



Iterative Learning-Based Intrusion Detection System for Performance Enhancement in Imbalanced Data Environments

Yu-Ran Jeon¹^a and Il-Gu Lee^{1,2}^b

¹Department of Future Convergence Technology Engineering, Sungshin Women's University, Seoul, 02844, South Korea

²Department of Convergence Security Engineering, Sungshin Women's University, Seoul, 02844, South Korea

Keywords: Challenges in Cybersecurity, Intrusion Detection Systems, Imbalanced Data Environments, Optimal Iterative Learning.

Abstract: To defend against advanced cyberattacks, various anomaly detection methods have been developed, including signature-based, machine learning (ML)-based, and tool-based approaches across multiple fields. The ML-based anomaly detection method analyzes the patterns of the input data and identifies malicious behavior using classifiers. However, the ML-based anomaly detection method faces the challenge of accurately distinguishing malicious behavior from benign behavior, and its performance is reduced in real-world environments because of the discrepancies between training and deployment environments. In this study, cybersecurity challenges were analyzed, focusing on intrusion detection systems (IDS) and the influence of ML performance degradation in imbalanced data environments. To counteract this performance degradation, an optimal iterative learning-based IDS is proposed that improves efficiency by approximately 24% compared to a conventional model.

1 INTRODUCTION


With the acceleration of digital transformation in society and industry through advances in the Internet of Things (IoT), cloud, wireless communications, and artificial intelligence technologies, cyberattacks are becoming increasingly diverse worldwide (Kilincer et al., 2021). In particular, the widespread deployment of IoT devices in technological fields, such as healthcare, agriculture, smart cities, and traffic management, has increased their vulnerability to cyberattacks. Advanced research is being conducted on detection and response technologies to mitigate the damage caused by such attacks. Signature-based anomaly detection methods commonly store data signatures collected from network traffic monitoring to identify benign behaviors. However, signature-based anomaly detection methods can only detect known attacks. In contrast, machine learning (ML)-based anomaly detection methods address these limitations by creating models that can identify unknown malicious behaviors (Han et al., 2023).


However, ML-based anomaly detection methods face the following challenges:

1) When benign and malicious behaviors are similar, it is difficult to distinguish between benign and malicious data.

2) Since malicious traffic occurs with a very low probability, there is a mismatch between the deployment of the model and the training environment.

As cyberattacks have become more diverse and complex, sophisticated attacks that mimic normal user behavior have increased. This makes it difficult to distinguish between malicious and benign behaviors, and creates areas where malicious and benign data distributions overlap. As the overlap increases, the performance of the model deteriorates more and more. In addition, malicious events are very unlikely to occur in real-world intrusion detection systems (IDS). A model trained on a balanced dataset is usually deployed in an IDS; however, malicious traffic is relatively rare in real-world environments. Consequently, the training data does not accurately represent the distribution of the deployment data,

^a <https://orcid.org/0009-0001-4149-9288>

^b <https://orcid.org/0000-0002-5777-4029>

which reduces the performance of the classifier (Kulkarni et al., 2020). The problem of imbalanced data becomes even more apparent when attackers actively modify malware to evade detection and pose a constant threat. In this study, an imbalanced data environment was modeled to analyze its effect on model performance. Furthermore, an optimal iterative learning-based IDS that performs iterative learning by incorporating new input data is proposed to address these challenges. This study offers the following major contributions:

- The challenges in ML-based anomaly detection were comprehensively analyzed.
- The data imbalance scenario was modeled by collecting data using an endpoint detection and response (EDR) tool, demonstrating the performance degradation in real-world IDS environments.
- An optimal iterative learning-based IDS that determines the optimal number of iterations to achieve efficient learning was proposed.

The remainder of this paper is organized as follows. Section 2 provides a research overview on ML-based anomaly detection and efforts to mitigate data imbalance in ML. Section 3 analyzes the challenges of ML-based IDS, and Section 4 proposes an optimal iterative learning method to address the IDS performance degradation caused by data imbalance. Finally, Section 5 concludes the paper and discusses future research directions.

2 RELATED WORK

With the rise of intelligent cyberattacks, ML-based anomaly detection for cybersecurity has recently become a research focus. However, conventional ML-based anomaly detection studies struggle to deploy the trained models in real-world environments because they often overlook the differences between training and deployment environments. In this section, existing ML-based anomaly detection research and its limitations are examined.

