# Green IoT Protection: Sustainability-Driven Machine Intelligence for Malware Defense

**Ayman H. Abdel-aziem[1*] , Tamer H. M. Soliman[2]**

[1]Faculty of Information Systems and Computer Science, October 6th University, Cairo, Egypt;

[2]Faculty of Computers and Informatics, Zagazig University, Zagazig 44519, Sharqiyah, Egypt

Emails: Ayman.Hasanein.comp@o6u.edu.eg; tamer.hasan.comp@o6u.edu.eg

**Abstract**

As the Internet of Things (IoT) continues to expand, the security of connected devices becomes a paramount concern. Malicious actors exploit vulnerabilities in these devices, leading to severe consequences such as data breaches, privacy infringements, and service disruptions. Traditional security measures struggle to keep pace with the evolving threat landscape, necessitating advanced solutions. In this paper, we present a pioneering approach to fortify the security of IoT environments against malware through the integration of advanced machine intelligence techniques. Our work addresses this critical concern by introducing a comprehensive Machine Intelligence Strategy designed to detect and classify malware in IoT ecosystem. Leveraging Support Vector Machines (SVM) with different kernel choices, our strategy offers a multi-faceted defense mechanism. Through extensive experimentation and evaluation on public dataset of malware images, we demonstrate the efficacy of our strategy in fortifying the guardianship of connected devices, fostering a safer and more resilient IoT ecosystem. Beyond technical contributions, our research fosters a deeper understanding of the symbiotic relationship between machine intelligence and IoT security, propelling advancements in safeguarding the ever-expanding landscape of interconnected devices.

**Keywords:** Internet of Things (IoT); Machine Intelligence; Malware Detection; Green IoT; Sustainability; Connected Devices; Artificial Intelligence; Sustainable Strategies.

## 1. Introduction

The rapid proliferation of the Internet of Things (IoT) has ushered in an era of unparalleled connectivity, revolutionizing how we interact with the world around us. IoT devices have seamlessly integrated into various aspects of our daily lives, offering unprecedented convenience and efficiency [1]. From smart homes and wearable devices to industrial automation and healthcare applications, the IoT has become an indispensable part of our modern society. However, as the number of connected devices continues to surge, so does the magnitude of security challenges [2]. The interconnectivity and diversity of IoT ecosystems create a complex attack surface, leaving them susceptible to exploitation by malicious actors. Cybercriminals have capitalized on the vulnerabilities in these devices, unleashing malware and sophisticated attacks that compromise user data, violate privacy, and disrupt critical services. The consequences of such security breaches can be devastating, posing a significant threat to individuals, organizations, and even the infrastructure that relies on the IoT [3].

Traditional security approaches have proven insufficient in thwarting the ever-evolving tactics employed by cyber adversaries. Static, rule-based defenses struggle to adapt to novel attack vectors, necessitating the adoption of more advanced and dynamic security solutions [4]. In this context, machine intelligence emerges as a beacon of hope, holding the potential to bolster the guardianship of connected devices and fortify the security of the entire IoT landscape [5].. In this paper, we present a comprehensive approach to bolstering the security of IoT environments against the rising threat of malware through the innovative integration of machine intelligence techniques. Our work addresses the critical need for enhanced protection in interconnected devices by proposing a robust strategy centered around a machine learning framework [14, 15]. The primary contribution of our research lies in the development and rigorous evaluation of a sophisticated Machine Intelligence Strategy that harnesses the capabilities of



Doi : https://doi.org/10.54216/JSDGT.020205

Support Vector Machines (SVM) with diffferent kernels. By fusing cutting-edge SVM classifiers with a public dataset of malware images, we showcase a novel solution that effectively identifies and classifies malicious activities in real-time, thus safeguarding IoT ecosystems from potential breaches [16, 17].

This paper is organized as follows. In Section II, we review the relevant literature and discuss the state-of-the-art in IoT security and machine intelligence for malware detection. Building upon this foundation, Section III presents our proposed methodology. In Section IV, we describe the experimental configurations used to evaluate the effectiveness of our approach, including the datasets, evaluation metrics, and comparative analyses with existing solutions. Section V presents the results and discussions, where we critically analyze the performance of our strategy and highlight its strengths and limitations. Finally, in Section VI, we draw conclusive remarks, summarizing the contributions of this paper.

