algorithms

MDPI

Article

Impacting Robustness in Deep Learning-Based NIDS through Poisoning Attacks

Shahad Alahmed [1], Qutaiba Alasad [2], Jiann-Shiun Yuan [3] and Mohammed Alawad [4,*]

[1] Department of Computer Science, Tikrit University, Al Qadisiyah P.O. Box 42, Iraq; shahad.m.mustafa@tu.edu.iq
[2] Department of Petroleum Processing Engineering, Tikrit University, Al Qadisiyah P.O. Box 42, Iraq; qutaibaeng@tu.edu.iq
[3] Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL 32816, USA; jiann-shiun.yuan@ucf.edu
[4] Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI 48202, USA
* Correspondence: alawad@wayne.edu

Abstract: The rapid expansion and pervasive reach of the internet in recent years have raised concerns about evolving and adaptable online threats, particularly with the extensive integration of Machine Learning (ML) systems into our daily routines. These systems are increasingly becoming targets of malicious attacks that seek to distort their functionality through the concept of poisoning. Such attacks aim to warp the intended operations of these services, deviating them from their true purpose. Poisoning renders systems susceptible to unauthorized access, enabling illicit users to masquerade as legitimate ones, compromising the integrity of smart technology-based systems like Network Intrusion Detection Systems (NIDSs). Therefore, it is necessary to continue working on studying the resilience of deep learning network systems while there are poisoning attacks, specifically interfering with the integrity of data conveyed over networks. This paper explores the resilience of deep learning (DL)—based NIDSs against untethered white-box attacks. More specifically, it introduces a designed poisoning attack technique geared especially for deep learning by adding various amounts of altered instances into training datasets at diverse rates and then investigating the attack's influence on model performance. We observe that increasing injection rates (from 1% to 50%) and random amplified distribution have slightly affected the overall performance of the system, which is represented by accuracy (0.93) at the end of the experiments. However, the rest of the results related to the other measures, such as PPV (0.082), FPR (0.29), and MSE (0.67), indicate that the data manipulation poisoning attacks impact the deep learning model. These findings shed light on the vulnerability of DL-based NIDS under poisoning attacks, emphasizing the significance of securing such systems against these sophisticated threats, for which defense techniques should be considered. Our analysis, supported by experimental results, shows that the generated poisoned data have significantly impacted the model performance and are hard to be detected.

Keywords: deep learning; network intrusion detection system (NIDS); deep fool; poisoning attacks; pearson correlation method; CICIDS2019

Citation: Alahmed, S.; Alasad, Q.; Yuan, J.-S.; Alawad, M. Impacting Robustness in Deep Learning-Based NIDS through Poisoning Attacks. *Algorithms* **2024**, *17*, 155. https://doi.org/10.3390/a17040155

Academic Editors: Frank Werner, José Simão, Nuno Datia and Matilde Pato

Received: 1 March 2024
Revised: 2 April 2024
Accepted: 6 April 2024
Published: 11 April 2024

1. Introduction

Cybercriminal activities continue to pose significant threats to personal data, particularly amid an era of heightened technological advancements [1,2]. Each attacker, driven by the nature of target data, employs a distinct set of skills [3], often targeting valuable information, especially private data encompassing economic, military, and celebrity—related content [4]. These attackers carefully lead their attacks through using different techniques, such as data interruption, interception, modification, and manufacturing, to produce harm in the targets [5–7]. They orchestrate different cybercriminal activities using their effective ways to provide malicious threats, namely data poisoning [3].

To prevent such attacks, intrusion detection systems (IDSs) have been effectively introduced to play a crucial role in checking the data streams and give alerts to decision-makers once certain criteria are achieved [8–10]. More specifically, IDSs transfer into Network Intrusion Detection Systems (NIDS) at the network flow stage, which necessitates training data to identify network flow records [11]. Traditional retraining of NIDS is important to improve the flow behavior [12].

Deep Learning (DL) is considered to be one of the best techniques in many domains, such as pattern recognition, spanning language, images, speech, and video content [13–16]. Its superiority over other techniques stems from its improved computational capabilities, cost-effectiveness of computing equipment, and breakthroughs in Machine Learning (ML) [15]. However, poisoning cyberattacks targeting training datasets have emerged as a prominent concern among machine learning practitioners [12]. These attacks aim to corrupt training data intentionally, leading to poor performance of ML systems [12]. Yet, creating feasible poisoning attacks for cyber activities remains challenging, with limited understanding of mitigation strategies [17].

Motivated by the discrepancy between proposed adversarial settings in NIDS studies and their actual feasibility [18], our research aims to address this gap. Many studies often assume that threat models are without sufficient consideration of their practicality [18–20]. Moreover, research on poisoning attempts in the cyber realm has been confined to specific applications, such as computer vision, tabular, and text data [21–23]. However, reliance on outdated datasets like NSL-KDD (35%) [24], fraught with flaws and unreflecting modern networks, underscores the necessity for updated datasets, such as CICIDS2017, CSECICIDS2018, and LITENT2020 [25–28].

Utilizing these updated datasets allows for accurate conclusions regarding modern technological challenges in networks [27,28]. Label manipulation attack techniques in training data, though limited in attacking power, present an evident drawback as it is incapable of achieving sophisticated adversarial goals [29,30]. Current hostile machine learning studies predominantly focus on evasion attacks during inference [31]. Recent surveys highlight poisoning attacks as a primary concern among implementing ML organizations, necessitating a shift towards addressing poisoning attack during the training [32]. This shift emphasizes the need for further research in this critical area.

While poisoning assault strategies have been extensively explored within conventional ML techniques, only few studies have been specifically tailored for DL [33,34]. To address these issues, we introduce a poisoning attack approach aimed at evaluating the performance of DL model. Our experimentation involves inserting varying quantities of harmful samples into the model. These poisoned samples, generated via the DeepFool method as an untargeted attack [35], ensure minimal modifications to repeatedly deceive the model's classification [36]. This methodology strategically avoids large modifications that might drastically deviate samples from benign elements, resulting in diminished performance and rapid detection [37].

Subsequently, these poisoned instances are introduced into the original training dataset with diverse poisoning rates ranging from 1% to 10%. The injection involves altering the distance among the injected poisoned samples, randomly placing them within the original training locations. Given the substantial data requirements for DL techniques and hyperparameter adjustment, our research leverages extensive datasets provided by the Canadian Institute of Cybersecurity (CIC), specifically the Communications Security and Establishment dataset (CIC2019 or CICIDS2019) [38]. This choice showcases the efficacy of our proposed assault approach in a real-world context.

Feature selection crucially relied on Pearson's Correlation technique to discern relationships between features, acknowledging the stability of associations in larger datasets compared to smaller ones [39]. Leveraging a high-dimensional dataset encompassing recent network attacks, we rigorously assessed the robustness of the system and presented our findings based on multiple metrics, such as accuracy, True Positive Rate (TPR), and True Negative Rate (TNR). This comprehensive evaluation offers a comprehensive insight into the efficacy of the proposed approach.