2.1 Machine Learning-Based Anomaly Detection

Abbasi et al. (2022) proposed a particle swarm optimization (PSO)-based clustering algorithm to overcome the limitations of behavior-based ransomware detection models. They classified the presence of attacks using random forest (RF) and achieved 97% accuracy in detecting attacks. However, this study shows significant performance degradation

when applied to multi-classification models. In addition, it lacks an F-score measurement, making it difficult to evaluate performance in environments with data imbalance.

Sun et al. (2022) proposed an intelligent attack detection technique based on a frequency-differential selection (FDS) feature selection algorithm and a weighting calculation. They collected attack and normal data from Android devices and achieved an accuracy of 99 %. However, this study did not consider the data-imbalance problem in ML, and the performance degraded when deployed in a real-world environment.

Gezer et al. (2019) applied four ML techniques to detect Trojan horse malware and selected the optimal hyperparameters and features. Using an RF classifier, the model achieved an accuracy of approximately 99.95%. However, it is difficult to say whether the model was tested in a reliable evaluation environment, as the authors measured the accuracy on an imbalanced dataset.

2.2 Mitigating the Data Imbalance in Cybersecurity

Thirumuruganathan et al. (2024) proposed a method to effectively detect and mitigate data imbalance even in environments with unlabeled data. Their method addresses unknown attacks and data imbalance problems in real-world environments by assigning pseudo-labels to unlabeled data through unsupervised learning. However, their method is limited in that performance degrades when there is a distributional difference between the test and training data. In addition, manual labeling was required for some data during the initial training stage.

Balla et al. (2023) improved performance degradation due to data imbalance in IDS and intrusion prevention systems (IPS). They classified imbalanced datasets into majority and minority classes by applying undersampling to the majority class data and oversampling to the minority class data. They demonstrated that their proposed method outperforms conventional methods by measuring the accuracy, precision, detection rate, and F-score for four public datasets. However, the proposed method has problems in detecting attacks in real time as it requires retraining to compensate for data imbalances. Moreover, it is difficult to counteract adversarial attacks as this study utilizes the sampling method.

Wang et al. (2021) proposed an oversampling method to overcome the challenges posed by the imbalance between attack and normal data in a network IDS. Their approach uniformly adjusts the

data distribution by oversampling minority class data and adaptively generates samples for minority data based on training difficulty. This study improved the detection performance of minority class data. However, their method relies on fixed thresholds for adaptive sample generation, which may limit their flexibility in responding to real network environments.

3 CHALLENGES IN MACHINE LEARNING-BASED ANOMALY DETECTION

ML offers high-quality intelligent services that increase user convenience. However, several challenges within ML environments can lead to performance degradation when the models are deployed in the real world. In this section, two major challenges that hinder the performance of ML-based anomaly detection models in real-world environments are examined.

3.1 Differentiating Benign and Malicious Behaviors

When malicious behavior closely resembles benign behavior, distinguishing between malicious and benign data becomes difficult. Adversaries often design attacks to mimic normal behavior as closely as possible to avoid detection by users or security

systems. For example, Trojan horse malware infiltrates systems by posing as normal user software and executing attacks in the background. Therefore, malicious traffic that closely resembles normal activity has been generated.

Figure 1 shows the data distribution of benign and malicious behaviors. The data were collected using Google Rapid Response (GRR), an EDR tool (GRR team, 2017a), in Ubuntu environments. The GRR server and client are deployed on a VMware virtual machine. Powershell ransomware simulator (PSRansom) (JoelGMSec, 2022), MSFVenom (Offensive Security, 2015) and MEMZ Trojan (Endermanch, 2020) were used to collect malicious data.

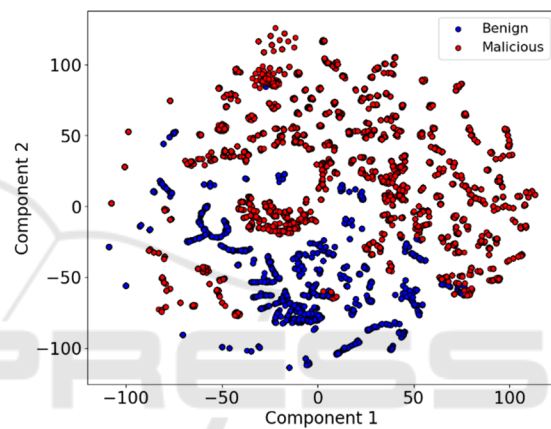


Figure 1: Visualization of the dataset using t-SNE.

