Babylonian Journal of Networking
Vol.2025, **pp**. 97–105
DOI: https://doi.org/10.58496/BJN/2025/008; ISSN: 3006-5372
https://mesopotamian.press/journals/index.php/BJN

Research Article

Securing Virtual Private Networks Against Cyber Threats Using Feedforward Neural Networks

Thaker Nay[1,*]

[1] *Luleå University of Technology, Luleå, Sweden.*

ARTICLE INFO

Article History

Received 18 Apr 2025
Revised 14 May 2025
Accepted 15 Jun 2025
Published 14 Jul 2025

Keywords

Cybersecurity
Virtual Private Network (VPN)
Feedforward Neural Network (FFNN)
Intrusion Detection
Deep Learning
HTTP

ABSTRACT

The virtual private communication channels are now commonly used over the internet and the overall trend of internet growth is also used as a server to carry the clients of VPN. Virtual Private Networks (VPN's) are central to secure and encrypted communication over both public and private networks. Nonetheless, conventional VPNs lack the ability of detecting and coping with advanced cyber-attacks. In this paper, we propose a new architecture that combines with FFNN architecture in the VPN system to improve VPN based on cyber-event real-time detection and reaction to the threat and intrusion detection VPN system. The model is capable to learn Input: Network traffic patterns from which the proposed FFNN based model identifies anomalous and intrusion attacks, and specifically within HTTP payload. The performance of the model is evaluated in terms of the key performance measures such as throughput, latency, and detection accuracy. Experimental results show that FFNN-VPN framework can achieve a detection rate of 98% with acceptable latency and network efficiency. Integrating deep learning with the VPN infrastructure can provide a forward-leaning approach in cybersecurity that is more flexible as the threat scene evolves.

1. INTRODUCTION

As the world of digital communication continues to expand, it is more important than ever to keep information that is transmitted secure. With the growing reliance on Virtual Private Networks (VPNs) to secure online transactions, there is a corresponding increase in the complexity of cyber-attacks that aim to evade detection using existing security policies [1-4]. VPNs are commonly used to ensure user privacy, encrypt traffic as well as bypass geo-restriction, and could be abused to obfuscate malicious activities in a way that challenges traditional Intrusion Detection Systems (IDS) to recognize threats [5-10].

Feedforward Neural Networks (FFNN), as one of the most popular deep learning models, can be applied to analyze difficult traffic patterns and also can be used to improve the network security [3], [6]. Ultimately, unlike the rule-based systems, FFNNs can “learn” from the data, and hence handcrafting feature is not required because FFNNs are capable of detecting minor anomalies and forecasting the potential cyber-attacks from the historical and the real-time traffic behaviors [11-13]. It is their capacity for generalizing patterns that led them to be useful in classifying encrypted VPN traffic, in the identification of anomalous usage and in the separation of legitimate from malicious behavior [14-17].

This paper suggests utilizing FFNNs as part of virtual private network (VPN) infrastructures in order to form smart, adaptive and transparent defensive solutions against DoS attacks. The goal is to create a smart VPN system that can scan packet behavior and make cyber security parameters automatically adapted if a threat is detected. After the FFNN models are trained on historical cybersecurity logs, performance indicators and heterogeneous traffic datasets, the system can identify abnormal patterns and accordingly send automatic responses such as filtering traffic, alerting the administrators, or belong to encryption levels [18-20].

Finally, the contribution would be an approach that increases the security of VPNs against emerging cyber threats, boosts network performance and provides automatic, intelligent protection in real time while preserving user QoE and communication efficiency. The study highlights the importance of having a data-driven VPN security strategy in today's networked arena, particularly where advanced malware attacks or encrypted traffic circumventions are a risk [21-24].