This paper contributes significantly by bridging critical gaps in the field of network security. Specifically, this work presents a strong poisoning-based attacks method designed explicitly for DL to address a prominent deficiency in most recent literature that focuses on orthodox ML techniques. Our research provides an evaluation on the resilience and susceptibility of DNNs towards poisoning-based attacks via meticulous experiments leveraging the novel DeepFool method to create poisoned data. Furthermore, the study presents a comprehensive analysis of the attack's influence on the performance of the system by injecting various amounts of such manipulated samples into the training datasets. Employing inclusive dataset provided by the Canadian Institute of Cybersecurity (CIC), notably as CIC2019 (CICIDS2019), offers real-world pertinence and validity to the findings of this study. Applying Pearson's Correlation method to select features in datasets with high-dimensions further enhances the accuracy and depth of this investigation. Ultimately, the paper's contribution lies in its meticulous exploration of data poisoning-based attacks on DNNs, shedding light on their implications for network security and offering insights into fortifying systems against such threats.

The article is structured into distinct sections, each dedicated for specific facets of the research. Section 2 delves into a review of related work. Following this, Section 3 elucidates the fundamental concepts introduced in this paper. Section 4 intricately details the experimental setup. The methodological implementation is thoroughly outlined in Section 5, providing insight into the research approach. Section 6 examines and presents the findings derived from the study's experimental cases. Finally, Section 7 offers insightful concluding remarks and outlines potential avenues for future research endeavors.

2. Related Work

2.1. DL-Based NIDS

The rapid evolution of network services has spurred the development of IDSs bolstered by DL techniques. Xu et al. devised a multi-level deep-learning-based IDS that showcased exceptional performance metrics when tested on benchmark datasets like KDD 99 and NSL-KDD. Their technique produces high identification rates, 99.42% and 99.31%, accompanied by low False Positive Ratios (FPR) of 0.05% and 0.84%, respectively [40]. In [41], Peng, Kong et al. proposed a reliable Neural Network (NN) designed to retrieve features from a traffic network to offer higher accuracy compared to conventional ML systems based on the KDD99 dataset evaluation. Fernandez, Xu, and Aldallal research improved the effectiveness of DL-based IDSs in real-world situations. A DL is proposed to identify abnormal transactions in many datasets to outperform different ML techniques with a high True Positive Rate (TPR) of 99.93% in certain examples [42]. An IDS framework is introduced to detect DDoS attacks to ensure high resilience and speed against adversarial attacks, where the framework achieves a remarkable recall rate of 98.2% [43]. These works have effectively underlined the increasing dependence on DL models to fortify IDSs against strong and resilient cyber-attacks.

2.2. Poisoning Attacks

Extensive research about poisoning attacks has been conducted recently due to the escalating challenges of ML security vulnerabilities. Many approaches have been presented by researchers to compromise ML systems to highlight their susceptibility towards adversarial manipulation. In [44], a gradient ascent technique with the MNIST dataset has been used to construct optimal threats through employing poisoning attacks to target Support Vector Machines (SVM). In contrast, the realm of targeted clean-label-based attackers has been presented to introduce slight changes in labeled samples to perturb certain examples in the training set, even though such kind of attack could be detectable in the created disruptions [45].

Furthermore, many studies have concentrated on data poisoning-based attacks in the feature selection setting and interpretability of the system model. Data poisoning-based attacks have been suggested against two embedded feature selection approaches, namely: ridge regression and LASSO. A sub-gradient ascent method is leveraged to compromise system models. The research shows the effectiveness of the poisoning attack on the targeted

system [46]. The investigation of data poisoning-based attacks across many different machine learning models and methodologies of feature selection has ensured the necessity to have strong defenses to counteract such attacks. These explorations assure the immediate urgency to protect ML systems against poisoning-based threats, a pressing challenge in contemporary cybersecurity fields.

In contrast to prior studies, our proposal addresses and evaluates the robustness of DL-based NIDS against poisoning-based attacks. Data-based poisoning attack has been launched to exploit the DL model during the training phase. This threat occurs when the training data of the DL model is changed, affecting the ML decision-making findings. This type of attack intends to deceive the system into drawing incorrect conclusions although deep learning models are usually considered as black boxes. Therefore, the attack compromises the integrity of the model. The resilience of DL-based NIDS has been evaluated via injecting manipulated samples, which are generated using the DeepFool method, into training datasets. Our research assesses different quantities of poisoned instances, focusing on the effectiveness of such assaults on the DL-based NIDS performance. It also aims to provide insights to boost DL-based NIDS against contemporary network security attacks to address the vital aspect of system robustness in the face of poisoning-based threats leveraging the most recent dataset CIC2019 (CICIDS2019) produced by the Canadian Institute of Cybersecurity. The reliability and depth of our proposal has been further improved by incorporating the application of feature selection approach, which is Pearson's Correlation in high-dimensional datasets. A summarization of prior works discussed above are illustrated in Table 1.

Table 1. Summary of related work.

Reference	Dataset	Methodology	Conclusion
[40]	KDD99 NSL-KDD	Developed a Recurrent Neural Network (RNN) algorithm and Gated Recurrent Units (GRU) with automatic feature-selection.	Comparative experiments showed that the GRU is more suitable as a memory unit for intrusion systems than LSTM.
[41]	UNSW-NB15	Focused on the utility of DL frameworks for NIDS systems that scan traffic across networks to detect and record a violation based on the intrusion's behavioral patterns discovered in the dataset.	The suggested technique achieved a total accuracy of (95.4% and 95.6%) for the prepartitioned and multiple-class categorization systems.
[42]	ISCXIDS 2012	Proposed Deep Neural Network (DNN) that trains an NIDS using supervised training and uses an autoencoder to identify and categorize attack traffic by unsupervised learning.	DNN works satisfactorily in supervised IDSs, particularly when only the first 3 octets of IP addresses are processed. Autoencoders can additionally recognize anomalies when trained on benign traffic.
[43]	CICIDS2018	Developed a DL technique for detecting cloud computing assaults, utilizing Pearson's Correlation method.	The suggested system has an extremely low FPR of 0.003%. This minimizes the need for human analysts to manage many alerts.
[44]	MNIST	Suggested an attack to employ a gradient ascent technique, in which the gradient is calculated using characteristics of the SVM's best solution to raise the classifier's test errors.	The findings show that the gradient ascent approach accurately determines the local maxima of the non-convex validation error surface.
[45]	ImageNet	Focused on targeted poisoning assaults that change the classification in modern-DNN of unaltered test images, compromising system integrity.	This approach enables highly successful data poisoning assaults against completely retrained models.
[46]	PDF file.	A methodology has been suggested to better analyze and define assaults on feature extraction strategies, which are: LASSO, ridge regression, and the elastic net.	Malware detection outcomes indicate that feature selection approaches can be greatly compromised during attacks, focusing on the necessity for particular countermeasures.

3. Background

3.1. Vulnerabilities in Deep Learning Models

DL models represent a cornerstone in modern ML, harnessing complex neural structures akin to the human brain to excel in learning intricate patterns from expansive datasets [37]. Their hierarchical architecture, often comprising multiple layers including input, hidden, and output layers, enables sophisticated learning and abstraction of features within data [47,48]. The iterative training process refines DNNs' performance by adjusting internal parameters, continuously improving outcomes [49,50].

