*Journal of Intelligent Decision Support System (IDSS) 7 (2) (2024), pp. 113-120*

Published by: Institute of Computer Science (IOCS)

Journal of Intelligent Decision Support System (IDSS)

Journal homepage: www.idss.iocspublisher.org

# Anomaly detection in network security systems using machine learning

Nughroho Adhi Santoso[1], Rezi Lutfayza[2], Bangkit Indarmawan Nughroho[3], Gunawan Gunawan[4]

[1] *Information System, STMIK YMI TEGAL, Indonesia*
[2] *Informatics Engeneering, STMIK YMI TEGAL, Indonesia*

**Article Info**

***Article history:***

Received May 30, 2024
Revised Jun 04, 2024
Accepted Jun 09, 2024

***Keywords:***

Algoritma Naïve Bayes;
Anomaly Detection;
Cross-Validation;
Machine Learning;
Network Security;

**ABSTRACT**

Anomaly Detection in Network Security Systems Using Machine Learning highlights the importance of developing effective models for data security. This research aims to develop an adaptive and automated anomaly detection model using the Naive Bayes algorithm and cross-validation. The methodology applied includes security log data collection, data preprocessing, implementation of Naive Bayes algorithms, and model evaluation using metrics such as accuracy, precision, recall, and F1-score. The results showed that the developed model was able to achieve high accuracy in detecting anomalies, with significant performance in identifying real threats without negative errors. The implication of this research is the improvement of network security through the application of machine learning, providing practical solutions for practitioners to deal with increasingly complex cybersecurity challenges.



***Corresponding Author:***

Rezi Lutfayza,
Informatics Engineering,
STMIK YMI TEGAL,
#1 Pendidikan Street, Tegal City, Central Java 52142, Indonesia.
Email: reziilutfayzaa03@gmail.com

## Introduction

In today's digital era, network security is one of the important aspects in maintaining the integrity and confidentiality of data in information systems (Saura et al., 2022). The increasing number of increasingly sophisticated cyberattacks requires an effective approach to detect and prevent threats before they compromise systems (Trim & Lee, 2021). One of the main challenges in network security is the detection of anomalies, which are activities that deviate from the normal behavior of the network that could indicate an attack (Wang & Zhu, 2022). Traditional approaches to detecting anomalies often rely on manually defined rules and cannot adapt to changing attack patterns (Martins et al., 2022). Therefore, there is an urgent need to develop more adaptive and automated methods for detecting anomalies in networks (Erhan et al., 2021). The use of machine learning in anomaly detection offers a promising solution with its ability to learn from data and identify unusual patterns without the need for rigid rule definitions (Pang et al., 2021). However, the selection of the right algorithm and evaluation method is the main key in increasing the effectiveness of detection (Liu et al., 2020). In this context, Naive Bayes algorithms and cross-validation evaluation methods were selected based on their ability to process big data and provide accurate predictions in previous studies (Lee et al., 2021).

The Naive Bayes algorithm is a classification method based on Bayes' Theorem assuming independence between features. This algorithm works by calculating the posterior probability of a target class given a specific feature value. The process involves calculating the prior probability of each class, the probability or probability of a given feature of a particular class, and the posterior probability by multiplying the prior and likelihood. Naive Bayes was chosen to detect anomalies in this study because of several advantages: simplicity and efficiency in terms of computation suitable for large datasets, good generalization ability even though the assumption of feature independence is not fully met, and high accuracy on various classification problems including anomaly detection in network security.

This research was conducted to respond to the problem of detecting anomalies in network security systems using a machine learning approach (Nassif et al., 2021). Through analysis of security log data, this research aims to develop models that can detect anomalies effectively (Sarker, 2021). This issue is important to discuss considering the increasing and complex impact of cyberattacks, so it requires solutions that are able to adapt to the latest threat dynamics (Safitra et al., 2023).

This research proposes innovations in the application of Naive Bayes algorithms for anomaly detection in network security systems, using cross-validation evaluation methods to ensure the reliability and validity of models (Uzun & Ballı, 2022) . It is hoped that this research will contribute to filling the gaps of previous research by providing new insights into the effectiveness of machine learning approaches, particularly Naive Bayes, in the context of network security (Shaukat et al., 2020). In addition, this research also has the potential to provide a basis for the development of more adaptive and automated network security strategies (Wu et al., 2020).