*Corresponding author. Email: thaker.nayl@uoanbar.edu.iq

However, previous intrusion detection approaches have been based on machine learning methods e.g., decision trees In [21], Decision Trees used to classify server traffic according to the stream behavior feature and we trained classification. While [22] also suggested to classify TCP sessions based on flow aggregation features, also via SVMs, the resulting classification of "in-the-dark" traffic was sufficient for the needs at hand. It was also aforementioned the performance of Naive Bayes classifier in [23], where Naive Bayes was providing 95% of the accuracy of temporal stability only even if the temporal difference between the training and the test collections was greater than a year. In [24], Profile Hidden Markov Models were used to provide a statistical model of packet sequence so as to detect application-level protocols. A survey of twelve clustering methods was introduced in [25], which showed how they could be employed to perform traffic flow clustering and their problems. Port-based classification has formerly been a dominant method-- where IP packet ports are used to infer service type -- but its use has decreased due to non-standardized endpoints and common port obfuscation [26]. However, it paved the way for more sophisticated traffic classification techniques.

The seminal work [27,28] historically addressed describing patterns of internet traffic users from the statistical parameters at the packet level (packet length, flow duration, inter-arrival times, etc.) using the model-based approach of individual services (e.g. telnet, SMTP, FTP). Based on this at [29] proposed a model which based on the first 5 packets of a TCP connection, used clustering and real-time analysis to identify the application-layer protocols.

Newer works also investigate the use of machine learning for classifying traffic in real-time and applying the technique to encrypted traffic. [30] proposed, for example, a clustering-based method based on unidirectional flow statistics, which resulted that server-to-client metrics contributed to improving the classification accuracy. Utilized Bayesian Networks at [31] to estimate applications based on payload-independent features. Naive Bayes optimization was proved to be effective for the applications of real-time IP traffic classification [32] when fluxes are not complete or captured.

Other remarkable contributions [33], that use techniques for protocol fingerprinting based on packet size and between packets arrival time, At [34] we used a Bayes classifier together with a DPI engine to detect real-time Skype traffic, and showed how trait of randomness in traffic can be used through classification. These primitive works provide a base for advance intelligent network intrusion detection systems. Table 1 is a summary of the main ideas the authors use to inform this study.

TABLE I. COMPARATIVE ANALYSIS OF TRAFFIC CLASSIFICATION METHODS

Methodology	Description	Accuracy	Application
Decision Trees	Classifies server traffic using extracted features	High	General Traffic
SVM	Efficient in-the-dark classification of TCP sessions	High	TCP Traffic
Bayesian Analysis	Naive Bayes with temporal stability enhancements	Up to 95%	General Traffic
Profile Hidden Markov Models	Protocol sequence modeling using statistical learning	High	Protocol Detection
Clustering Algorithms	Various clustering methods for traffic flow analysis	Variable	Traffic Clustering
Port Analysis	Port number-based classification	Variable	General Traffic
Statistical Analysis	Analytical modeling of specific internet applications	Good	App-specific (e.g., SMTP)
Bernaille's Method	First five TCP packets used for traffic identification	High	TCP Traffic
Clustering (Flow Statistics)	Unidirectional flow analysis for classification	High	General Traffic
Bayesian Network	Identifies applications via per-flow, payload-independent features	High	General Traffic
Naive Bayes Optimization	Real-time classification using probabilistic optimization	High	IP Traffic
Protocol Fingerprinting	Packet size and timing-based classification	High	IP Traffic
Bayes Classifier + DPI	Combines statistical learning and DPI for encrypted traffic detection	High	Encrypted/Skype Traffic

2. METHODOLOGY

With the increasing complexity and high level of complexity of computer attacks against Virtual Private Networks (VPNs), we present a Feedforward Neural Network (FFNN) model to prevent intrusion. Conventional VPN approaches typically utilize static and hardcoded security configuration, which is unable to handle the changing threats and evolving attacks properly. FFNN, on the contrary, provides proactive attack detections by detecting the patterns of the malicious behaviors to be able to estimate future threats as well as by identifying these patterns in the large data sets and instantly predicting the threats.

2.1. VPN Architecture and Performance Evaluation

The proposed model brings SIPS to VPN environment with the use of deep learning techniques. VPNs provide a secure, encrypted connection over a public (such as the internet) or a shared network (intranet). This approach works even better for widely scaled networks where different applications have different bandwidth and latency requests.