Despite their remarkable capabilities, DL models exhibit vulnerabilities to adversarial attacks and data poisoning. Adversarial example-based attacks contain malicious input instances to fool the ML models, leading to an unsuccessful classification and incorrect results [51]. From another side, inserting poisoning sample-based threats in the training datasets creates malicious samples to reduce the performance of the ML model [52]. Such attacks have challenged the reliability of DNNs model due to exposing the vulnerability of the ML model. Adversarial example-based threats can lead to incorrect predictions or misclassifications, which highlight the need to propose a technique that is capable of protecting DNNs against such threats [53]. Revealing the vulnerability of the ML models has become a concern, and it is necessary to create a strong model against these threats in many applications. This kind of high level protection renders the system strong against possible threats and guarantee the best performance of the model.

In this article, DNNs against malicious data poisoning-based threats have been effectively evaluated via inserting different rates of poisoning sample-based threats, which are created leveraging DeepFool approach, into the training datasets.

3.2. DeepFool Method

The DeepFool technique, found by Moosavi-Dezfooli, is considered as an untargetable white-box threat among adversarial data-based threats [54]. This technique operates via estimating the distance between a given classifier and input data and then applying an orthogonal projection towards the closest boundary. During this operation, a computational approach is used to find the least required perturbations that are necessary to generate adversarial examples that can affect the classifiers [55]. DeepFool repeats this procedure iteratively until the input is successfully changed to lead to a misclassification in the network system. The resilience metric connected to such threat is based on measuring the lowest required modification to fool the network once it is provided with a certain input data [56].

3.3. Pearson's Correlation Method

Although feature selection is an important aspect in the ML field, the abundancy of features in a dataset can increase model complexity and extend training durations. To enhance efficiency and ensure effective attack identification, it has become imperative to transform collected data into a more streamlined dimension [57]. Various feature selection strategies exist in ML, including but not limited to backward feature selection, chi-square, and recursive feature removal, and each considers specific dataset characteristics, dimensionality, and correlation [58]. Our chosen approach for feature extraction revolves around Pearson correlation coefficient method. This method evaluates variable correlation by illustrating the linear relationship between dataset features [39]. Ranging between -1 and $+1$, the correlation coefficient denotes the strength and nature of the relationship between variables. A value of '0' means that there is no relationship, while '1' refers to a strong positive correlation, and '-1' indicates a strong negative correlation. Higher absolute values that are closer to '+1' imply strong relationship between variables [59]. This method is used to identify feature relationships that are vital to get a robust model with high accurate intrusion detection rate.

4. Experimental Setup

Our experimental setup utilizes a computer server equipped with a powerful configuration, comprising a 16 GB NVIDIA Graphics Card RTX 4090 Ti, a 3.0 GHz multithread Core i9 CPU, and 64 GB of RAM. We employ Anaconda Python 3.6 software to execute our program and evaluate the effectiveness of the proposed technique.

4.1. Proposed System Framework

The proposed technique of this work, illustrated in Figure 1, involves several phases. Initially, the system starts with fetching the network security dataset that contains various attacks, focusing on leveraging new and pertinent dataset for experimental purposes. Data preprocessing is considered as a critical step to treat the inherent noise, empty values, or inconsistencies of the collected data from different sources. The preprocessing includes refining and cleaning dataset, removing incorrect values, to futher enhance the accuracy of the next analyses. Afterwrds, feature selection method is used to remove redundant or inconsequential features that could significantly affect the model's accuracy and complexity. Employing Pearson's Correlation analysis is necessary to eliminate unrelated and undesirable features and to ensure that only essential pertinent features to the study's objectives are retained.

As for the deep neural networks, models are usually classified into several layers that include interconnected neurons. Each neuron processes incoming data through weighted connections and activation functions to obtain the output. Ultimately, the framework produces poisoned data-based threat. In this study, the generation of such data has been achieved using the DeepFool as a white box attack methodology, in which these adversarial samples are injected into the training dataset at different rates to evaluate their impact on the model performance.

The final step of the framework contains an analysis of the model's efficiency and resilience against these adversarial threats. Several evaluation metrics are leveraged to test the performance of the system, highlighting the strength of the system once adversarial instances are injected. A thorough investigation of the system's behavior under adversarial attacks is achieved using our comprehensive framework, in order to reveal the vulnerabilities of the system and fortify its resilience in real-world applications.

4.2. Dataset Description and Preprocessing Procedures

The dataset leveraged in this work, namely CIC2019, has been recently produced by the Canadian Cyber Security Institute, to further fix flaws and address limitations that have been recorded in the two previous versions, CICIDS2017 and CICIDS2018 datasets [38]. The new version of this dataset has been collected during a period of two days from a recording network flow, and then the recorded data instances are stored in a CSV format [60]. The CIC2019 dataset involves eighty-seven features divided into normal and malware intrusion samples. Table 2 provides more information regarding the dataset. It also includes an extensive collection of valuable samples that belong to the Distributed Denial of Service (DDOS)-based threats.

An important preliminary step involved initializing and preprocessing the dataset is required to remove common presence of undesirable data, including but not limited to noise, irregularities, outliers, and null values. The given dataset goes through two primary steps: cleaning and standardization. Initially, certain columns that involve "Flow ID", "SimillarHTTP", "Timestamp", "Source IP", and "Destination IP" have been eliminated to further reduce the complexity of the dataset and keep it with only eighty-two features. The "Flow Bytes" column has been standardized via replacing missing and infinite values with zeros to make the data consistent. In fact, real-world datasets usually include features with different units, magnitudes, and ranges to standardize the dataset and guarantee that the features are uniformed in ML models. During the scaling methods, the features have been rescaled to a range between zero and one in order to ensure homogeneity and enhance

the performance of the ML algorithm. Note that some ML models, e.g., Neural Networks (NNs), assign standardized input data patterns for optimization purposes.