## 2. Related Works

This section provides a comprehensive overview of the existing research and advancements in the fields of IoT security and machine intelligence for malware detection. By examining the latest literature and studies, we contextualize our proposed machine intelligence strategy within the broader landscape of IoT security solutions. Paricherla et al. [5] proposed a machine learning framework that addresses the unique challenges posed by IoT security. Their approach aimed to leverage the power of machine learning algorithms to analyze IoT data streams in real-time and detect anomalous patterns indicative of potential malware activity. Akhtar and Feng [6] delved into the application of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models for real-time malware detection. Their work showcases the potential of deep learning in efficiently extracting meaningful features from IoT data, enabling accurate classification of benign and malicious behavior. Baek et al. [7] proposed a two-stage hybrid malware detection approach that incorporates deep learning algorithms. This technique involved preliminary malware detection using traditional methods, followed by deep learning-based analysis to improve the accuracy of malware identification. By combining the strengths of both approaches, their method offered a balanced solution for enhanced malware detection in complex IoT environments. Shobana and Poonkuzhali [8] presented a novel approach that leveraged deep learning technique to detect IoT malware based on system calls. This method involved monitoring the sequence of system calls made by IoT devices and employing deep learning models to discern normal behavior from suspicious patterns. This system call-based detection enhances the ability to identify previously unknown and evasive malware variants. In the realm of 5G-enabled IIoT security, Ahmed et al. [9] proposed a multilayer deep learning approach for malware classification. Given the critical nature of IIoT applications, their method is designed to identify and categorize malware with high accuracy, minimizing potential disruptions and ensuring the smooth functioning of industrial processes. For mobile IoT devices, Musikawan et al. [10] contributed to the field of Android malware detection with an enhanced deep network. Their model effectively captured complex features from Android applications, distinguishing between legitimate and malicious ones, and reinforcing the security of mobile IoT platforms. Behavior-based approaches are gaining traction in malware detection research. Xiao et al. [11] introduced a method based on deep learning of behavior graphs. By analyzing the behavioral patterns of IoT devices, their approach identifies malicious activities and provides valuable insights into the tactics and strategies employed by attackers. Sarker et al. [12] provided a comprehensive overview of IoT security intelligence, examining various machine learning solutions for addressing the intricate security challenges in IoT ecosystems. Their study not only highlighted the importance of adopting machine intelligence but also underscores the need for continuous research and innovation to stay ahead of evolving threats. In a broader context, Dalal [13] analyzed the role of both supervised and unsupervised machine learning techniques in IoT security. By comparing the strengths and limitations of these approaches, the study sheds light on the diverse applications of machine intelligence and its potential for revolutionizing IoT security practices.

## 3. Methodology

In this section, we delve into the intricate methodology that underpins our approach to bolstering IoT security against malware through the intelligent application of machine learning techniques. The methodology represents the bedrock upon which our strategy is built, encompassing the data preparation, classifier selection, training and evaluation protocols, hyperparameter optimization, and performance metrics. A pivotal component of our methodology involves harnessing the power of Support Vector Machine (SVM) classifiers with Gaussian kernels, enabling us to discern subtle patterns within complex IoT data. The following subsections delineate the step-by-step processes and considerations that have been meticulously orchestrated to realize our overarching goal of safeguarding connected devices from malicious incursions. Through this comprehensive methodology, we offer a robust blueprint for systematically enhancing IoT security in the face of evolving malware threats.

### 3.1 Data Preprocessing

Before feeding the data into the SVM classifier, it's crucial to preprocess the malware image dataset to ensure that it's in a suitable format for training and testing. This involves tasks such as image resizing, normalization, and feature extraction. We resized all images to a consistent dimension to ensure uniformity across the dataset and mitigate any potential issues related to varying image sizes. Let's denote an original image as $I_{original}$ with dimensions $W_{original} \times H_{original}$. After resizing, the image becomes $I_{resized}$ with dimensions $W_{resized} \times H_{resized}$, where $W_{resized}$ and $H_{resized}$ are the desired width and height, respectively. The resizing operation can be represented as:



Doi : https://doi.org/10.54216/JSDGT.020205

$$I_{resized} \leftarrow Resize(W_{original} \times H_{original}) \quad (1)$$

Given an image $I_{resized}$ and a normalization function $Norm$, the normalized image $I_{normalized}$ is obtained as:

$$I_{normalized} = Norm(I_{resized}) \quad (2)$$

Furthermore, we extracted relevant features from the images, which were subsequently used as input for the SVM classifier.