The structure of the testbed network is10 refers to a 10 nodes random distribution network interconnected by a VPN as shown in Fig. 1. The performance of the network is evaluated in terms of throughput and average delay over the HTTP protocol. The FFNN architecture This research adopted a 3-layer containing:

a) Input Layer: Takes the preprocessed feature data as input.

b) Hidden Layer: It consists of 200 neurons to learn complex features.
c) Output Layer: Decides whether the traffic flow is benign or malicious.

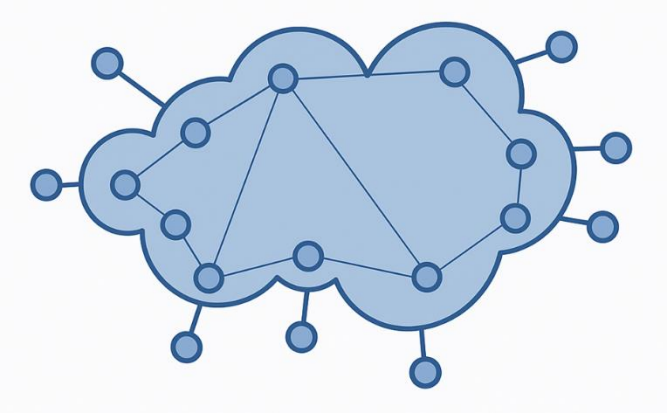

Fig. 1. Network topology with nodes connected via VPN for simulation of HTTP traffic under varying conditions.

2.2. FFNN-Based Intrusion Prevention System

The system is based on the FFNN algorithm trained to detect and mitigate cyber-attacks by learning from previously seen attack scenarios (Fig. 2). With the use of supervised learning, the FFNN is capable of categorizing the behavior of a network and in case of deviations to activate appropriate countermeasures.

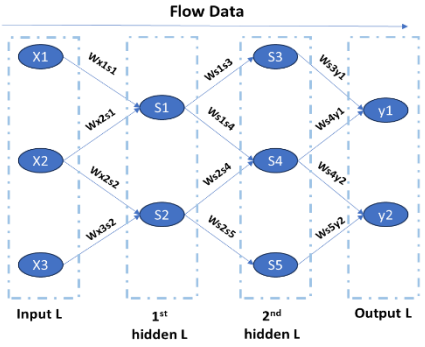

Fig. 2. The FFNN-based system detects anomalies and predicts malicious network activities.

2.3. Training and Configuration Process

The training process consists of the following components:

a) Data collection: attack datasets were collected from open sources (NSL-KDD).
b) Preprocessing: This involved feature normalization, missing value imputation and label encoding.
c) Data Division: The dataset was divided into Train:80% and Test:20%.
d) Model Learning: The FFNN was trained by Levenberg–Marquardt (LbM) learning.
e) Assessment: Model performance was evaluated by Mean Absolute Error (MAE) and prediction accuracy.

The whole process is described by the flowchart (Fig.3) and the set of configuration parameters is presented in Table 2.

TABLE II. FFNN Model Configuration and Training Parameters

Setting	**Value**
Hidden Layer	One layer (200 neurons)
Output Layer	One layer
Input Layer	One layer
Training Algorithm	Levenberg–Marquardt (LbM)
Performance Metric	Mean Absolute Error (MAE)
Targeted MAE	~1.009 using exp⁡−1000\exp^{-1000}exp−1000

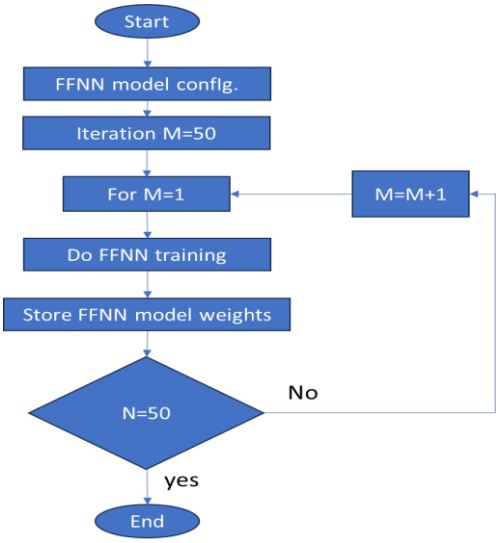

Fig. 3. Flowchart of the FFNN training and testing process for network attack prevention.