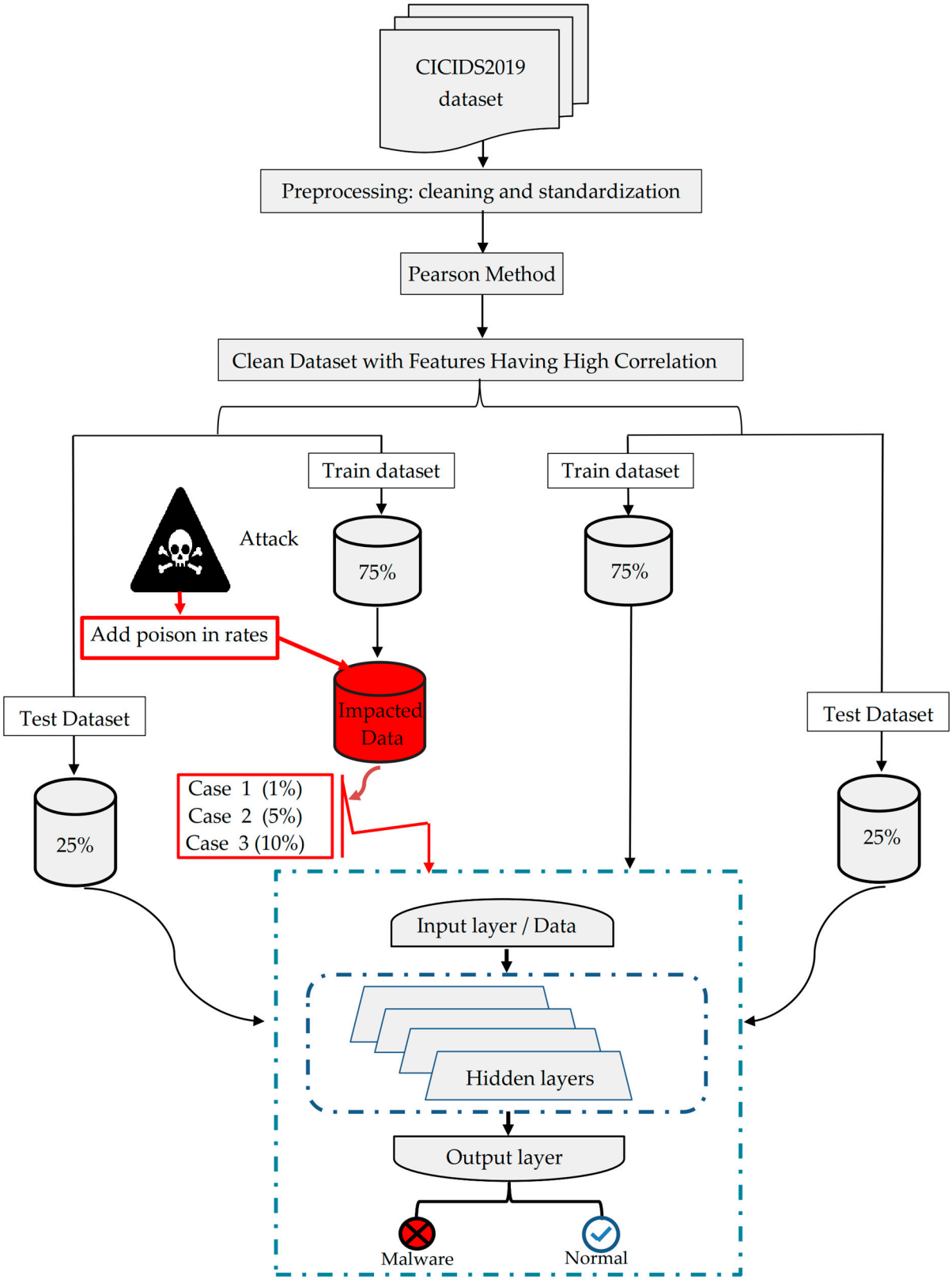

Figure 1. The architecture of our proposed technique.

In this work, the dataset has been divided into 75% for training and 25% for testing through all levels to assure a comprehensive evaluation on the performance and generalizability of the ML model.

Table 2. The number of normal (benign) and malware (threat) data in the CIC2019 network dataset.

Type	Number
Normal	14,040
Malware	99,219

4.3. Performance Evaluation

To assess the performance of the model, we employ the Confusion Matrix (CM), a pivotal tool used for binary classification purposes to identify the differences between normal (benign) and malware (threat) data samples. CM provides a complete malfunction regarding the performance of the ML model, and it contains four primary elements: True Positive (True-P) refers to the accurate classification of normal instances, True Negative (True-N) denotes the correct identification of deceitful threats, False Positive (False-P) implies the misclassification of normal activity in the model as malware, and False Negative (False-N) represents the incorrect labeling of data attacks as normal tasks. Such elements are fundamental in order to successfully evaluate the accuracy of the ML model. Moreover, we incorporated different measurements, in which each has been designed for a specific task. Among these, the primary measurements utilized include:

1. Accuracy (Acc): Represents the percentage of successfully classified records in the whole dataset after training the algorithm.
2. Positive Predictive Value (PPV): Refers to the percentage of successfully recognized threat samples from all predicted threats.
3. False Positive Rate (FPR): Signifies the proportion of incorrect distinguished threat samples to the total number of actual threats.
4. Mean Squared Error (MSE): Represents the average squared dissimilarity between the predictions of the ML model and the original monitored outcomes. While MSE is traditionally used in regression problems rather than binary classification tasks, we employ MSE in our study for specific analytical purposes. Our objective is to analyze and monitor the changes in the system's behavior when different ratios of poisoning data are applied. By utilizing MSE, we are able to quantify the deviations between the predicted and original outputs, providing valuable insights into the impact of poisoning attacks on model performance.

5. Experimental Methodology

This section of our work depicts the utilized experimental procedures to evaluate the detection effectiveness of the DL model against data poisoning-based attacks in two steps. The first phase involves assessing the model's performance using pristine, unaltered data, while the second phase focuses on updated data exposed to poisoning attacks generated through the DeepFool method. These poisoned points are strategically inserted into the training dataset to deceive and compromise the model. Following each experimental phase, an extensive evaluation of the DL model's performance is conducted. The subsequent subsections provide comprehensive details for each phase of the experimentation.

5.1. Phase-I Experiment

In the first phase of the experiments, a DL model is adopted for intrusion detection. The architecture of this model comprises four hidden layers with a neuron configuration of 200, 100, 50, and 20 neurons, respectively, for the input and output layers. Activation functions—ReLu for the hidden layers and Softmax for the output layer—are applied to classify normal and malware labels. The model's parameters are detailed in Table 3 below. Once the model's hyperparameters are fine-tuned and the dataset is preprocessed,

the model training commences using features derived through Pearson correlation-based feature deduction method as depicted in Figure 2. In Figure 2, the correlation strength of the linear relationship between features is depicted. The correlation coefficient value ranges from 1 to −1. A value of 1 signifies a strong positive linear correlation, depicted by a dark color in the figure. Conversely, a value of −1 indicates a strong negative linear correlation, portrayed in a light color. Values close to 1, typically above 0.60, are considered a strong correlation, suggesting a significant relationship between the features. Conversely, correlation coefficients below this threshold are regarded as weak correlations and are often disregarded in the analysis. The extracted features utilized in our analysis are outlined in Table 4.

Table 3. The parameters of DNN model.

Hyperparameter	Value
Learning rate	0.001
Batch size	512
Epochs	20
Loss function	Binary cross-entropy
Dropout	0.4
Optimizer	Adam
Activation function -hidden layers	ReLU
Activation function -output layer	Sigmoid

Table 4. High correlation features.