This research is expected to not only enrich the academic literature in the field of network security and machine learning, but also provide practical solutions for practitioners in the face of increasingly complex anomaly detection challenges (Moustafa et al., 2023). Through an interdisciplinary approach that combines expertise in the field of informatics and networking, this research aims to offer new perspectives and improve the effectiveness of network security systems in identifying and responding to cyber threats in a timely manner (DeGraba et al., 2021).

Previous research relevant to anomaly detection using the Naive Bayes algorithm showed significant results. Developed a selective Naive Bayes algorithm that shows high accuracy in big data classification (Chen et al., 2020). Proposed an anomaly detection framework for cybersecurity data using machine learning methods, although the main focus is not on Naive Bayes (Evangelou & Adams, 2020). Highlight challenges in cybersecurity on smart networks and potential solutions including machine learning methods such as Naive Bayes (Gunduz & Das, 2020).

However, there are some gaps in previous studies that drive the need for this study. Many studies focus more on deep learning methods or other complex algorithms, while simpler and faster methods such as Naive Bayes have received less attention (Kravchik & Shabtai, 2021). In addition, some studies did not use cross-validation methods which are important to improve the reliability and validity of the model (Tama & Lim, 2021). Finally, some previous methods were inefficient in handling large dynamic datasets, while Naive Bayes can provide a more practical and efficient solution (Bagaa et al., 2020). Therefore, this research aims to fill this gap by applying Naive Bayesian algorithms and cross-validation methods in detecting anomalies in network security systems, which is expected to provide a more adaptive and automated solution to increasingly complex cybersecurity challenges.

## Method

### Research Design

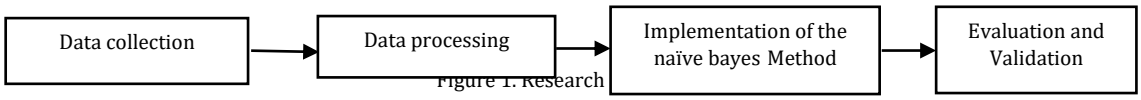


Figure 1. Research

Figure 1. Explaining the flow of research, this study begins with data collection from the network security system which includes the speed of information and actions taken, after which data processing is carried out to identify patterns and trends of potential anomalies. In the machine learning implementation phase, the naïve bayes method is applied for classification and anomaly detection. Evaluation is performed to measure the performance of the anomaly detection system, and the results are used to validate the effectiveness of the methods used. This research aims to improve network security through the application of machine learning in anomaly detection, the effectiveness of machine learning algorithm users in detecting network security anomalies.

**Data Collection**

The data collected includes username, ip address, time, action performed, traffic status and speed, bytes sent, bytes received, connection duration (seconds), protocol, and failed login attempts. For the type of data captured, namely the speed and actions carried out in the network, then perform data processing to identify patterns and trends that can be an indication of potential anomalies. The data set in table 1.

Table 1. Data Log

| Username | Ip | Time | Activity | Status | Speed (mbps) | Duration (detik) | bytes sent | bytes received | protocol | Login attempt failed |
|---|---|---|---|---|---|---|---|---|---|---|
| Admin | 192.168.1.1 | 08/12/2024 10:45 | Normal | Active | 10.34 | 360 | 1200 | 800 | TCP | 2 |
| User 1 | 192.168.1.2 | 21/09/2023 22:30 | Normal | Active | 05.12 | 210 | 2600 | 1300 | HTTP | 10 |
| User 2 | 192.168.1.3 | 12/10/2022 01:50 | Abnormal | Active | 1015.10 | 15 | 48000 | 50000 | UDP | 52 |
| User 3 | 192.168.1.4 | 30/01/2021 08:15 | Normal | Active | 24.53 | 170 | 1400 | 100 | TCP | 8 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| User 17 | 10.10.10.2 | 05/02/2024 09:00 | Abnormal | Active | 2208.45 | 39 | 73000 | 90000 | UDP | 89 |
| User 18 | 172.16.0.1 | 22/09/2022 20:30 | Abnormal | Active | 1175.37 | 23 | 68000 | 180000 | UDP | 90 |
| User 19 | 20.30.40.1 | 24/07/2019 02:35 | Normal | Active | 27.82 | 150 | 2400 | 1200 | HTTPS | 10 |
| User 20 | 172.16.1.3 | 27/10/2020 04:25 | Normal | Active | 58.04 | 263 | 1600 | 1300 | TCP | 17 |

The dataset includes various variables that are used to monitor network activity. These variables include username which is the name of the user performing the activity, Ip is the IP address from which the activity originated, time is the time when the activity was performed, and activity which indicates the action performed by the user. In addition, there is status which describes the current status of the activity recorded, and speed (mbps), which indicates the speed of the data traffic. Duration (seconds) records the length of time the connection lasted, while bytes sent and bytes received record the amount of data sent and received. Protocol indicates the network protocol used,

and failed login attempts records the number of unsuccessful login attempts. Each of these variables is critical for analysis in identifying patterns and trends that could be indicative of anomalies in the network (Javaheri et al., 2023).