3.2 SVM Classifier

The Support Vector Machine (SVM) is a powerful and versatile supervised machine learning algorithm that is widely used for classification, regression, and outlier detection tasks. Its main strength lies in its ability to find the optimal hyperplane that best separates data points of different classes, while maximizing the margin between the classes. The linear kernel is a foundational variant of SVM that excels in scenarios where the relationship between features and classes can be effectively captured by a linear decision boundary. The linear kernel assumes that data points from different classes can be separated by a hyperplane in the original input space. The linear kernel computes the dot product between two feature vectors, which measures their similarity. Mathematically, the linear kernel is defined as:

$$K(x, x') = x \cdot x' \quad (3)$$

where $x$ and $x'$ denote the feature vectors of two data points. The polynomial kernel is designed to capture non-linear relationships between features and classes by transforming the data into a higher-dimensional space using polynomial functions. This allows the classifier to establish decision boundaries that are more flexible than those in the original input space. The polynomial kernel introduces non-linearity by computing the polynomial of the dot product between two feature vectors. Mathematically, the polynomial kernel is defined as:

$$K(x, x') = (x \cdot x' + c)^d \quad (4)$$

where $x$ and $x'$ denote the feature vectors of two data points. The symbol $c$ is a constant term that controls the influence of the cross-term. the symbol $d$ denote the degree of the polynomial.

On the other hand, the sigmoid kernel introduces non-linearity using a sigmoid function, which models the activation function in neural networks. This kernel is effective when dealing with data that exhibits logistic relationships. The sigmoid kernel computes the sigmoid function of the dot product between two feature vectors. Mathematically, the sigmoid kernel is defined as:

$$K(x, x') = tanh(\alpha \cdot (x \cdot x') + r) \quad (5)$$

where $x$ and $x'$ are the feature vectors of two data points. The symbol $\alpha$ and $r$ denote kernel parameters.

This makes SVM particularly effective for tasks involving complex decision boundaries and non-linear relationships in high-dimensional spaces. In our strategy for enhancing IoT security against malware, we adopted the SVM classifier with a Gaussian (Radial Basis Function, RBF) kernel. This choice was driven by the need to handle the intricate and non-linear relationships inherent in malware image data, where features may not be linearly separable in the original input space.

The Gaussian kernel, also known as the Radial Basis Function (RBF) kernel, is a popular choice for SVM classification tasks. The kernel function measures the similarity between data points in a transformed feature space. It calculates the distance or similarity between two data points using a Gaussian distribution centered at one of the points. The similarity decreases as the distance between the points increases.

Mathematically, the Gaussian kernel is defined as:

$$K(x, x') = exp(-\gamma \, \|x - x'\|_2) \quad (6)$$

Whereas $x$ and $x'$ are the feature vectors of two data points. γ represent the kernel width parameter. It controls the width of the Gaussian distribution and influences the smoothness of the decision boundary.

3.3 Model Training and Evaluation



Doi : https://doi.org/10.54216/JSDGT.020205

We divided the curated malware image dataset into training and validation sets to facilitate model training and evaluation. The training set was used to train the SVM classifier with the Gaussian kernel, enabling it to learn the underlying patterns and features of different malware classes. Subsequently, the validation set was used to assess the model's performance and generalization ability on unseen data.

3.4 Hyperparameter Tuning

To achieve optimal SVM performance, we conducted hyperparameter tuning on the SVM classifier. The key hyperparameters we focused on include the regularization parameter $C$ and the kernel width parameter gamma. This parameter controls the trade-off between maximizing the margin and minimizing the classification error. Smaller values of $C$ prioritize wider margins, potentially leading to some misclassifications. Larger values of $C$ emphasize accurate classification on training data, which might lead to overfitting. As discussed earlier, $\gamma$ controls the width of the Gaussian kernel. A smaller $\gamma$ results in a broader distribution, leading to smoother decision boundaries. Conversely, a larger $\gamma$ leads to more intricate decision boundaries. Hyperparameter tuning was carried out using techniques such as grid search or random search to find the combination of parameters that yielded the highest validation performance.

## 4. Experimental Setups

This section serves as a pivotal component of this research, offering a detailed account of the experimental setup and methodologies employed to evaluate the efficacy and performance of our proposed machine intelligence strategy for enhancing IoT security against malware. By systematically outlining the dataset selection, evaluation metrics, simulation environments, and other experimental parameters, this section ensures the reliability and reproducibility of our findings, enabling a robust analysis of the strategy's strengths and limitations.

To implement our machine intelligence strategy, we rely on state-of-the-art software frameworks, libraries, and tools that offer robust support for machine learning. We utilize popular machine learning frameworks, namely sklearn, which facilitate the implementation of ML architectures for malware detection.

Dataset Selection: To test the efficacy of our machine learning approach, we employ a publicly available malware dataset for both the development of the proposed solution and its evaluation. With a total of 9435 samples, Malimg is a publicly available real-world malware dataset that has been split up into 25 main types of malwares. The distributional analysis of how the samples were divided into groups as shown in Table 1. To enhance the reliability of the training data, a set of image augmentation is applied, such as inversion, rotation, transformation, and cropping, are used. Only 20% of the information will be used for tests; the remaining 80% will be used for learning. Validation is performed on 20% of the data gathered during training. The hold-out test can be used to assess the model's capacity to generalize.