Features	Correlations
Source Port	0.885809
Protocol	0.805606
Total Length of Fwd Packets	0.797673
Fwd Packet Length Max	0.952294
Fwd Packet Length Min	0.992838
Fwd Packet Length Mean	0.988572
Flow Bytes/s	0.643855
Flow Packets/s	0.557369
Fwd Packets/s	0.558265
Min Packet Length	0.994147
Max Packet Length	0.825010
Packet Length Mean	0.988031
URG Flag Count	0.679175
Down/Up Ratio	0.691483
Average Packet Size	0.990900
Avg Fwd Segment Size	0.988572
Subflow Fwd Bytes	0.797673
Init_Win_bytes_forward	0.503775
Inbound	0.940320

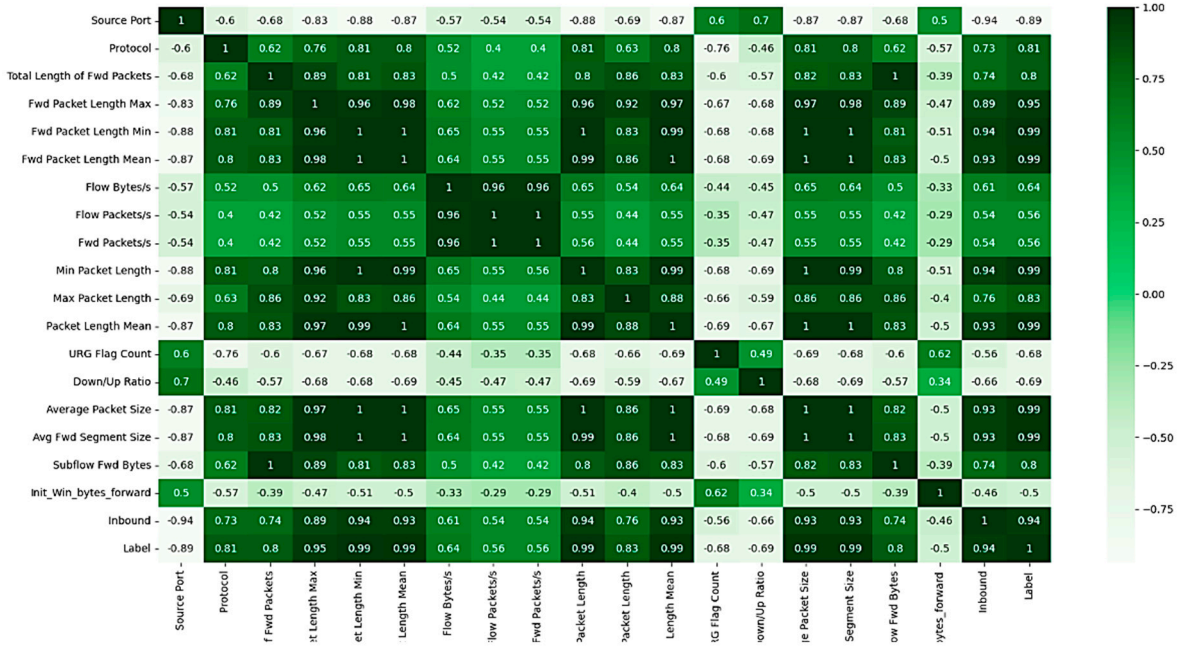

Figure 2. The correlations between features based Pearson method.

The model initiates training using these non-crafted (clean) data features. Subsequently, the testing phase follows the training, focusing on a binary classification problem (zero for normal labels and one for malware labels). Multiple evaluation metrics detailed in Section 4.3 are employed to assess the model's efficiency and performance. The experiment's results are analyzed and compared to the subsequent section's outcomes.

5.2. Phase-II Experiment

In the second phase of the experiments, the same DL model undergoes multiple tests to explore potential attacks during the training. The objective is to determine whether varying rates of poisoned data impact the system detection's functionality. Poisonous points are generated from attributes selected based on their correlation, replicating the features present in the original dataset. These highly correlated features are chosen not to compromise the model but to align the generated points with the original data to reduce the performance of the model. The creation of poisoned points and data updates is executed using the DeepFool white-box method. The investigation aims to understand the influence of increased data insertion rates (1%, 5%, 10%, and 50%) on the model's efficiency in each experiment case. The training set is segregated from the testing set to observe the model's natural performance and accuracy when dealing with contaminated data. The primary goal of this attack is not to manipulate the model's output but to influence the classifier's decision-making process through an untargeted attack.

6. Results

The pursuit of enhancing DL-based NIDS has garnered considerable attention among researchers. Various methodologies have surfaced, encompassing diverse technologies, tools, algorithms, datasets, and benchmarks. Notably, many researchers have adopted DL algorithms in their investigations, each utilizing distinct datasets. The efficacy of our system can be readily gauged through its accuracy, when compared to previous works, which are detailed in Table 5 below.

Table 5. Comparison of DL-based NIDS Accuracy with Existing Studies.

Authors	Year	Dataset	Technique	Acc
Tang [61]	2018	NSL-KDD	GRU-RNN	0.89
Le [62]	2019	NSL-KDD	RNN	0.89
Kim [15]	2020	CICIDS2017	CNN-LSTM	0.93
Choras [63]	2021	CICIDS2017	ANN	0.99
Fu, Zeyuan [64]	2022	IADA, IADB	BiLSTM-DNN	0.97
Proposed idea	2024	CICIDS2019	DL-ANN	0.99

Figure 3 depicts the accuracy curve of our DL model after training on selected features in the first phase, in which the generated poisoned data are not included. The curve demonstrates a steady increase as epochs progress, indicating the suitability of the chosen features for training the system and the sufficiency of epochs in achieving high accuracy. Additionally, Figure 4 illustrates the model's loss, indicating a consistent decrease with increasing epochs.

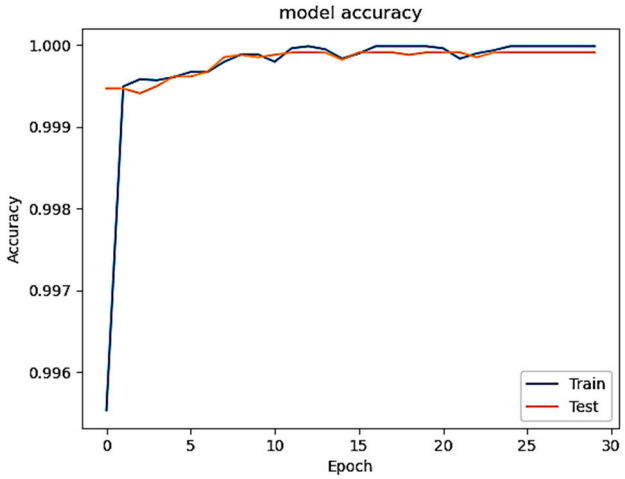

Figure 3. DL accuracy curve on clean data.

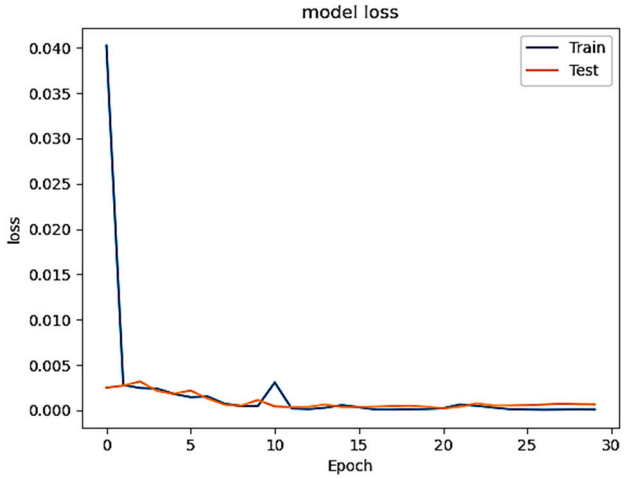

Figure 4. DL loss curve on clean data.