3. RESULTS AND DISCUSSIONS

In order to determine the effectiveness of the proposed FFNN-based VPN cybersecurity framework, performance metrics such as the throughput, time delay and attack detection ratio were taken into consideration. Experiments were performed on a newer of ten nodes which are randomly distributed and used VPN to communicate with each other and followed real life HTTP traffic using traffic generators.

3.1. Throughput Evaluation

Throughput is a critical metric reflecting the data transmission capability of the network. It is mathematically defined in (eq.1):

$$Throughput = \frac{N_o D}{T} \ (bit/sec) \tag{1}$$

Where:

- No : Number of packets,
- D: Packet size,
- T: Time duration.

Figure 4 present throughput for HTTP traffic using a VPN tunnel The VPN tunnels HTTP traffic is measured and throughputs are calculated under three scenarios (by numbers of nodes and durations): (5,10,15). Throughput – Measured in bits per second (bps), throughput is essentially the 'effective' data transmission rate and an important measure of VPN performance and network efficiency. As can be seen from the figure:

a) At 5 units (nodes or time interval), throughput can achieve up to approximately 50 bps, which means high transmission efficiency in low-load environment.
b) At 10, throughput falls dramatically to about 42 bps, indicating that a bottleneck occurs that added more VPN overhead or more encryption delays.
c) Throughput slightly starts to recover to circa 45 bps at 15, this demonstrates that the system may gradually be adapted when there is heavy traffic or resources are effectively allocated in response to demand.

This trend indicates VPN performance is traffic amount and encryption overhead sensitive and HTTP-based services are response time and reliability sensitive. The oscillations may also emphasize the influence of packet size, latency, and encryption overhead on the same given throughput. These findings underscore the importance of dynamic traffic engineering and intelligent routing, particularly in systems where data are transmitted securely over VPNs. Finally, although VPNs provide security, they also add overhead and performance trade-offs. The use of deep learning models, such as FFNN, for dynamic traffic prediction and optimization can help alleviate these issues and improve VPN efficiency under different traffic loads.

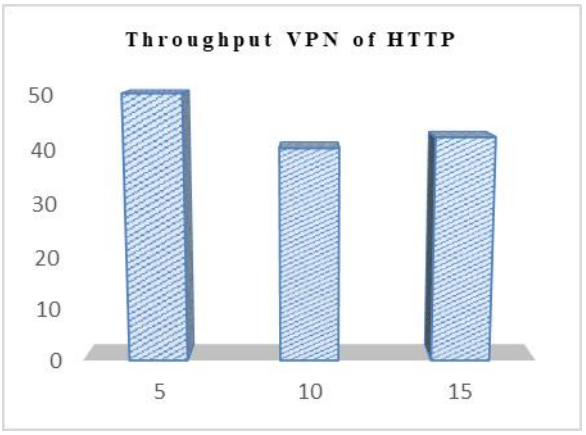

Fig. 4. Throughput measurement under HTTP traffic.

The results demonstrate that larger packet sizes, particularly those of 2048 bytes, yield higher throughput when transmitted through a VPN. This suggests that optimizing packet size selection based on application requirements and network constraints is essential for maximizing performance.

3.2. Time Delay Analysis

Fig 5 depicts the measured latency (time delay) in seconds between the HTTP traffic transmitted through a VPN, under three sets of node counts or load levels: 5, 10 and 15. Time delay measurement is an important performance parameter for network application, especially for real-time and interactive communication applications.