**Data Pre-processing**

In the data preprocessing phase, a series of steps are performed that include data format conversion, label cleaning, and data type transformation to ensure dataset uniformity and consistency. This process involves setting time columns, adjusting IP address formats, and merging various features that have already been processed. In addition, we normalize numerical features through standardization to ensure that machine learning algorithms can work effectively, especially on methods that are sensitive to data scale.

**Variable Selection and Model Optimization**

The variables used are the action performed and the speed. Model optimization is carried out by using naïve bayes algorithms to determine normal and abnormal speeds, and ensure that the model can classify data effectively and efficiently, and has good capabilities when faced with new data.

**Algorithm Implementation**

The naïve bayes algorithm is implemented for its efficiency in handling data sets and its ability to provide rapid predictive results, as well as its maximum optimization and effectiveness in detecting and responding to new threats.

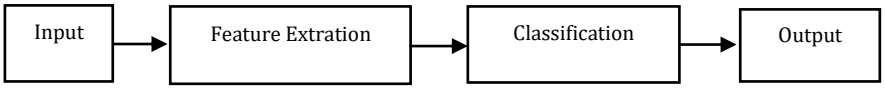


Figure 2. Research Flow

Based on Figure 2. Anomaly detection in network security systems using machine learning starts with the collection of network data, then relevant features are extracted to distinguish normal and suspicious activities. In addition, its features are classified as anomalous or not using naïve bayes algorithms, thus helping cybersecurity analysts to recognize potential threats early on and effectively protect the network. The formula of the naïve bayes algorithm is as in equation (1).

$$P(x|C_k) = \prod_{i=1}^{n} P(x_i|c_k) \quad (1)$$

Where is the prior probability of the class is the probability of the feature vector remembering the class and is the proof, i.e. the scaling factor that ensures the total probability of totaling one, is calculated as $P(c_k)c_k, P(x|C_k)xc_k, P(x)P(x) = \sum_{j=1}^{k} P(C_J)P(x|C_J)$.

**Algorithm Configuration and Model Evaluation**

The configuration of the algorithm in this study is an important key in ensuring accuracy in anomaly detection. The Naïve Bayes algorithm is configured to process pre-processed data by utilizing features such as speed and the type of actions performed within the network. The selection of algorithm parameters, such as prebabilities and conditional distributions, is set to match the characteristics of the network security dataset. Model evaluation is done through reviewing performance metrics such as accuracy, precision, recall, and f1-score as in equations (2), (3), (4), and (5).

$$akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Where TP models positive classes, TN models negative classes, FP predicts negative classes as positive classes, and FN predicts positive classes as negative classes.

$$Presisi = \frac{TP}{TP + Fp} \tag{3}$$

Where FP adds the overall number of type I (false positives) in the denominator, precision becomes an important metric when the cost of false positives is high.

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

Where FN adds the number of type II errors (false negatives) in the denominator, recall is an important metric for capturing all positive cases.

$$F1 - Score = 2 \times \frac{Presisi \times Recall}{Presisi + Recall} \tag{5}$$

Here the F1-Score reaches the highest value at 1 (precision and perfect memory) and the lowest at 0 (precision or zero memory).

Through a rigorous evaluation process, models can be validated and calibrated to improve reliability in threat detection. Using this experimental approach, the study aims not only to achieve high detection performance, but also to explore how algorithm configuration and model evaluation can be continuously optimized in the face of dynamic and evolving network security threat trends.

## Results and Discussions

### Data Pre-processing

In the results and discussion section, data preprocessing steps were successfully applied to the dataset used in this study. As seen in the attached image, the 'Time' column has been converted to timestamp format, facilitating time-based analysis. IP addresses have also been broken down and converted into numerical form, with each IP address segment represented as a separate column, allowing further processing with machine learning algorithms. Furthermore, cleaning labels in the 'Actions performed' column and combining various processed features, including speed, demonstrated success in creating a uniform and consistent dataset. The normalization process using StandardScaler on numeric features has also been berhasil dilakukan, yang krusial untuk efektivitas model Naive Bayes, yang sensitif against the scale of the data. These steps ensure that the data is ready to be modeled, with optimal settings to efficiently detect anomalies in the network. As for the results of Data Pre-processing in table 2.