In the study's second phase, we evaluated the performance of the model after injecting three different ratios, 1%, 5%, and 10%, of constructed poisoned patterns into the training set, where the inserting has been performed randomly to make the attack more realistic (closer to the real-world scenario) and render revealing the attack much harder once a traditional defense technique is employed. The main purpose of leveraging this technique is to appraise the effectiveness of the injected poisoned data to the model performance. Remarkably, even after injecting all three percentages of patterns into the original training set while isolating the test data, the accuracy remained relatively high. Figures 5–7 display the accuracy across these three cases, while Figures 8–10 visualize the corresponding model loss.

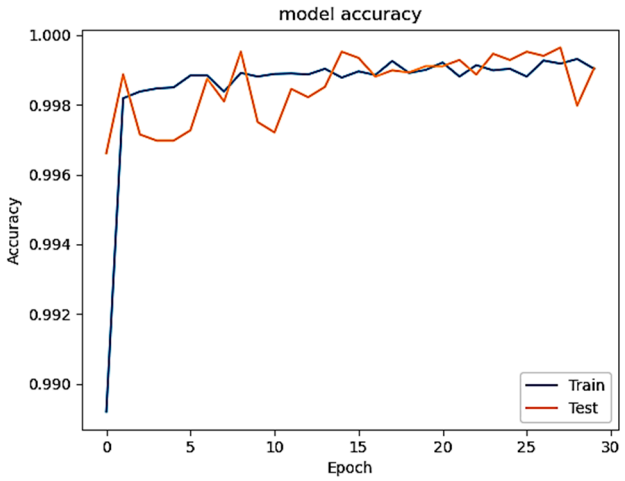

Figure 5. DL accuracy curve—case 1 (1000 training samples).

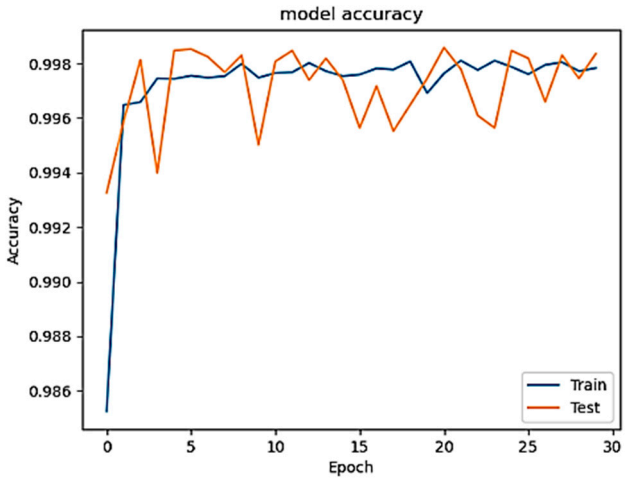

Figure 6. DL accuracy curve -case 2 (5000 training samples).

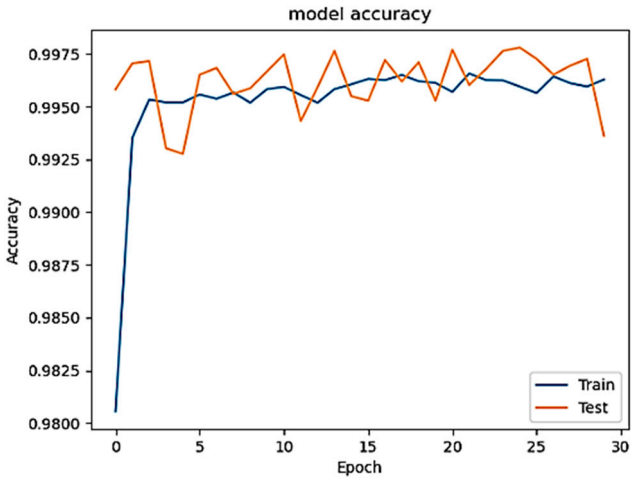

Figure 7. DL accuracy curve-case 3 (10,000 training samples).

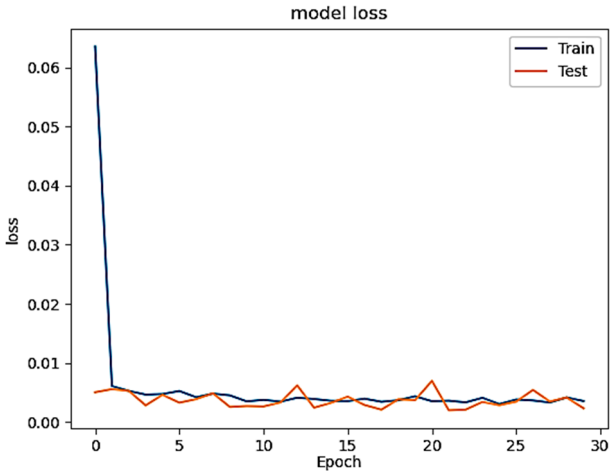

Figure 8. DL loss curve-case 1.

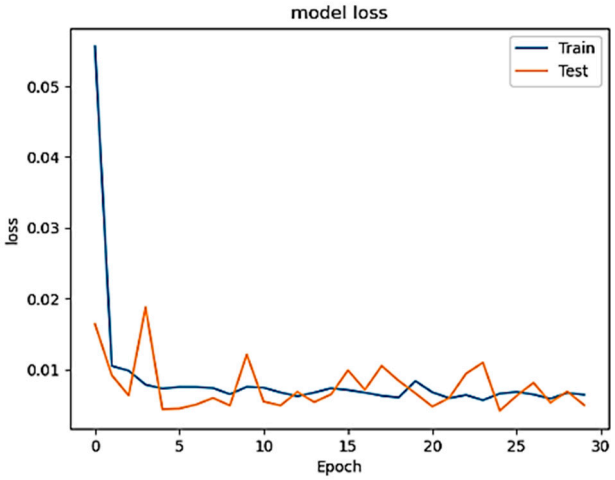

Figure 9. DL loss curve-case 2.

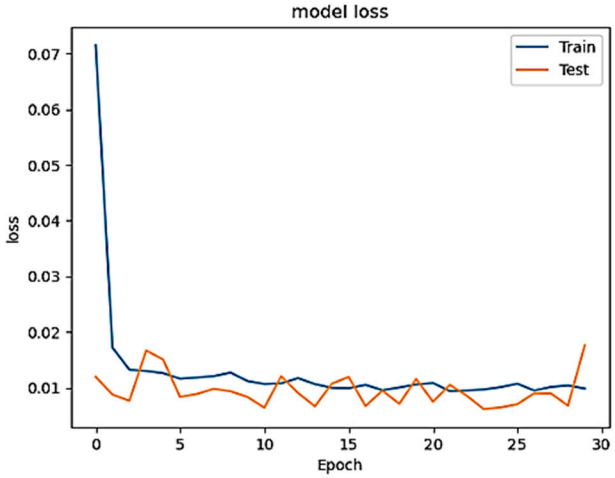

Figure 10. DL loss curve-case 3.

During the first injection of the poisoned data with the ratio of 1% (1000) into the training set, there is no much difference in the result of the Acc when comparing the trained model with and without injecting poisoned data, but the MSE has increased and PPV has decreased. When the 5% ratio dataset is injected, the MSE and PPV have been further impacted, and the Acc has slightly affected. When the poisoned data ratio is increased to 10%, the MSE and PPV have been more affected while still the Acc has barely changed. In general, the effectiveness of the injected poisoned data to the model is summarized in Table 6, in which the impact of the Acc, PPV, FPR and MSE are reported when the ratio of the injected poisoned samples is elevated. This explains the effect of the randomly injected points within the training group, and how the rate of misclassification increases as the injection rate elevates.