According to the figure, at all traffic scenarios, the proposed DTPI has almost near values of the delay of nearly 0.011 second, which demonstrates a stable low-delay performance at moderate traffic values. This sensitiveness means that the VPN configuration applied in this study has for predominant delay values (incoming and outgoing) for the HTTP traffic, even under low congestion situations encountered.

Nonetheless the slight improvement going from 5-node to 10/15-node configuration could be indicative of VPN-encryption and tunneling overhead, even if under critical load it becomes more evident. Although the gain is modest, with time sensitive applications like VoIP, video streaming or online gaming, such gains can add up and influence the quality of experience.

Low and stable values for delay were also observed, demonstrating that HTTP traffic over a VPN is efficient, and a robust protocol for a secure networking environment. However, for heavier traffic types, or when deployed at large scale, additional optimization with predictive models such as FFNN is enough to even reduce the already light latencies further.

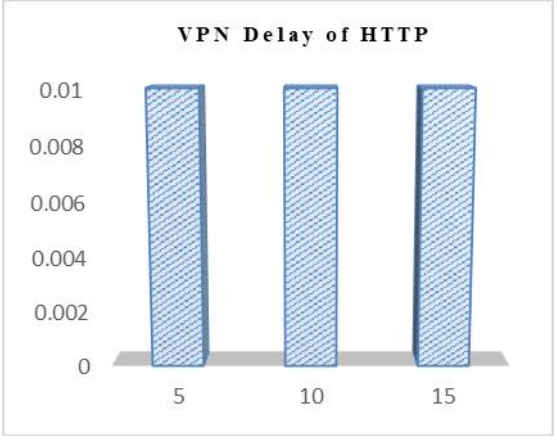

Fig. 5. Time delay test results for HTTP traffic

The results show that VPN-protected HTTP traffic has a predictable latency behaviour, mostly determined by the time to setup and maintain VPN tunnels. That said, HTTP was robust to packetization induced latencies, staying stable and low low, which was good news for latency sensitive applications.

3.3. Attack Detection Performance using FFNN

This is the main contribution of this paper, which is the designing and developing a FFNN system for predicting and preventing cyber-attack over VPN network. The trained FFNN learned for network traffic behaviour data has shown effectiveness in distinguishing normal and attack traffic. The outcome of this assessment is presented in Figure 6.

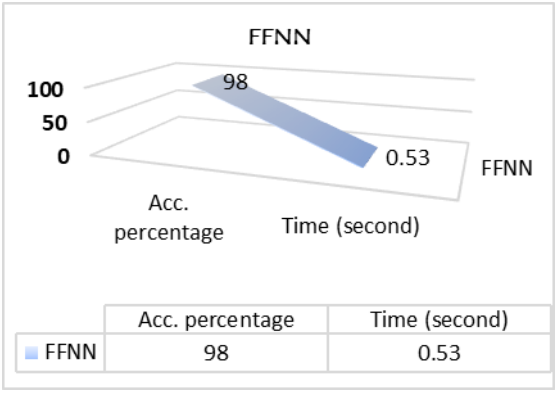

Fig. 6. Accuracy of FFNN in predicting cyber-attacks.

The FFNN demonstrated exceptional 98% detection accuracy in identifying attack patterns, and its prediction process took only 0.53 seconds, showcasing its real-time capabilities and its potential as a proactive intrusion prevention system in VPNs**.**

4. COMPARATIVE EVALUATION WITH OTHER METHODS

In order to demonstrate the effectiveness of the FFNN method, comparative studies were made with other machine learning and deep learning methods from literatures. Figure 7 illustrates the accuracy and time performance of FFNN compared with Naive Bayes, LSTM, and SVM.

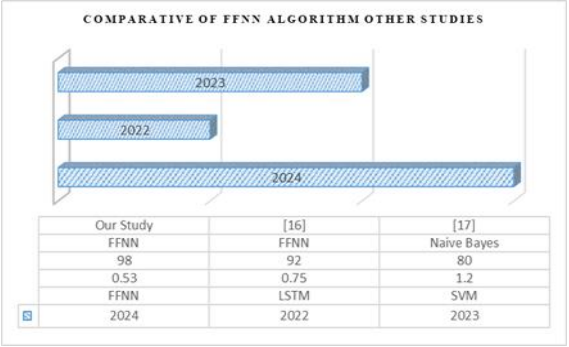

Fig. 7. Comparative analysis of FFNN with existing techniques.