Table 2. Data Preprocessing Results

| Time | Speed (mbps) | Ip_part_1 | Ip_part_2 | Ip_part_3 | Ip_part_4 | Actions performed |
|---|---|---|---|---|---|---|
| 1733654700 | 10.34 | 192 | 168 | 1 | 1 | Normal |
| 1695335400 | 5.12 | 192 | 168 | 1 | 1 | Normal |
| 1665539400 | 1015.10 | 192 | 168 | 1 | 1 | Abnormal |
| 1611994500 | 24.53 | 192 | 168 | 1 | 1 | Normal |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 1707123600 | 2208.45 | 10 | 10 | 10 | 2 | Abnormal |
| 1663878600 | 1157.37 | 172 | 16 | 0 | 1 | Abnormal |
| 1563935700 | 27.82 | 20 | 30 | 40 | 1 | Normal |
| 1603772700 | 58.04 | 172 | 16 | 1 | 3 | Normal |

**Variable Selection and Model Optimization**

The selection of variables such as 'action performed' and 'velocity', as well as the use of Naive Bayes algorithms, have been successful in classifying velocity as normal or abnormal. The model optimization process involves implementing StandardScaler to normalize features, a crucial step given the sensitivity of the Naive Bayes algorithm to feature scale. The 10-fold cross-validation test yields a perfect accuracy of 1.0, which indicates the consistency and reliability of the model in classifying new data. The data is divided into training sets and testing sets, with 30% of the total 21 data, or about 7 data, allocated as testing sets. Model training on the training set has verified the model's ability to generalize from available data.

**Algorithm Implementation**

The implementation of the Naive Bayes algorithm shows significant effectiveness in the analysis of network security systems. The algorithm was chosen for its efficiency in handling large data sets and its ability to provide rapid predictions, which are critical aspects of effectively detecting and responding to new threats. The results of this implementation confirm that Naive Bayes can efficiently classify and distinguish between normal and suspicious network activity, enabling rapid response to potential threats and improving overall network security.

**Algorithm Configuration and Model Evaluation**

After the implementation of the algorithm, the results of the model evaluation are obtained in table 3.

Table 3. Anomaly Detection Model Performance

| | precision | recall | F1 - score | support |
|---|---|---|---|---|
| Abnormal | 0.67 | 1.00 | 0.80 | 2 |
| Normal | 1.00 | 0.80 | 0.89 | 5 |
| Accuracy | | | 0.86 | 7 |
| Marco avg | 0.83 | 0.90 | 0.84 | 7 |
| Weighted avg | 0.90 | 0.86 | 0.86 | 7 |

The configuration of the Naïve Bayes algorithm adapted to features such as speed and type of actions performed in the network has shown satisfactory results in the detection of network anomalies. Evaluation of the model using accuracy, precision, recall, and F1-score performance metrics reveals success in data classification. The model achieves an overall accuracy of 86%, with a precision of 0.90 and a recall of 0.86 on a weighted average. Notably, the model managed to identify all abnormal cases with a recall of 1.00 but with lower precision in this category (0.67), showing some false positives. High F1-scores in the normal (0.89) and abnormal (0.80) categories indicate that the model is fairly balanced between precision and the ability to catch positive cases.

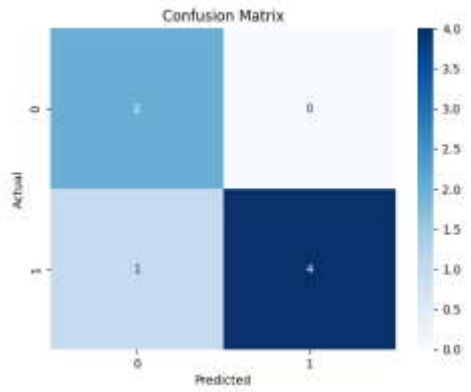


Figure 3. Fusion Matrix Results

Figure 3 shows a fusion matrix showing that the model successfully identified all abnormal cases correctly with no false negative error (FN), indicated by two correct abnormal predictions (TP) and the absence of abnormal cases misclassified as normal (FN). However, there is one false false positive (FP) error where a normal case is misclassified as abnormal. On the other hand, four normal cases were successfully classified correctly (TN).