Table 6. Summarized the obtained results.

DNN Model Training on	Count	Acc	PPV	FPR	MSE
Clean data (19 features)		0.999	0.999	1.00	0.000008
Poisoned data with different number of samples	1000 (1%)	0.998	0.96	0.998	0.001
	5000 (5%)	0.995	0.92	0.998	0.004
	10,000 (10%)	0.992	0.89	0.971	0.007
	Support (50%)	0.932	0.082	0.295	0.67

To provide an additional proof, we injected half of the training set with the generated poisoned data (50%) to further show the effectiveness of the injected poisoned data on the model performance. The result indicates a significant drop in accuracy, as shown in Figures 11 and 12 and also in the prediction, as summarized in Table 6. To sum up, the poisoned data are considered strong if the PPV, FPR, and MSE are significantly impacted while the accuracy is slightly influenced, and this is clearly shown in Table 6. In other words, the injected poisoned data, that are carefully constructed from highly correlated features and randomly distributed into the training data, cause deluding the model and ensuring that these generated patterns are highly concealed since the model fails to detect such poisoned data. Unlike other techniques, e.g., Flipping Label method, in which the accuracy has dramatically decreased and resulted in exposing the attack and the position of the enemy. This type of attack poses a significant threat as it directly manipulates the data, rendering it invisible to traditional detection methods. Its insidious nature demands highly skilled defenders capable of promptly processing and validating data to mitigate potential damage to system outputs. While these attacks may not immediately impact the

accuracy of the system, other metrics utilized in this study revealed their detrimental effects. Additionally, deep learning-based systems exhibit inherent weaknesses, rendering them particularly vulnerable to such attacks unless robust preventive measures are implemented.

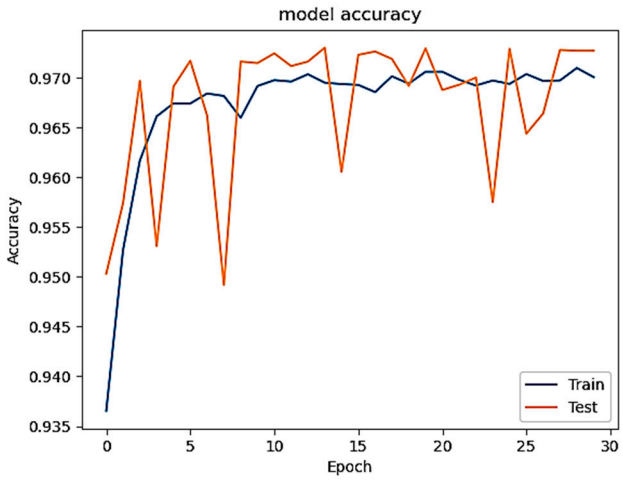

Figure 11. Accuracy curve for supported case.

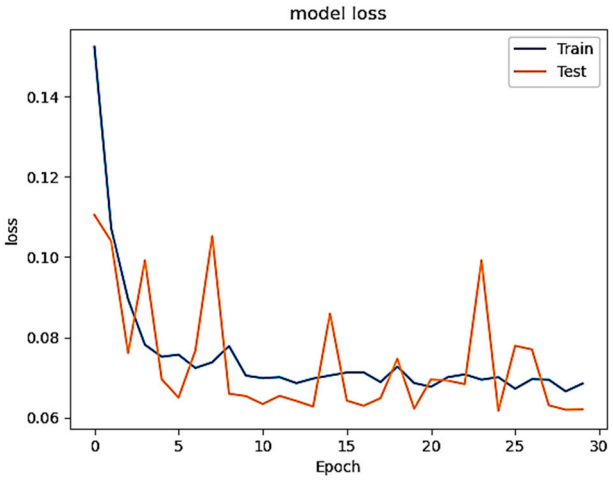

Figure 12. Loss curve for supported case.

Considering the overall performance of our proposed technique shown in Table 6, we can conclude that deep learning model can be considered as a reliable system for detecting intrusion tasks on network field. However, defense techniques against strong poisoned data that are carefully generated leveraging highly correlated features with large data volume are required, especially for sensitive information that are pertinent to medical healthcare field [65]. Also, analyzing the model's performance of deep neural networks using other types of generation methods, particularly when the target is to attack the model, not just the data should be taken into consideration during the evaluation. Moreover, it is not enough to say that the model is strong when only the accuracy is high. As shown in this work, an attacker can inject poisoned data into the training set and such data cannot be detected since the accuracy is still high, where the impact of such attack is shown on other metrics, e.g., precision.

7. Conclusions and Future Works

In this study, we investigate the robustness of a DL model employed for the detection of computer network attacks. Our research is centered on examining the susceptibility of the model towards adversarial attacks, particularly focusing on the impact of such attacks on its performance and resilience. Experiments were conducted utilizing the widely recognized CICIDS2019 network dataset. Employing Pearson correlation coefficient method for feature deduction, we meticulously identified relevant features crucial for effective attack detection. The DeepFool white-box attacks has been included to carefully evaluate the resilience of the model against strong poisoning-based threats, which are designed to evade the system's detection and significantly affect the performance of the model. Our analysis, supported by experimental results, indicates that the generated poisoned data is highly concealed and hard to be detected, which is shown in accurate results that range from 0.99 to 0.93, which indicates the significant concealment achieved by the proposed threats. It also has a significant impact on the performance of the model to correctly recognize normal from malignant instances in the network, as evidenced by variations in Positive Predictive Value (PPV) ranging from 0.99 to 0.082, False Positive Rate (FPR) from 1.00 to 0.29, and Mean Squared Error (MSE) from 0.000008 to 0.67. Also, our proposal aims to explore alternative techniques to further improve the robustness of the system, including but not limited to the selection of more distinguished features, experimenting different features, and exploring alternative deep learning architectures. Furthermore, future attempts will include multiclass classification tasks and investigating various feature selection methods to generate better defense mechanisms and efficient strategies against complicated network threats. This holistic approach underscores our commitment to advance the state-of-the-art in network security, with a keen focus on developing resilient and adaptive defense mechanisms capable of mitigating emerging threats effectively.