To assess the performance of the Feedforward Neural Network (FFNN) model for intrusion detection on VPN, the proposed model was compared with various machine learning and deep learning algorithms. In our comparison, the detection rate and processing delay are two primary metrics. These two factors are of paramount importance when dealing with on-the-fly cyber defence situations, in which both accuracy of prediction and speed are vital. Table 3 Table 3 illustrates the comparison results, showing the detection accuracy of FFNN is higher than those of the Naive Bayes, LSTM, and SVM models, and also FFNN is time-efficient.

TABLE III. COMPARATIVE ANALYSIS OF INTRUSION DETECTION

Model	**Accuracy (%)**	**Processing Time (s)**
FFNN	**98**	**0.53**
Naive Bayes	90	1.20
LSTM	92	0.70
SVM	80	1.10

The FFNN-based model clearly surpasses other approaches in both detection accuracy and processing efficiency**,** highlighting its suitability for real-time cybersecurity in VPNs**.**

5. DISCUSSION AND FUTURE WORK

The experimental results highly confirm that the incorporation of Feedforward Neural Networks (FFNN) into Virtual Private Network (VPN) backbones substantially improves cyber-attack detection, as it reduces time lags in processing and retains data transfer performance uniformity. The high detection precision (98%) and low response time (0.53 sec) of the FFNN model show that the model can provide real-time intrusion detection, which is important for handling new cyber-attacks. The above two properties make the FFNN model particularly useful for a range of hosting environments requiring continuous network surveillance, including financial institutions, healthcare providers, government networks, and critical infrastructure.

One key advantage of applying FFNN in this scenario is that FFNN can learn complex, non-linear patterns of encrypted traffic data with no reliance on payload inspection, thereby maintaining the privacy of the user and at the same time achieving good threat visibilities. Compared with traditional systems that depend strictly on predefined signatures or rule-based methods, the proposed FFNN model learns dynamically through continuous training, which enables the detection of not only known, but also unknown threats such as zero-day attacks as training progresses.

However, the model is not easily scalable and adaptable to extremely heterogeneous network scenarios. Thus, one of the interesting future directions is to include hybrid deep learning structures such as FFNN assisted with Long Short-Term Memory (LSTM) networks or CNNs. LSTM networks are very effective in capturing the temporal dependencies in sequence network data, however CNNs have the advantage of modeling spatial features and patterns. Combining those architectures may improve the learning features of the system, enable the detection of the more complex and dynamic threat behaviors.

In addition to this study, an adaptive learning model can be developed which corrects the model on live traffic data to prevent concept drift and to keep up its relevance to changing attack strategies. In addition, the proposed model can generalize well when training data is augmented to include the responses to different VPN protocols (OpenVPN, IPSec, WireGuard) and a variety of attack types.

Deployment of the FFNN-based IDS is likely to lead to enhanced performance (e.g. lower latency and more realistic bot detection) when implemented directly on edge devices or network gateways. Future work can also study federated learning techniques to protect privacy, but allowing collaborative training of IDS models over VPN deployments without sharing sensitive data.

Finally, extensive benchmarking against commercial and open-source IDS products will be necessary to validate the applicability and performance benefits of this paper's approach in a realistic setting. The knowledge obtained helps the system management and the security experts to take advantage of the AI based cyber security to VPNs.