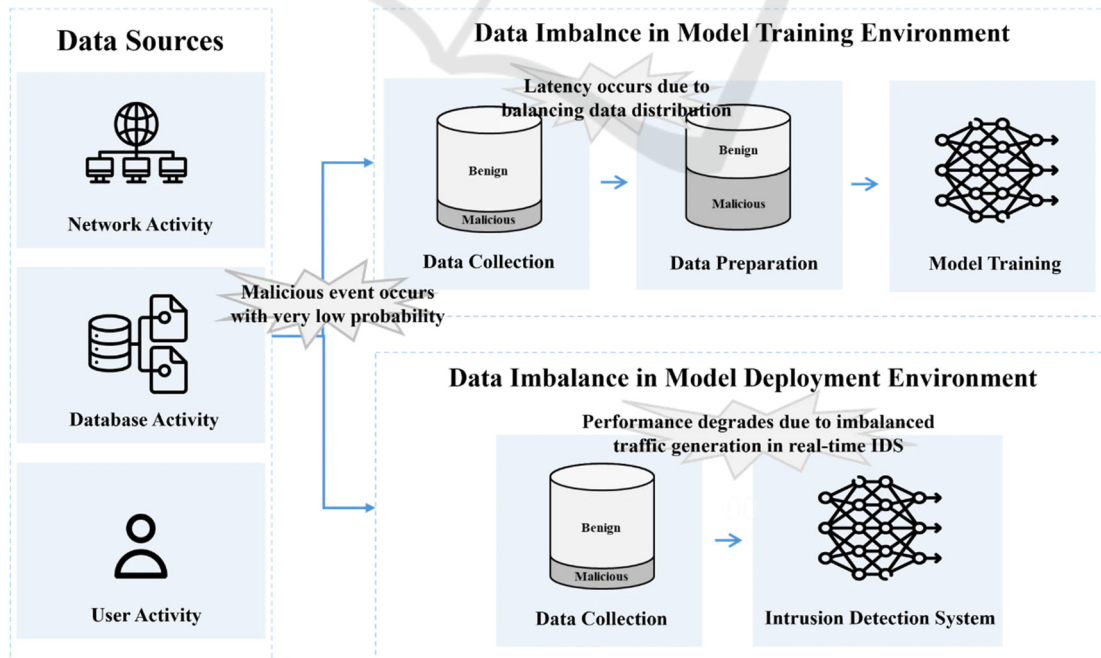


Figure 2: Data imbalance in the model training and deployment environments.

Using t-distributed stochastic neighbor embedding (t-SNE) (Van der Maaten and Hinton, 2008), the data were reduced to two dimensions, with benign and malicious data represented as blue and red circles, respectively. There is considerable overlap between benign and malicious behavior. As this area increases in size, it becomes increasingly difficult to distinguish between benign and malicious data. Consequently, there is a risk that benign data is incorrectly classified as an attack, or that an actual attack is not detected.

3.2 Data Imbalance in Model Training and Deployment Environments

Malicious behavior is very unlikely to occur in real networks. Consequently, data imbalance issues may arise when training a model or deploying it in a real-world IDS. Figure 2 illustrates the challenges that arise during model training and deployment.

Benign traffic is much more common than malicious traffic in real-world IDS environments, making it difficult to collect balanced training data. Collecting a sufficiently balanced dataset from an IDS can be time-consuming as malicious data is sparse. In addition, the data were balanced before training to improve model performance. However, benign behavior is much more likely to occur in a real-world environment than malicious behavior, resulting in significant discrepancies. Thus, even if the model performs well during testing, its performance often degrades in an actual deployment.

4 ENHANCING IMPROVING PERFORMANCE THROUGH OPTIMAL ITERATIVE LEARNING

In this section, the imbalanced data environment encountered in the IDS is modeled, and its effect on model performance is analyzed. In addition, an optimal iterative learning method has been proposed to improve the performance of the model when data imbalance leads to performance degradation.