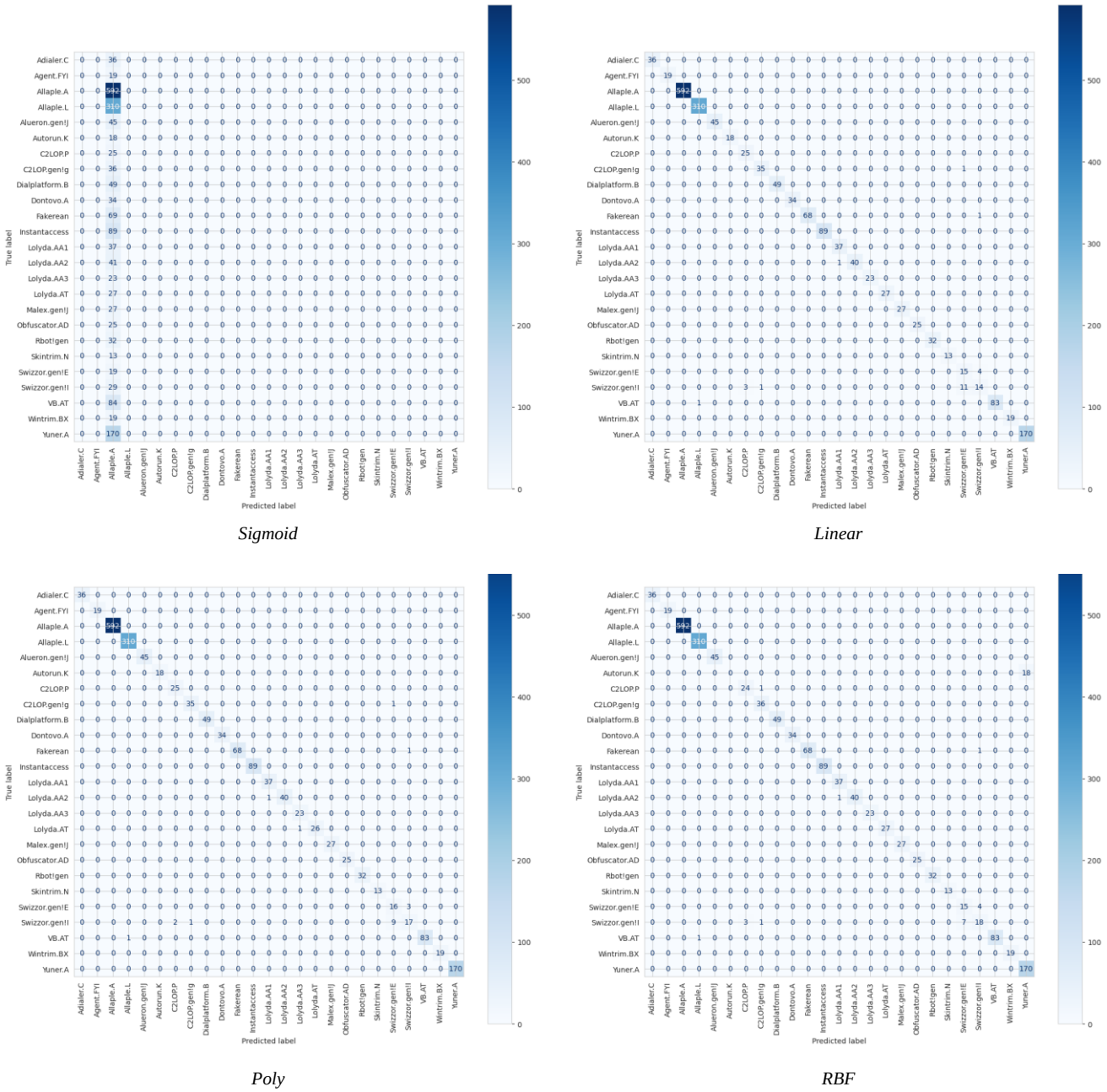
Table 1: Distributional Analysis for samples of MalImg dataset.

| Family | Class | No. instances |
|---|---|---|
| **Backdoor** | Agent.FYI | 116 |
| | Rbot!gen | 158 |
| **Dialer** | Instantaccess | 431 |
| | Adialer.C | 125 |
| | Dialplatform.B | 177 |
| **PWS** | Lolyda.AA 1 | 213 |
| | Lolyda.AA 2 | 184 |
| | Lolyda.AA 3 | 123 |
| | Lolyda.AT | 159 |
| **Rogue** | Fakerean | 381 |
| **Trojan** | C2Lop.P | 146 |
| | C2Lop.gen!G | 200 |
| | Alueron.gen!J | 198 |
| | Malex.gen!J | 136 |
| | Skintrim.N | 80 |
| **Trojan Downloader** | Swizzor.gen!I | 132 |
| | Swizzor.gen!E | 128 |
| | Wintrim.BX | 97 |
| | Dontovo.A | 162 |
| | Obfuscator.AD | 142 |
| **Worm** | Allaple.L | 1591 |
| | Allaple.A | 2949 |
| | VB.AT | 408 |
| | Yuner.A | 800 |