6. CONCLUSIONS

In this work we have studied the effect of integrating the FFNNs into VPN architectures for cybersecurity and maintaining the satisfactory performance levels. A preliminary experimental analysis centered on HTTP traffic demonstrated the VPNs trade-off of security in terms of enhanced processing overheads such as reduced throughput alongside with increased latency. To solve this problem, the FFNN-based intrusion detection and prevention could secure a high detection accuracy of 98% at a processing time of 0.53 seconds that allows near real-time threat response. In contrast to conventional security means, we enable FFNNs to continuously adapt to changing attack behaviors, making them intelligent to mount an adaptive defense against advanced cyber threats. From the experimental findings, we can conclude that the integrated FFNN approach enhances the robustness of VPN systems without imposing serious performance degradation, leading to its applicability to sensitive and large-scale environments such as academic networks or enterprise networks. The future work would be integrating FFNNs and other deep learning models such as LSTM or CNN for detection, extending its applications for real-time in multispecies (by considering different species of IoT) over the heterogeneous network, and integrating ZigBee or federated learning to provide scalability, efficiency, and privacy-preserving ability toward the next-generation VPN infrastructures.

Conflict of Interest

The authors declare that there is no conflict of interest.

Funding

This article does not contain any funding

Acknowledgment

The author would like to express gratitude to the institution for their invaluable support throughout this research project.

References

[1] D. Zhou, W. Zhang, Y. Tian, S. Kong, and M. Ren, "Research on the method of improving long-distance link transmission rate based on TCP window size," in *Proc. 2021 IEEE 5th Adv. Inf. Technol., Electron. Autom. Control Conf. (IAEAC)*, vol. 5, pp. 538–541, Mar. 2021.

[2] F. Kuntke, M. Sinn, and C. Reuter, "Reliable data transmission using low power wide area networks (LPWAN) for agricultural applications," in *Proc. 16th Int. Conf. Availability, Reliability and Security*, pp. 1–9, Aug. 2021.

[3] O. Salman, I. H. Elhajj, A. Kayssi, and A. Chehab, "A review on machine learning–based approaches for Internet traffic classification," *Ann. Telecommun.*, vol. 75, no. 11, pp. 673–710, 2020.

[4] A. Azab, M. Khasawneh, S. Alrabaee, K. K. R. Choo, and M. Sarsour, "Network traffic classification: Techniques, datasets, and challenges," *Digit. Commun. Netw.*, vol. 10, no. 3, pp. 676–692, 2024.

[5] A. Tousi and M. Luján, "Comparative analysis of machine learning models for performance prediction of the spec benchmarks," *IEEE Access*, vol. 10, pp. 11994–12011, 2022.

[6] M. Uğurlu, İ. A. Doğru, and R. S. Arslan, "A new classification method for encrypted internet traffic using machine learning," *Turk. J. Electr. Eng. Comput. Sci.*, vol. 29, no. 5, pp. 2450–2468, 2021.

[7] A. Jenefa and M. BalaSingh Moses, "A multi-phased statistical learning-based classification for network traffic," *J. Intell. Fuzzy Syst.*, vol. 40, no. 3, pp. 5139–5157, 2021.

[8] A. Pavano, *REAL-TIME IDENTIFICATION OF VoIP APPLICATIONS TRAFFIC*, Doctoral dissertation, Politecnico di Torino, 2020.

[9] R. Kumar, M. Swarnkar, G. Singal, and N. Kumar, "IoT network traffic classification using machine learning algorithms: An experimental analysis," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 989–1008, 2021.

[10] M. S. Sheikh and Y. Peng, "Procedures, criteria, and machine learning techniques for network traffic classification: A survey," *IEEE Access*, vol. 10, pp. 61135–61158, 2022.

[11] W. Zou, M. Han, and S. Hu, "Intrusion detection in industrial control systems based on deep learning: A review," *J. Adv. Manuf. Syst.*, vol. 20, no. 1, pp. 1–16, 2021.

[12] Y. Wang, X. Zheng, and L. Zhang, "Deep learning-based intrusion detection with adversaries," *IEEE Trans. Netw. Serv. Manag.*, vol. 17, no. 4, pp. 2540–2553, 2020.

[13] X. Wu and W. Zhang, "Enhancing VPN security through AI and deep learning techniques," *Int. J. Comput. Appl.*, vol. 177, no. 19, pp. 10–15, 2019.