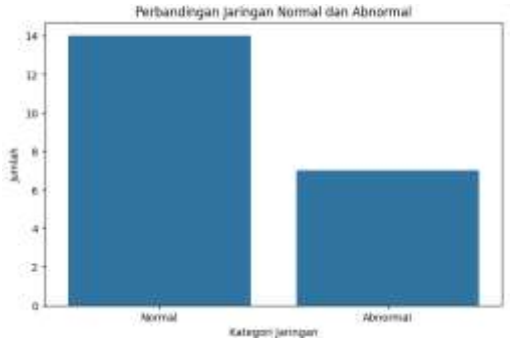


Figure 4. Comparison if normal tissue and abnormal tissue

Figure 4 shows a comparison graph of the amount of normal and abnormal tissue showing that normal tissue outweighs abnormal tissue, with 14 cases of normal tissue and 7 cases of abnormal tissue. This information provides deeper insights into model performance in the context of actual data and highlights potential areas for improved precision in detecting abnormal events. Parameter adjustment and conditional distribution have proven effective in detecting threats, with opportunities for further optimization in the face of dynamic and evolving threats.

The results of this study show that the anomaly detection model using Naive Bayes' algorithm and cross-validation is able to achieve a high level of accuracy with a good balance of precision and recall. Compared with fuzzy logic-based anomaly detection methods that require complex computing and machine learning models that lack the focus on simple algorithms, this research offers a more practical and efficient solution. The deep learning method shows high performance but requires large computing resources, while this study shows that the simpler Naive Bayes method is also effective. Thus, this research successfully fills the gap by offering a fast, lightweight, and cross-validation method that improves the reliability of the model, thus making a significant contribution to the development of a more adaptive and automated network security strategy.

## Conclusions

This study develops an effective anomaly detection model in the network security system using Naive Bayesian algorithm and cross-validation. The evaluation results show that this model achieves high accuracy with precision and balanced memory, effectively identifying suspicious network activity. Theoretically, this study enriches the literature by proving the effectiveness of Naive Bayes in anomaly detection. Practically, this model offers a quick and efficient solution to improve data security and response to cyber threats. This module should be integrated with a broader security strategy, demonstrate the potential for rapid adaptation to new threats, and make a significant contribution to cyber objectives and open up opportunities for future research to improve readiness to respond to security incidents.

## Reference

Chen, S., Webb, G. I., Liu, L., & Ma, X. (2020). A novel selective naïve Bayes algorithm. *Knowledge-Based Systems*, *192*, 105361. https://doi.org/https://doi.org/10.1016/j.knosys.2019.105361

DeGraba, T. J., Williams, K., Koffman, R., Bell, J. L., Pettit, W., Kelly, J. P., Dittmer, T. A., Nussbaum, G., Grammer, G., & Bleiberg, J. (2021). Efficacy of an interdisciplinary intensive outpatient program in treating combat-related traumatic brain injury and psychological health conditions. *Frontiers in Neurology*, *11*, 580182. https://doi.org/https://doi.org/10.3389/fneur.2020.580182

Erhan, L., Ndubuaku, M., Di Mauro, M., Song, W., Chen, M., Fortino, G., Bagdasar, O., & Liotta, A. (2021). Smart anomaly detection in sensor systems: A multi-perspective review. *Information Fusion*, *67*, 64–79. https://doi.org/https://doi.org/10.1016/j.inffus.2020.10.001

Javaheri, D., Gorgin, S., Lee, J. A., & Masdari, M. (2023). Fuzzy logic-based DDoS attacks and network traffic anomaly detection methods: Classification, overview, and future perspectives. *Information Sciences*, *626*, 315–338. https://doi.org/10.1016/j.ins.2023.01.067

Lee, M., Kwon, W., & Back, K.-J. (2021). Artificial intelligence for hospitality big data analytics: developing a prediction model of restaurant review helpfulness for customer decision-making. *International Journal of Contemporary Hospitality Management*, *33*(6), 2117–2136. https://doi.org/https://doi.org/10.1108/IJCHM-06-2020-0587

Liu, K., Xu, S., Xu, G., Zhang, M., Sun, D., & Liu, H. (2020). A review of android malware detection approaches based on machine learning. *IEEE Access*, *8*, 124579–124607. https://doi.org/https://doi.org/10.1109/ACCESS.2020.3006143

Martins, I., Resende, J. S., Sousa, P. R., Silva, S., Antunes, L., & Gama, J. (2022). Host-based IDS: A review and open issues of an anomaly detection system in IoT. *Future Generation Computer Systems*, *133*, 95–113. https://doi.org/https://doi.org/10.1016/j.future.2022.03.001