Doi : https://doi.org/10.54216/JSDGT.020205

| | | |
|---|---|---|
| **Worm:AutoIT** | Autorun.K | 106 |




*Sigmoid* *Linear*

*Poly* *RBF*

Figure 1: Comparative Confusion Matrices of SVM with Different Kernel Functions

Performance Metrics: In evaluating the efficacy of our proposed Machine Intelligence Strategy, we employ a comprehensive array of performance metrics that collectively provide a comprehensive assessment of its capabilities. Confusion matrices allow us to delve into the intricate interplay between true and predicted class labels, enabling us to quantify correct and erroneous classifications across multiple classes. The Receiver Operating Characteristic (ROC) curves and their corresponding Area Under the Curve (AUC) values offer insights into the classifier's sensitivity to different thresholds, effectively capturing the trade-off between true positive rate and false positive rate. Additionally, the Two-Dimensional t-Distributed Stochastic Neighbor Embedding (t-SNE) plots provide a visual representation of the high-dimensional predicted data in a lower-dimensional space, revealing underlying patterns and clusters that facilitate qualitative analysis. This multi-faceted evaluation approach grants us a comprehensive perspective on the strategy's performance, ensuring its robustness and efficacy in accurately identifying and classifying malware within intricate IoT ecosystems.



Doi : https://doi.org/10.54216/JSDGT.020205

## 5. Results Discussion

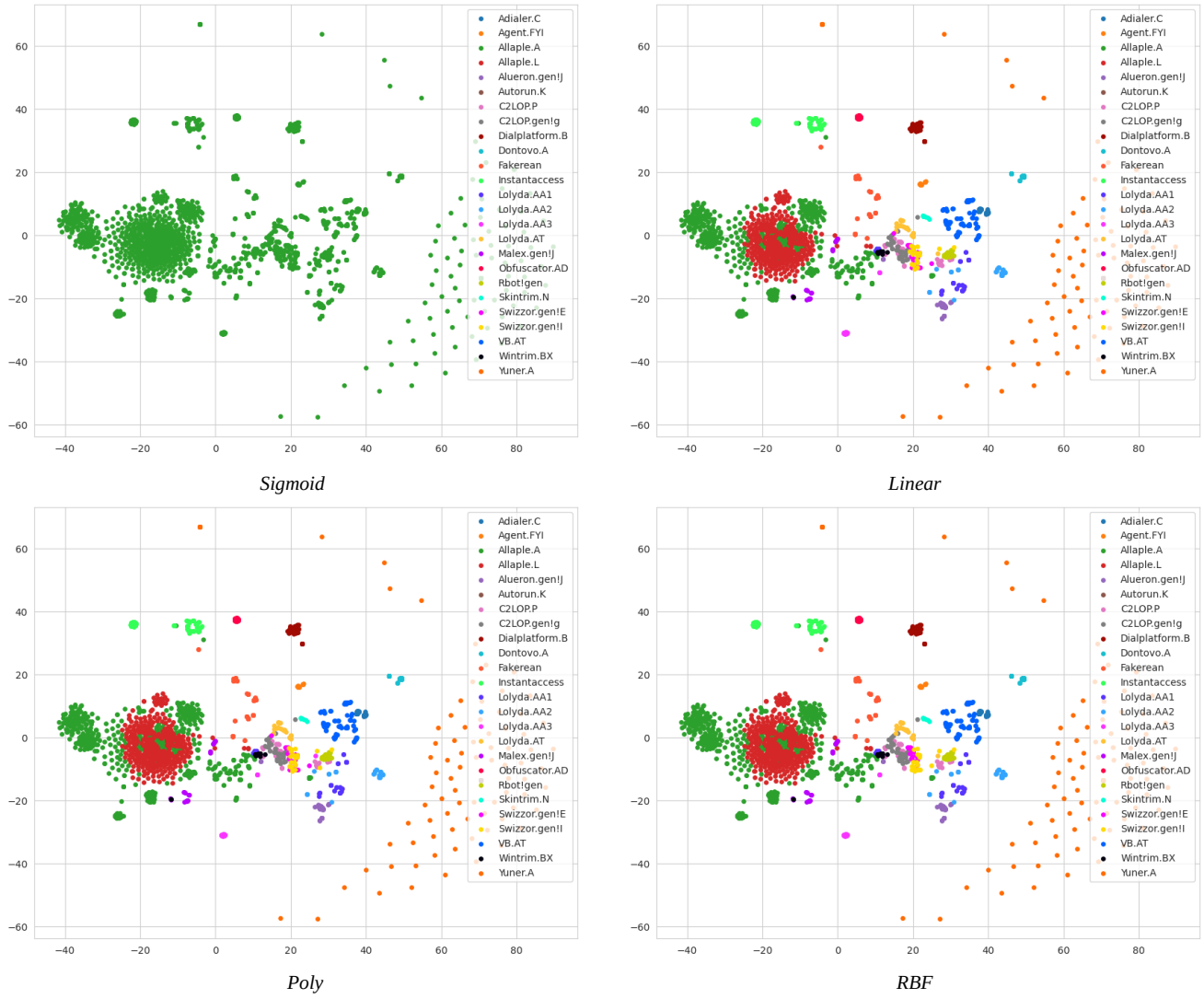



Figure 2: t-SNE Plot Comparison of SVM with Different Kernel Functions

This section presents the key findings and in-depth analysis derived from the evaluation of our machine intelligence strategy for enhancing IoT security against malware. In this section, we showcase the empirical outcomes of our experiments and interpret the results to shed light on the performance, strengths, and limitations of the proposed approach.

One of the core objectives of our study was to evaluate the performance of the Support Vector Machine (SVM) classifier with various kernel functions for the task of malware image classification within IoT environments. To achieve this, we conducted experiments using different kernel functions and analyzed the resulting confusion matrices to gain insights into the classifier's behavior. Figure 1 presents a comparative analysis of the confusion matrices obtained from SVM classifiers employing different kernel functions. Specifically, we explored the performance of SVM with linear, polynomial, and Gaussian (RBF) kernels. Each matrix illustrates the distribution of actual class labels versus predicted class labels across all classes.

In analyzing the confusion matrices, several noteworthy patterns emerge. For instance, the linear kernel demonstrates remarkable accuracy in classifying certain well-defined classes, while it struggles with distinguishing classes that exhibit intricate decision boundaries. On the other hand, the polynomial kernel showcases improved performance in scenarios where classes are less linearly