4.1 Modeling the Imbalanced Data Environment

Data imbalance was modeled in the training and deployment environments and its effect on performance was assessed. Model performance was measured using the F-score, an indicator that is well

suited for evaluating the performance of imbalanced data. Malicious and benign data were collected using the GRR and the RF classification model was used as our ML algorithm. To quantify the imbalance of the data, the imbalanced data ratio was calculated using Equation (1) and the performance was measured as the imbalanced data ratio increased. The imbalanced data ratio represents the proportion of the majority class ($D_{majority}$) in a dataset. In a real-world IDS environment, the benign data forms the $D_{majority}$, whereas the malicious data forms the $D_{minority}$.

$$\begin{aligned} \text{Imbalanced data ratio (\%)} \\ &= \frac{D_{majority}}{D_{majority} + D_{minority}} \times 100 \end{aligned} \quad (1)$$

Figure 3 shows the results of varying the imbalanced data ratio for the entire dataset to simulate the data imbalance that occurs during model training in an IDS. The dataset was divided into a training and a test dataset in an 8:2 ratio to ensure that the data distribution remained consistent between the datasets. To reflect the real network conditions, the data imbalance scenarios in which benign traffic occurred with a probability ranging from 10% to 90% were assumed. The results show that the F-score decreases when the dataset is heavily skewed towards benign or malicious data.

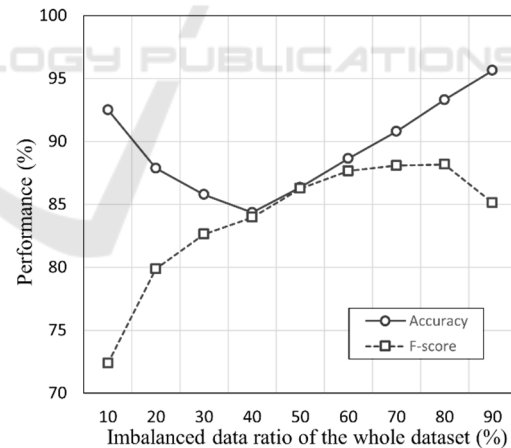


Figure 3: Accuracy and F-score vs. the imbalanced data ratio of the whole dataset.

Figure 4 illustrates the effect of varying the imbalanced data ratio within the test dataset. The data imbalance, that arises when a trained model is deployed in a real IDS environment, was modeled. The data distribution used for training differed from that used in the deployment environment, leading to an imbalanced data environment. During training, the ratio of benign to malicious data was set to 1:1. In the

deployment environment, model performance was measured as the proportion of benign traffic varied from 0% to 100%. As the proportion of benign traffic increased, the data imbalance worsened, resulting in a gradual decline in the F-score. In a highly imbalanced environment, where benign traffic has a 100% probability of occurring, the F-score dropped to 48.94%.

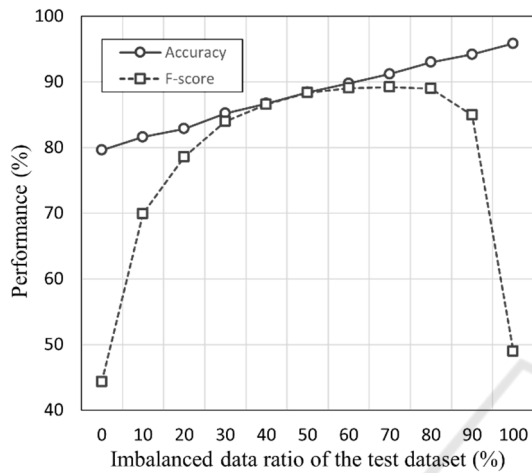


Figure 4: Accuracy and F-score vs. the imbalanced data ratio of the test dataset.

As shown in Figures. 3 and 4, the F-score deteriorates significantly as the ratio of imbalanced data increases. In contrast, the accuracy remains relatively high and gradually improves. Accuracy is sensitive to data distribution, making it an unsuitable performance metric for highly imbalanced environments. When the proportion of majority class data becomes very high, the ML model assigns a greater weight to this class, thus optimizing accuracy when majority class data is input (Thabtah et al., 2020). Consequently, performance decreases

significantly when minority class data is input and the number of false detections increases.

4.2 Optimal Intrusion Detection System Based on Iterative Learning

In this section, an optimal intrusion detection method based on iterative learning is proposed to improve the performance degradation of the model caused by data imbalance. The overall architecture of the proposed model is shown in Figure 5. When new data is provided through the IDS, the dataset is reconstructed and additional training is performed with an optimal number of iterations. When new data is added, the dataset is balanced by fixing the majority class volume and adding new data to the minority class. This iterative training helps the model to achieve a more balanced data distribution and improve its performance. However, this process also increases latency. Therefore, it is essential to determine the optimal number of iterations to achieve a balance between performance improvement and minimum latency.