Author Contributions: Conceptualization, Q.A. and M.A.; methodology, S.A., Q.A. and M.A.; software, S.A. and Q.A.; visualization, S.A. and Q.A.; validation, S.A., Q.A., J.-S.Y. and M.A.; investigation, S.A., Q.A. and M.A.; writing, review, and editing, S.A., Q.A., J.-S.Y. and M.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Amanuel, S.V.A.; Ameen, S.Y.A. Device-to-device communication for 5G security: A review. *J. Inf. Technol. Inform.* **2021**, *1*, 26–31.
2. Piplai, A.; Chukkapalli, S.S.L.; Joshi, A. NAttack! Adversarial Attacks to bypass a GAN based classifier trained to detect Network intrusion. In Proceedings of the 2020 IEEE 6th Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing,(HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS), Baltimore, MD, USA, 25–27 May 2020; pp. 49–54.
3. Ahmed, I.M.; Kashmoola, M.Y. Threats on machine learning technique by data poisoning attack: A survey. In Proceedings of the Advances in Cyber Security: Third International Conference, ACeS 2021, Penang, Malaysia, 24–25 August 2021; Revised Selected Papers 3; pp. 586–600.
4. Khalid, L.F.; Ameen, S.Y. Secure Iot integration in daily lives: A review. *J. Inf. Technol. Inform.* **2021**, *1*, 6–12.
5. Ibitoye, O.; Abou-Khamis, R.; Matrawy, A.; Shafiq, M.O. The Threat of Adversarial Attacks on Machine Learning in Network Security—A Survey. *arXiv* **2019**, arXiv:1911.02621.
6. Apruzzese, G.; Colajanni, M.; Ferretti, L.; Marchetti, M. Addressing adversarial attacks against security systems based on machine learning. In Proceedings of the 2019 11th International Conference on Cyber Conflict (CyCon), Tallinn, Estonia, 28–31 May 2019; pp. 1–18.
7. Ayub, M.A.; Johnson, W.A.; Talbert, D.A.; Siraj, A. Model evasion attack on intrusion detection systems using adversarial machine learning. In Proceedings of the 2020 54th Annual Conference on Information Sciences and Systems (CISS), Princeton, NJ, USA, 18–20 March 2020; pp. 1–6.
8. Apruzzese, G.; Andreolini, M.; Ferretti, L.; Marchetti, M.; Colajanni, M. Modeling realistic adversarial attacks against network intrusion detection systems. *Digit. Threat. Res. Pract. (DTRAP)* **2022**, *3*, 31. [CrossRef]
9. Alahmed, S.; Alasad, Q.; Hammood, M.M.; Yuan, J.-S.; Alawad, M. Mitigation of Black-Box Attacks on Intrusion Detection Systems-Based ML. *Computers* **2022**, *11*, 115. [CrossRef]

10. Alasad, Q.; Hammood, M.M.; Alahmed, S. Performance and Complexity Tradeoffs of Feature Selection on Intrusion Detection System-Based Neural Network Classification with High-Dimensional Dataset. In Proceedings of the International Conference on Emerging Technologies and Intelligent Systems (Virtual Conference), Virtual, 2–3 September 2022; pp. 533–542.
11. Ring, M.; Wunderlich, S.; Scheuring, D.; Landes, D.; Hotho, A. A survey of network-based intrusion detection data sets. *Comput. Secur.* **2019**, *86*, 147–167. [CrossRef]
12. Izmailov, R.; Venkatesan, S.; Reddy, A.; Chadha, R.; De Lucia, M.; Oprea, A. Poisoning attacks on machine learning models in cyber systems and mitigation strategies. In Proceedings of the Disruptive Technologies in Information Sciences VI, Orlando, FL, USA, 3–7 April 2022; p. 1211702.
13. Amjad, A.; Khan, L.; Chang, H.-T. Semi-natural and spontaneous speech recognition using deep neural networks with hybrid features unification. *Processes* **2021**, *9*, 2286. [CrossRef]
14. Phyo, P.P.; Byun, Y.-C. Hybrid Ensemble Deep Learning-Based Approach for Time Series Energy Prediction. *Symmetry* **2021**, *13*, 1942. [CrossRef]
15. Kim, A.; Park, M.; Lee, D.H. AI-IDS: Application of deep learning to real-time Web intrusion detection. *IEEE Access* **2020**, *8*, 70245–70261. [CrossRef]
16. Sahlol, A.T.; Abd Elaziz, M.; Tariq Jamal, A.; Damaševičius, R.; Farouk Hassan, O. A novel method for detection of tuberculosis in chest radiographs using artificial ecosystem-based optimisation of deep neural network features. *Symmetry* **2020**, *12*, 1146. [CrossRef]
17. Lin, J.; Dang, L.; Rahouti, M.; Xiong, K. Ml attack models: Adversarial attacks and data poisoning attacks. *arXiv* **2021**, arXiv:2112.02797.
18. Huang, Y.; Verma, U.; Fralick, C.; Infantec-Lopez, G.; Kumar, B.; Woodward, C. Malware evasion attack and defense. In Proceedings of the 2019 49th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W), Portland, OR, USA, 24–27 June 2019; pp. 34–38.
19. Hashemi, M.J.; Cusack, G.; Keller, E. Towards evaluation of nidss in adversarial setting. In Proceedings of the 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks, Orlando, FL, USA, 9 December 2019; pp. 14–21.
20. Peng, X.; Huang, W.; Shi, Z. Adversarial attack against DoS intrusion detection: An improved boundary-based method. In Proceedings of the 2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), Portland, OR, USA, 4–6 November 2019; pp. 1288–1295.
21. Gu, T.; Dolan-Gavitt, B.; Garg, S. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv* **2017**, arXiv:1708.06733.
22. Jagielski, M.; Oprea, A.; Biggio, B.; Liu, C.; Nita-Rotaru, C.; Li, B. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In Proceedings of the 2018 IEEE symposium on security and privacy (SP), San Francisco, CA, USA, 21–23 May 2018; pp. 19–35.
23. Jagielski, M.; Severi, G.; Pousette Harger, N.; Oprea, A. Subpopulation data poisoning attacks. In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, 15–19 November 2021; pp. 3104–3122.
24. Alatwi, H.A.; Morisset, C. Adversarial machine learning in network intrusion detection domain: A systematic review. *arXiv* **2021**, arXiv:2112.03315.
25. Mirza, A.H. Computer network intrusion detection using various classifiers and ensemble learning. In Proceedings of the 2018 26th Signal Processing and Communications Applications Conference (SIU), Izmir, Turkey, 2–5 May 2018; pp. 1–4.
26. Waskle, S.; Parashar, L.; Singh, U. Intrusion detection system using PCA with random forest approach. In Proceedings of the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2–4 July 2020; pp. 803–808.
27. McCarthy, A.; Andriotis, P.; Ghadafi, E.; Legg, P. Feature Vulnerability and Robustness Assessment against Adversarial Machine Learning Attacks. In Proceedings of the 2021 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), Dublin, Ireland, 14–18 June 2021; pp. 1–8.
28. Fitni, Q.R.S.; Ramli, K. Implementation of ensemble learning and feature selection for performance improvements in anomaly-based intrusion detection systems. In Proceedings of the 2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), Bali, Indonesia, 7–8 July 2020; pp. 118–124.
29. Nelson, B.; Barreno, M.; Chi, F.J.; Joseph, A.D.; Rubinstein, B.I.; Saini, U.; Sutton, C.; Tygar, J.D.; Xia, K. Exploiting machine learning to subvert your spam filter. *LEET* **2008**, *8*, 16–17.
30. Tian, Z.; Cui, L.; Liang, J.; Yu, S. A Comprehensive Survey on Poisoning Attacks and Countermeasures in Machine Learning. *ACM Comput. Surv.* **2022**, *55*, 1–35. [CrossRef]
31. Carlini, N.; Wagner, D. Towards evaluating the robustness of neural networks. In Proceedings of the 2017 IEEE Symposium on Security and Privacy (sp), San Jose, CA, USA, 22–24 May 2017; pp. 39–57.
32. Tan, