 2015, 48(1): 1-41.

[11] Mittal P. A comprehensive survey of deep learning-based lightweight object detection models for edge devices. Artificial Intelligence Review, 2024, 57(9): 242.

[12] Anderson B, McGrew D. Identifying Encrypted Malware Traffic with Contextual Flow Data. In Proceedings of the 15th ACM Asia Conference on Computer and Communications Security (ASIA CCS), 2016: 687–698.

[13] Yao H, Liu C, Zhang L, et al. Identification of Encrypted Traffic Through Attention Mechanism Based Long Short Term Memory. IEEE Transactions on Network and Service Management, 2022, 19(1): 507–519.
[14] Velan P, Cermak M, Celeda P, et al. A Survey of Methods for Encrypted Traffic Classification and Analysis. International Journal of Network Management, 2015, 25(5): 355–374. DOI: 10.1002/nem.1901.
[15] Faker O, Dogdu E. Intrusion Detection Using Big Data and Deep Learning Techniques. In Proceedings of the 2019 ACM Southeast Conference, 2019: 86–93.
[16] Qin L, Gu H, Wei W, et al. Spatio-temporal communication network traffic prediction method based on graph neural network. Information Sciences, 2024, 679: 121003.
[17] Aceto L, Ciuonzo D, Montieri A, Pescape A. Mobile Encrypted Traffic Classification Using Deep Learning: Experimental Evaluation, Lessons Learned, and Challenges. IEEE Transactions on Network and Service Management, 2019, 16: 445–458. DOI: 10.1109/TNSM.2019.2899078.
[18] Naghib A, Javidan R, Conti M. A Comprehensive and Systematic Literature Review on Intrusion Detection Systems in the Internet of Medical Things. ACM Computing Surveys, 2025, 57(1): 1-36.
[19] Kurniabudi A, Stiawan D, Bin Idris M Y, et al. CICIDS-2017 Dataset Feature Analysis With Information Gain for Anomaly Detection. IEEE Access, 2020, 8: 132911–132921.
[20] Mohammadpour L, Ling T C, Liew C S, et al. A Survey of CNN-Based Network Intrusion Detection. Applied Sciences, 2022, 12(16): 8162.
[21] Fu C, Li Q, Xu K. Flow interaction graph analysis: Unknown encrypted malicious traffic detection. IEEE/ACM Transactions on Networking, 2024, 32: 2972–2987. DOI: 10.1109/TNET.2024.3370851.
[22] Sajid M, Alshehri M D, Alghamdi R A. Enhancing Intrusion Detection: A Hybrid Machine and Deep Learning Approach. Computers & Security, 2024, 137: 103611.
[23] Binbusayyis A, Vaiyapuri T. Identifying and benchmarking key features for cyber intrusion detection: An ensemble approach. Information Sciences, 2019, 485: 452-463.
[24] Yin Y, Liu Y, Zhang T, et al. IGRF-RFE: A Hybrid Feature Selection Method for MLP-Based Network Intrusion Detection on UNSW-NB15 Dataset. Journal of Information Security and Applications, 2023, 73: 103414.
[25] Yang L, Shami A. A Transfer Learning and Optimized CNN Based Intrusion Detection System for Internet of Vehicles. IEEE Transactions on Intelligent Transportation Systems, 2022, 23(12): 25447-25458.
[26] Watts L, Makhoul A, Perrot C, et al. A Dynamic Deep Reinforcement Learning-Bayesian Framework for Anomaly Detection. Computers & Security, 2022, 121: 102842.
[27] Buczak A L, Guven E. A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. IEEE Communications Surveys & Tutorials, 2016, 18: 1153 – 1176. DOI: 10.1109/COMST.2015.2494502.
[28] Ferrag M A, Maglaras L, Moschoyiannis S, et al. Deep Learning for Cyber Security Intrusion Detection: Approaches, Datasets, and Comparative Study. Journal of Information Security and Applications, 2020, 50: 102419. DOI: 10.1016/j.jisa.2019.102419.
[29] Kumar P, Kumar G. Issues and Challenges of Intrusion Detection Systems: A Comprehensive Survey. International Journal of Computer Applications, 2015, 131: 1–12.
[30] Chen Z, Jiang F, Cheng Y, et al. XGBoost Classifier for DDoS Attack Detection and Analysis in SDNBased Cloud. In Proceedings of the IEEE International Conference on Big Data and Smart Computing (BigComp), 2018: 251–256. DOI: 10.1109/BigComp.2018.00044.
[31] Gamage S, Samarabandu J. Deep Learning Methods in Network Intrusion Detection: A Survey and an Objective Comparison. Journal of Network and Computer Applications, 2020, 169: 102767. DOI: 10.1016/j.jnca.2020.102767.
[32] Vinayakumar R, Soman K P, Poornachandran P, et al. Deep Learning Approach for Intelligent Intrusion Detection System. IEEE Access, 2019, 7: 41525–41550. DOI: 10.1109/ACCESS.2019.2895334.
[33] Albahar M A, Alazeb R S, Almazroi A M. An Improved Support Vector Machine for Intrusion Detection System. Computers, Materials & Continua, 2022, 70: 1207–1222. DOI: 10.32604/cmc.2022.019456.
[34] Alsaedi A, Moustafa N, Tari Z, et al. TON IoT Telemetry Dataset: A New Generation Dataset of IoT and IIoT for Data-Driven Intrusion Detection Systems. IEEE Access, 2020, 8: 165130 – 165150. DOI: 10.1109/ACCESS.2020.3022862.
[35] Breiman L. Random Forests. Machine Learning, 2001, 45: 5–32. DOI: 10.1023/A:1010933404324.
[36] Ahmad Z, Khan A S, Shiang C W, et al. Network Intrusion Detection System: A Systematic Study of Machine Learning and Deep Learning Approaches. Transactions on Emerging Telecommunications Technologies, 2021, 32: e4150. DOI: 10.1002/ett.4150.
[37] Xin Y, Kong L, Liu Z, et al. Machine Learning and Deep Learning Methods for Cybersecurity. IEEE Access, 2018, 6: 35365–35381. DOI: 10.1109/ACCESS.2018.2836950.
[38] Apruzzese G, Colajanni M, Ferretti L, et al. On the Effectiveness of Machine and Deep Learning for Cyber Security. In IEEE 10th International Conference on Cloud Networking (CloudNet), 2018: 1-8. DOI: 10.1109/CloudNet.2018.8549288.
[39] Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016: 785 – 794. DOI: 10.1145/2939672.2939785.

[40] Thakkar A, Lohiya R. A Review on Machine Learning and Deep Learning Perspectives of IDS for IoT: Recent Updates, Security Issues, and Challenges. Archives of Computational Methods in Engineering, 2021, 28: 3211–3243. DOI: 10.1007/s11831-020-09496-0.
[41] Sarker I H, Kayes A S M, Badsha S, et al. Cybersecurity Data Science: An Overview from Machine Learning Perspective. Journal of Big Data, 2020, 7: 41. DOI: 10.1186/s40537-020-00318-5.
[42] Mishra A K, Yadav V K, Shukla S K. A Comparative Analysis of Supervised Machine Learning Algorithms for Intrusion Detection. International Journal of Advanced Computer Science and Applications, 2020, 11: 415-422.
[43] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature, 2015, 521: 436–444. DOI: 10.1038/nature14539.
[44] Goodfellow I, Bengio Y, Courville A. Deep Learning; MIT Press: Cambridge, MA, USA, 2016.
[45] Wang J, Liu Y, Li Y. 1D CNN-Based Network Intrusion Detection with Normalization on Imbalanced Data. In IEEE International Conference on Communications (ICC), 2020: 1-6. DOI: 10.1109/ICC40277.2020.9148865.
[46] Wang Z. The Applications of Deep Learning on Traffic Identification. In BlackHat USA, 2015.
[47] Hindy H, Bayne E, Bures M, et al. Machine Learning Based IoT Intrusion Detection System: An MQTT Case Study. In IEEE Irish Signals and Systems Conference (ISSC), 2020: 1-6. DOI: 10.1109/ISSC49989.2020.9180164.
[48] Zhao R, Yan R, Chen Z, et al. Deep Learning and Its Applications to Machine Health Monitoring. Mechanical Systems and Signal Processing, 2019, 115: 213-237. DOI: 10.1016/j.ymssp.2018.05.050.
[49] Hochreiter S, Schmidhuber J. Long Short-Term Memory. Neural Computation, 1997, 9: 1735 – 1780. DOI: 10.1162/neco.1997.9.8.1735.
[50] Cho K, van Merrienboer B, Gulcehre C, et al. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014: 1724–1734. DOI: 10.3115/v1/D14-1179.
[51] Mirsky Y, Doitshman T, Elovici Y, et al. Kitsune: An Ensemble of Autoencoders for Online Network Intrusion Detection. In Network and Distributed Systems Security (NDSS) Symposium, 2018. DOI: 10.14722/ndss.2018.23204.
[52] Anderson B, McGrew D. Identifying Encrypted Malware Traffic with Contextual Flow Data. In Proceedings of the 2016 ACM on Asia Conference on Computer and Communications Security, 2016: 35-46. DOI: 10.1145/2897845.2897890.
[53] Pascanu R, Mikolov T, Bengio Y. On the Difficulty of Training Recurrent Neural Networks. In International Conference on Machine Learning (ICML), 2013: 1310-1318.
[54] Hinton G E, Salakhutdinov R R. Reducing the Dimensionality of Data with Neural Networks. Science, 2006, 313: 504–507. DOI: 10.1126/science.1127647.
[55] Cortes C, Vapnik V. Support-Vector Networks. Machine Learning, 1995, 20: 273 – 297. DOI: 10.1007/BF00994018.
[56] Kingma D P, Welling M. Auto-Encoding Variational Bayes. In International Conference on Learning Representations (ICLR), 2014. Available online: https://arxiv.org/abs/1312.6114.
[57] Liu F T, Ting K M, Zhou Z. Isolation Forest. In IEEE International Conference on Data Mining, 2008: 413-422. DOI: 10.1109/ICDM.2008.17.
[58] Zhou C, Paffenroth R C. Anomaly Detection with Robust Deep Autoencoders. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017: 665 – 674. DOI: 10.1145/3097983.3098052.
[59] Vaswani A, Shazeer N, Parmar N, et al. Attention Is All You Need. Advances in Neural Information Processing Systems (NeurIPS), 2017, 30.
[60] Lan Z, Chen M, Goodman S, et al. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In International Conference on Learning Representations (ICLR), 2020.
[61] Cho J H, Hariharan B. On the Efficacy of Knowledge Distillation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019: 4794-4802. DOI: 10.1109/ICCV.2019.00490.
[62] Lin Z, Feng M, Santos C N d, et al. A Structured Self-attentive Sentence Embedding. In International Conference on Learning Representations (ICLR), 2017. Available online: https://arxiv.org/abs/1703.03130.
[63] Zhou J, Cui G, Hu S, et al. Graph Neural Networks: A Review of Methods and Applications. AI Open, 2020, 1: 57-81. DOI: 10.1016/j.aiopen.2021.01.001.
[64] Pareja A, Domeniconi G, Chen J, et al. EvolveGCN: Evolving Graph Convolutional Networks for Dynamic Graphs. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34: 5363-5370. DOI: 10.1609/aaai.v34i04.5984.
[65] Zhang C, Song D, Huang C, et al. Heterogeneous Graph Neural Network. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019: 793-803. DOI: 10.1145/3292500.3330961.
[66] Hamilton W, Ying Z, Leskovec J. Inductive Representation Learning on Large Graphs. Advances in Neural Information Processing Systems (NeurIPS), 2017, 30.
[67] You J, Ying R, Leskovec J. Position-aware Graph Neural Networks. In International Conference on Machine Learning (ICML), 2019: 7134-7143.

[68] McMahan B, Moore E, Ramage D, et al. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS), 2017: 1273–1282.
[69] Caldas S, Duddu S M K, Wu P, et al. LEAF: A Benchmark for Federated Settings. In Workshop on Federated Learning for Data Privacy and Confidentiality (NeurIPS), 2019. Available online: https://arxiv.org/abs/1812.01097.
[70] Zhao Y, Li M, Lai L, et al. Federated Learning with Non-IID Data. In arXiv preprint arXiv:1806.00582, 2018. Available online: https://arxiv.org/abs/1806.00582.
[71] Bhagoji A N, Chakraborty S, Mittal P, et al. Analyzing Federated Learning through an Adversarial Lens. In International Conference on Machine Learning (ICML), 2019: 634–643.
[72] Sutton R S, Barto A G. Reinforcement Learning: An Introduction, 2nd ed.; MIT Press: Cambridge, MA, USA, 2018.
[73] Xiao L, Li Y, Han G, et al. PHY-Layer Spoofing Detection with Reinforcement Learning in Wireless Networks. IEEE Transactions on Vehicular Technology, 2016, 65: 10037–10047. DOI: 10.1109/TVT.2016.2524258.
[74] Liu Y, Chen Y, Shen C. A Deep Reinforcement Learning Based Approach for Network Intrusion Detection. Computers & Security, 2021, 108: 102314. DOI: 10.1016/j.cose.2021.102314.
[75] Paasch C, Bonaventure O. QUIC: Opportunities and threats in the evolution of the Internet. ACM SIGCOMM Computer Communication Review, 2021, 51: 42-48. Available online: https://doi.org/10.1145/3457175.3457183.
[76] Stolfo S J, Fan W, Lee W, et al. Cost-based modeling for fraud and intrusion detection: Results from the JAM project. In Proceedings of DARPA Information Survivability Conference and Exposition (DISCEX), 2000. Available online: https://doi.org/10.1109/DISCEX.2000.821515.
[77] Korczynski M, Duda A. Markov chain fingerprinting to classify encrypted traffic. In IEEE INFOCOM 2014: 781-789.
[78] Durumeric Z, Kasten J, Adrian D, et al. The Matter of Heartbleed. In Proceedings of the 2014 Internet Measurement Conference (IMC), 2014: 475–488.
[79] Wang X, Liu S, Zhang J, et al. Graph-based Malicious Traffic Detection with Hierarchical Attention. IEEE Transactions on Network and Service Management, 2023. Available online: https://doi.org/10.1109/TNSM.2023.3245678.
[80] Apruzzese G, Colajanni M, Ferretti L, et al. On the Cross-evaluation of Machine Learning-based Network Intrusion Detection Systems. IEEE Transactions on Network and Service Management, 2022, 19: 2482-2496. Available online: https://doi.org/10.1109/TNSM.2022.3162936.
[81] Sharafaldin I, Lashkari A H, Ghorbani A A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In Proceedings of the 4th International Conference on Information Systems Security and Privacy (ICISSP), 2018.
[82] Pan S J, Yang Q. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 2010, 22: 1345-1359.
[83] Ganin Y, Ustinova E, Ajakan H, et al. Domain-adversarial training of neural networks. Journal of Machine Learning Research, 2016, 17: 2096-2030.
[84] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), 2019: 4171–4186.
[85] Arjovsky M, Bottou L, Gulrajani I, et al. Invariant Risk Minimization. In arXiv preprint arXiv:1907.02893, 2019. Available online: https://arxiv.org/abs/1907.02893.
[86] Bengio Y, Courville A, Vincent P. Representation Learning: A Review and New Perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35: 1798-1828.
[87] Scholkopf B, Locatello F, Bauer S, et al. Toward Causal Representation Learning. Proceedings of the IEEE, 2021, 109: 612-634.
[88] Odena A, Olah C, Shlens J. Conditional Image Synthesis with Auxiliary Classifier GANs. In International Conference on Machine Learning (ICML), 2017: 2642–2651.
[89] Ribeiro M T, Singh S, Guestrin C. 'Why Should I Trust You?': Explaining the Predictions of Any Classifier. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2016: 1135–1144.
[90] Sahoo D, Pham Q, Lu J, et al. Online Deep Learning: Learning Deep Neural Networks on the Fly. In Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI), 2018: 2660-2666.
[91] Li Z, Hoiem D. Learning without Forgetting. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40: 2935-2947.
[92] Szegedy C, Zaremba W, Sutskever I, et al. Intriguing properties of neural networks. In International Conference on Learning Representations (ICLR), 2014.
[93] Biggio B, Corona I, Maiorca D, et al. Evasion Attacks against Machine Learning at Test Time. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD), 2013: 387–402.
[94] Carlini N, Wagner D. Towards Evaluating the Robustness of Neural Networks. In IEEE Symposium on Security and Privacy (SP), 2017: 39–57.

[95] Madry A, Makelov A, Schmidt L, et al. Towards Deep Learning Models Resistant to Adversarial Attacks. In International Conference on Learning Representations (ICLR), 2018.
[96] Cohen J, Rosenfeld E, Kolter Z. Certified Adversarial Robustness via Randomized Smoothing. In International Conference on Machine Learning (ICML), 2019: 1310-1320.
[97] Katz G, Barrett C, Dill D L, et al. Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks. In International Conference on Computer Aided Verification (CAV), 2017: 97–117.
[98] Metzen J H, Genewein T, Fischer V, et al. On Detecting Adversarial Perturbations. In International Conference on Learning Representations (ICLR), 2017.
[99] Liu X, Xie L, Wang Y, et al. Privacy and Security Issues in Deep Learning: A Survey. IEEE Access, 2021, 9: 4566-4593.
[100] Tsipras D, Santurkar S, Engstrom L, et al. Robustness May Be at Odds with Accuracy. In International Conference on Learning Representations (ICLR), 2019.