To determine the optimal number of iterations, *Efficiency* was defined as the improvement in F-score per second. *Efficiency* was calculated by measuring the improvement in F-score and latency at each iteration and determining the rate of F-score improvement over time using Equation (2). The iteration refers to the number of additional training cycles applied to the model. Then, the optimal number of iterations (i^*) that maximizes *Efficiency* can be determined according to Equation (3), which enables efficient learning.

$$Efficiency = \frac{F - score}{Latency} \tag{2}$$

$$i^* = \arg \max_i Efficiency \tag{3}$$

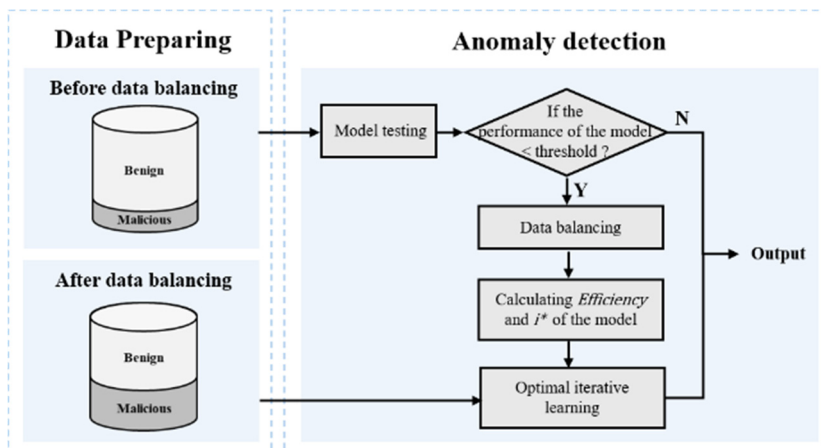


Figure 5: Flowchart of the proposed model.

Figure 6 shows a comparison of the *Efficiency* of the proposed and conventional models. The initial model was trained with imbalanced data at a benign-to-malicious data ratio of 99:1. For each iteration, the amount of data was increased by 1% of the total data, and the proposed model updated the minority-class data within the new input data. The proposed model performs iterative learning using the updated data, whereas the conventional model evaluates the performance of newly entered data without additional learning. In the initial model, the conventional and proposed models showed similar *Efficiency*. However, when iterative learning was applied, the proposed model improved the *Efficiency* of the IDS. The proposed model reached its optimal *Efficiency* at ten iterations, after which the *Efficiency* declined owing to increased latency. When the proposed model was trained for the optimal number of iterations, its performance improved by approximately 24% compared to that of the conventional model. Thus, when the minority class data is supplemented with only 10% of the total data, the *Efficiency* of the model may be maximized.

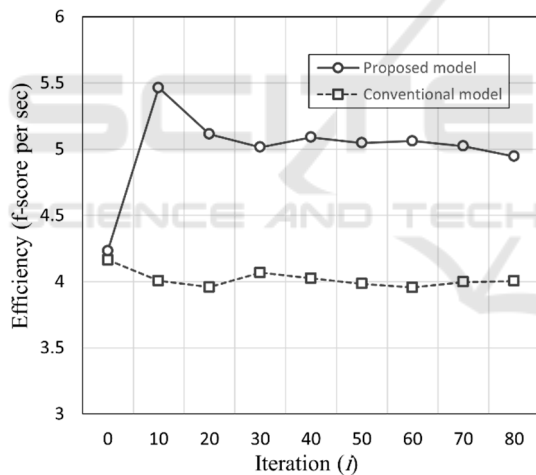


Figure 6: Efficiency vs. iteration.

Figure 7 compares the performance of the proposed model with that of a conventional model with varying data volumes. The performance was measured with i set to 10, as the proposed model showed the greatest improvement when ten iterations of learning were applied in imbalanced data environments. The F-scores of both the conventional and proposed models were compared as the data sampling ratio increased from 10% to 90%. As the data volume increases, the performance of the conventional model decreases significantly, whereas the performance of the proposed model gradually improves.

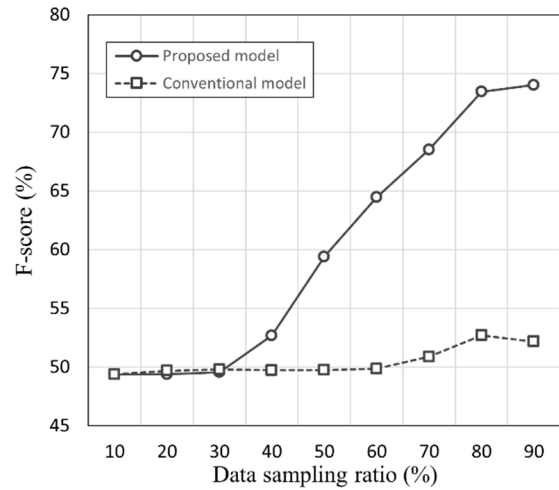


Figure 7: F-score vs. data sampling ratio.