Moustafa, N., Koroniotis, N., Keshk, M., Zomaya, A. Y., & Tari, Z. (2023). Explainable intrusion detection for cyber defences in the internet of things: Opportunities and solutions. *IEEE Communications Surveys & Tutorials*. https://doi.org/https://doi.org/10.1109/COMST.2023.3280465

Nassif, A. B., Talib, M. A., Nasir, Q., & Dakalbab, F. M. (2021). Machine learning for anomaly detection: A systematic review. *Ieee Access*, *9*, 78658–78700. https://doi.org/https://doi.org/10.1109/ACCESS.2021.3083060

Pang, G., Shen, C., Cao, L., & Hengel, A. Van Den. (2021). Deep learning for anomaly detection: A review. *ACM Computing Surveys (CSUR)*, *54*(2), 1–38. https://doi.org/https://doi.org/10.1145/3439950

Safitra, M. F., Lubis, M., & Fakhrurroja, H. (2023). Counterattacking cyber threats: A framework for the future of cybersecurity. *Sustainability*, *15*(18), 13369. https://doi.org/https://doi.org/10.3390/su151813369

Sarker, I. H. (2021). CyberLearning: Effectiveness analysis of machine learning security modeling to detect cyber-anomalies and multi-attacks. *Internet of Things*, *14*, 100393. https://doi.org/https://doi.org/10.1016/j.iot.2021.100393

Saura, J. R., Ribeiro-Soriano, D., & Palacios-Marqués, D. (2022). Evaluating security and privacy issues of social networks based information systems in Industry 4.0. *Enterprise Information Systems*, *16*(10–11), 1694–1710.

Shaukat, K., Luo, S., Varadharajan, V., Hameed, I. A., & Xu, M. (2020). A survey on machine learning techniques for cyber security in the last decade. *IEEE Access*, *8*, 222310–222354. https://doi.org/https://doi.org/10.1109/ACCESS.2020.3041951

Trim, P. R. J., & Lee, Y. I. (2021). The global cyber security model: Counteracting cyber attacks through a resilient partnership arrangement. *Big Data and Cognitive Computing*, *5*(3). https://doi.org/10.3390/bdcc5030032

Uzun, B., & Ballı, S. (2022). A novel method for intrusion detection in computer networks by identifying multivariate outliers and ReliefF feature selection. *Neural Computing and Applications*, *34*(20), 17647–17662. https://doi.org/https://doi.org/10.1007/s00521-022-07402-2

Wang, C., & Zhu, H. (2022). Wrongdoing monitor: A graph-based behavioral anomaly detection in cyber security. *IEEE Transactions on Information Forensics and Security*, *17*, 2703–2718.

Wu, H., Han, H., Wang, X., & Sun, S. (2020). Research on artificial intelligence enhancing internet of things security: A survey. *Ieee Access*, *8*, 153826–153848. https://doi.org/https://doi.org/10.1109/ACCESS.2020.3018170Bagaa, M., Taleb, T., Bernabe, J. B., & Skarmeta, A. (2020). A machine learning security framework for iot systems. *IEEE Access*, *8*, 114066–114077. https://doi.org/https://doi.org/10.1109/ACCESS.2020.2996214

Chen, S., Webb, G. I., Liu, L., & Ma, X. (2020). A novel selective naïve Bayes algorithm. *Knowledge-Based Systems*, *192*, 105361. https://doi.org/https://doi.org/10.1016/j.knosys.2019.105361

Evangelou, M., & Adams, N. M. (2020). An anomaly detection framework for cyber-security data. *Computers & Security*, *97*, 101941. https://doi.org/https://doi.org/10.1016/j.cose.2020.101941

Gunduz, M. Z., & Das, R. (2020). Cyber-security on smart grid: Threats and potential solutions. *Computer Networks*, *169*, 107094. https://doi.org/https://doi.org/10.1016/j.comnet.2019.107094

Kravchik, M., & Shabtai, A. (2021). Efficient cyber attack detection in industrial control systems using lightweight neural networks and pca. *IEEE Transactions on Dependable and Secure Computing*, *19*(4), 2179–2197. https://doi.org/https://doi.org/10.1109/TDSC.2021.3050101

Tama, B. A., & Lim, S. (2021). Ensemble learning for intrusion detection systems: A systematic mapping study and cross-benchmark evaluation. *Computer Science Review*, *39*, 100357. https://doi.org/https://doi.org/10.1016/j.cosrev.2020.100357