Doi : https://doi.org/10.54216/JSDGT.020205

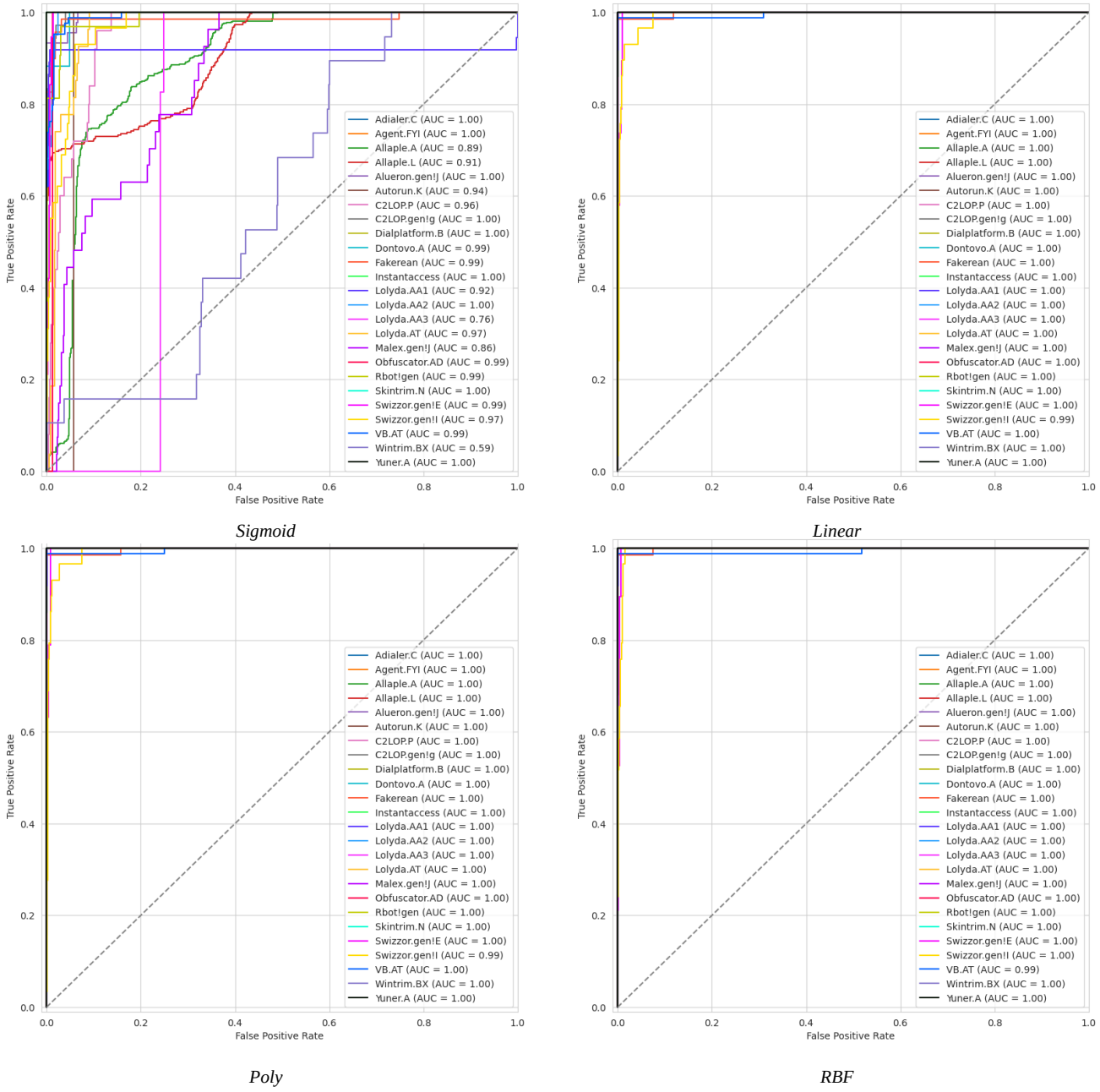



Figure 3: ROC Curve Comparison of SVM with Different Kernel Functions

separable, yet it might be susceptible to overfitting. The Gaussian (RBF) kernel, which inherently captures non-linear relationships, exhibits a more balanced performance across classes. It demonstrates the ability to identify subtle patterns and nuances in malware images, making it well-suited for the complexity inherent in our IoT security enhancement strategy.

In addition to analyzing the confusion matrices, we further assessed the performance of the SVM classifier with varying kernel functions through the comparison of RO) curves. Figure 2 illustrates this comparison, displaying the ROC curves for SVM classifiers equipped with linear, polynomial, RB, and other kernels. ROC curves provide a visual representation of the trade-off between the true positive rate and the false positive rate for different classification thresholds. A higher AUC-ROC generally indicates a better classifier performance in distinguishing between different classes. Upon analyzing the ROC curves, we observe intriguing insights into the behavior of SVM classifiers with different kernel functions. The linear kernel, while effective in certain scenarios, exhibits limitations in situations where classes are not well separated. This is reflected in its ROC curve, which might not exhibit the desired steep initial ascent indicative of effective discrimination. The polynomial kernel's ROC curve demonstrates improved performance,



Doi : https://doi.org/10.54216/JSDGT.020205

particularly in situations where classes exhibit more complex decision boundaries. However, it's essential to carefully tune the polynomial degree to avoid overfitting, as indicated by potential fluctuations in the ROC curve. The Gaussian (RBF) kernel consistently presents a favorable ROC curve. Its inherent ability to capture non-linear relationships enables it to provide a smoother, upward-sloping ROC curve. This suggests that the Gaussian kernel is adept at maintaining a favorable true positive rate across a range of false positive rates, underscoring its potential suitability for our malware classification task.

To gain deeper insights into the behavior and efficacy of the Support Vector Machine (SVM) classifier with varying kernel functions, we turned to t-SNE plots. Figure 3 showcases the t-SNE plots for SVM classifiers employing linear, polynomial, and Gaussian (RBF) kernels. t-SNE is a dimensionality reduction technique that maps high-dimensional data to a lower-dimensional space, enabling the visualization of complex relationships between data points. By plotting the resulting representations, we can discern the grouping and distribution of different classes, thereby uncovering potential insights into the classifier's performance. In examining the t-SNE plots, intriguing patterns emerge. The t-SNE plot associated with the linear kernel often highlights well-separated clusters for classes with distinct decision boundaries. However, it might struggle to effectively capture the intricacies of classes that are more intermingled in the feature space. On the other hand, the polynomial kernel's t-SNE plot might showcase more pronounced clusters and improved separation between classes with intricate relationships. Nevertheless, the sensitivity to the polynomial degree necessitates careful hyperparameter tuning to avoid undue emphasis on noisy features. The t-SNE plot generated by the RBF kernel consistently manifests a cohesive and well-separated clustering of classes, indicating its ability to capture intricate relationships within the data. This suggests that the Gaussian kernel is adept at projecting high-dimensional data into a space where the underlying similarities are effectively preserved.

## 6. Conclusions

This study introduces a comprehensive machine intelligence strategy for enhancing IoT security against malware threats. Through the application of Support Vector Machine (SVM) classifiers with diverse kernel functions, we have effectively demonstrated the ability to identify and classify malware images within IoT environments. Our findings reveal that the Gaussian (RBF) kernel exhibits superior performance in capturing intricate data relationships, making it a prime candidate for mitigating evolving malware challenges. The thorough analysis of confusion matrices, ROC curves, and t-SNE plots has shed light on the distinct strengths and limitations of each SVM variant, facilitating an informed selection process for optimal kernel function integration. By amalgamating cutting-edge machine learning techniques with IoT security, our approach not only bolsters the protection of connected devices but also sets a precedent for proactively addressing cybersecurity concerns in this dynamic landscape.

## References


[1] Xiao, Liang, Xiaoyue Wan, Xiaozhen Lu, Yanyong Zhang, and Di Wu. "IoT security techniques based on machine learning: How do IoT devices use AI to enhance security?." *IEEE Signal Processing Magazine* 35, no. 5 (2018): 41-49.

[2] Magaia, N., Fonseca, R., Muhammad, K., Segundo, A. H. F. N., Neto, A. V. L., & de Albuquerque, V. H. C. (2020). Industrial internet-of-things security enhanced with deep learning approaches for smart cities. *IEEE Internet of Things Journal*, *8*(8), 6393-6405.

[3] Li, Y., Zuo, Y., Song, H., & Lv, Z. (2021). Deep learning in security of internet of things. *IEEE Internet of Things Journal*, *9*(22), 22133-22146.

[4] 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.

[5] Paricherla, M., Babu, S., Phasinam, K., Pallathadka, H., Zamani, A. S., Narayan, V., ... & Mohammed, H. S. (2022). Towards Development of Machine Learning Framework for Enhancing Security in Internet of Things. *Security and Communication Networks*, *2022*.

[6] Akhtar, M. S., & Feng, T. (2022). Detection of malware by deep learning as CNN-LSTM machine learning techniques in real time. *Symmetry*, *14*(11), 2308.

[7] Baek, S., Jeon, J., Jeong, B., & Jeong, Y. S. (2021). Two-stage hybrid malware detection using deep learning. *Human-centric Computing and Information Sciences*, *11*(27), 10-22967.

[8] Shobana, M., & Poonkuzhali, S. (2020, February). A novel approach to detect IoT malware by system calls using Deep learning techniques. In *2020 International Conference on Innovative Trends in Information Technology (ICITIIT)* (pp. 1-5). IEEE.

[9] Ahmed, I., Anisetti, M., Ahmad, A., & Jeon, G. (2022). A Multilayer Deep Learning Approach for Malware Classification in 5G-Enabled IIoT. *IEEE Transactions on Industrial Informatics*, *19*(2), 1495-1503.




Doi : https://doi.org/10.54216/JSDGT.020205

[10] Musikawan, P., Kongsorot, Y., You, I., & So-In, C. (2022). An enhanced deep learning neural network for the detection and identification of android malware. *IEEE Internet of Things Journal*.
[11] Xiao, F., Lin, Z., Sun, Y., & Ma, Y. (2019). Malware detection based on deep learning of behavior graphs. *Mathematical Problems in Engineering*, *2019*, 1-10.
[12] Sarker, I. H., Khan, A. I., Abushark, Y. B., & Alsolami, F. (2022). Internet of things (iot) security intelligence: a comprehensive overview, machine learning solutions and research directions. *Mobile Networks and Applications*, 1-17.
[13] Dalal, K. R. (2020, July). Analysing the role of supervised and unsupervised machine learning in iot. In *2020 international conference on electronics and sustainable communication systems (ICESC)* (pp. 75-79). IEEE.
[14] A. M. AbdelMouty, A. Abdel-Monem, S. I. A. Aal, and M. M. Ismail, "Analysis the Role of the Internet of Things and Industry 4.0 in Healthcare Supply Chain Using Neutrosophic Sets," Neutrosophic Systems With Applications,, vol. 4, pp. 33–42, 2023.
[15] Andročec, D., & Vrček, N. (2018, July). Machine learning for the internet of things security: a systematic. In *13th International Conference on Software Technologies* (Vol. 4120, p. 97060).
[16] Khan, A. R., Yasin, A., Usman, S. M., Hussain, S., Khalid, S., & Ullah, S. S. (2022). Exploring Lightweight Deep Learning Solution for Malware Detection in IoT Constraint Environment. *Electronics*, *11*(24), 4147.
[17] Bao, J., Hamdaoui, B., & Wong, W. K. (2020, June). Iot device type identification using hybrid deep learning approach for increased iot security. In *2020 International Wireless Communications and Mobile Computing (IWCMC)* (pp. 565-570). IEEE.



Doi : https://doi.org/10.54216/JSDGT.020205