5 CONCLUSIONS

As cyberattacks become more diverse and complex, research into ML-based detection of cyberattacks has gained significant momentum. However, intelligent attacks that mimic normal system behavior are becoming increasingly difficult to distinguish from benign activities in the cybersecurity landscape. Furthermore, data imbalance during model training and deployment degrades the model performance. This study analyzed the challenges in ML-based IDS and proposed an optimal iterative learning method to address performance degradation in imbalanced data environments. While this study determined the optimal number of learning iterations in an imbalanced data environment and improved efficiency, it did not evaluate performance in dynamic network scenarios. Future research will focus on minimizing the impact of data imbalance in IDS environments, where network conditions change dynamically.

ACKNOWLEDGEMENTS

This work is supported by the Ministry of Trade, Industry and Energy (MOTIE) under the Training Industrial Security Specialist for High-Tech Industry program (RS-2024-00415520) under the supervision of y the Korea Institute for Advancement of Technology (KIAT), and by the Ministry of Science and ICT (MSIT) under the ICAN (ICT Challenge and Advanced Network of HRD) program (IITP-2022-RS-2022-00156310) and Information Security Core

Technology Development (RS-2024-00437252) under the supervision of the Institute of Information & Communication Technology Planning & Evaluation (IITP).

Wang, Z., Jiang, D., Huo, L., & Yang, W. (2021). An efficient network intrusion detection approach based on deep learning. *Wireless Networks*, 1-14.

REFERENCES

- Abbasi, M. S., Al-Sahaf, H., Mansoori, M., & Welch, I. (2022). Behavior-based ransomware classification: A particle swarm optimization wrapper-based approach for feature selection. *Applied Soft Computing*, 121, 108744.
- Endermanch; 2020. MalwareDatabase [Online]. Available from: <https://github.com/Endermanch/MalwareDatabase/tree/master/trojans>.
- Balla, A., Habaebi, M. H., Elsheikh, E. A., Islam, M. R., & Suliman, F. M. (2023). The effect of dataset imbalance on the performance of SCADA intrusion detection systems. *Sensors*, 23(2), 758.
- Gezer, A., Warner, G., Wilson, C., & Shrestha, P. (2019). A flow-based approach for Trickbot banking trojan detection. *Computers & Security*, 84, 179-192.
- GRR Team; 2017a. What is GRR? [Online]. Available from: <https://grr-doc.readthedocs.io/en/latest/what-is-grr.html>.
- Han, D., Wang, Z., Chen, W., Wang, K., Yu, R., Wang, S., & Yin, X. (2023). Anomaly detection in the open world: Normality shift detection, explanation, and adaptation. *In NDSS*.
- JoelGMSSec; 2022. PSRansom [Online]. Available from: <https://github.com/JoelGMSSec/PSRansom>.
- Kilincer, I. F., Ertam, F., & Sengur, A. (2021). Machine learning methods for cyber security intrusion detection: Datasets and comparative study. *Computer Networks*, 188, 107840.
- Kulkarni, A., Chong, D., & Batarseh, F. A. (2020). Foundations of data imbalance and solutions for a data democracy. *In Data Democracy* (pp. 83-106). Academic Press.
- Offensive security; 2015. MSFVENOM. Available from: <https://www.offensive-security.com/metasploit-unleashed/Msfvenom/>.
- Sun, H., Xu, G., Wu, Z., & Quan, R. (2022). Android malware detection based on feature selection and weight measurement. *Intelligent Automation and Soft Computing*.
- Thabtah, F., Hammoud, S., Kamalov, F., & Gonsalves, A. (2020). Data imbalance in classification: Experimental evaluation. *Information Sciences*, 513, 429-441.
- Thirumuruganathan, S., Deniz, F., Khalil, I., Yu, T., Nabeel, M., & Ouzzani, M. (2024). Detecting and mitigating sampling bias in cybersecurity with unlabeled data. *In 33rd USENIX Security Symposium (USENIX Security 24)* (pp. 1741-1758).
- Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11).