[14] L. Huang and Q. Chen, "A comprehensive survey on deep learning-based network traffic classification," *IEEE Commun. Surv. Tutor.*, vol. 24, no. 1, pp. 209–232, 2022.

[15] C. Lee and J. Kim, "Deep packet inspection with recurrent neural networks," in *Proc. 2018 IEEE Int. Conf. Big Data*, pp. 2445–2450.

[16] T. M. Mitchell, *Machine Learning*. New York, NY, USA: McGraw Hill, 1997.

[17] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.

[18] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. Hoboken, NJ, USA: Wiley, 2000.

[19] B. Mor, S. Garhwal, and A. Kumar, "A systematic review of hidden Markov models and their applications," *Arch. Comput. Methods Eng.*, vol. 28, pp. 1429–1448, 2021.

[20] G. Gan, C. Ma, and J. Wu, *Data Clustering: Theory, Algorithms, and Applications*. Philadelphia, PA, USA: SIAM, 2020.

[21] F. D. Malliaros, C. Giatsidis, A. N. Papadopoulos, and M. Vazirgiannis, "The core decomposition of networks: Theory, algorithms and applications," *VLDB J.*, vol. 29, no. 1, pp. 61–92, 2020.

[22] G. James, D. Witten, T. Hastie, R. Tibshirani, and J. Taylor, "Statistical learning," in *An Introduction to Statistical Learning: With Applications in Python*, Cham, Switzerland: Springer, 2023, pp. 15–67.

[23] S. Sugahara, K. Kato, and M. Ueno, "Learning Bayesian network classifiers to minimize the class variable parameters," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 18, pp. 20540–20549, Mar. 2024.

[24] X. Chai, H. Wang, X. Ji, and L. Wang, "Identification of switched linear systems based on expectation-maximization and Bayesian algorithms," *Trans. Inst. Meas. Control*, vol. 43, no. 2, pp. 412–420, 2021.

[25] A. E. Ezugwu et al., "A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects," *Eng. Appl. Artif. Intell.*, vol. 110, p. 104743, 2022.

[26] H. Peng et al., "Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting," *Inf. Sci.*, vol. 521, pp. 277–290, 2020.

[27] G. Ravikumar, D. Ameme, S. Misra, S. Brahma, and R. Tourani, "iCASM: An information-centric network architecture for wide area measurement systems," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3418–3427, 2020.

[28] R. Kumar, M. Swarnkar, G. Singal, and N. Kumar, "IoT network traffic classification using machine learning algorithms: An experimental analysis," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 989–1008, 2021.
[29] W. Zheng, C. Gou, L. Yan, and S. Mo, "Learning to classify: A flow-based relation network for encrypted traffic classification," in *Proc. Web Conf.*, Apr. 2020, pp. 13–22.
[30] A. A. Afuwape, Y. Xu, J. H. Anajemba, and G. Srivastava, "Performance evaluation of secured network traffic classification using a machine learning approach," *Comput. Stand. Interfaces*, vol. 78, p. 103545, 2021.
[31] M. Lotfollahi, M. Jafari Siavoshani, R. Shirali Hossein Zade, and M. Saberian, "Deep packet: A novel approach for encrypted traffic classification using deep learning," *Soft Comput.*, vol. 24, no. 3, pp. 1999–2012, 2020.
[32] A. Jenefa and M. BalaSingh Moses, "A multi-phased statistical learning-based classification for network traffic," *J. Intell. Fuzzy Syst.*, vol. 40, no. 3, pp. 5139–5157, 2021.
[33] A. Pavano, *REAL-TIME IDENTIFICATION OF VoIP APPLICATIONS TRAFFIC*, Doctoral dissertation, Politecnico di Torino, 2020.
[34] S. Soleymanpour, H. Sadr, and H. Beheshti, "An efficient deep learning method for encrypted traffic classification on the web," in *Proc. 2020 6th Int. Conf. Web Res. (ICWR)*, Apr. 2020, pp. 209–216.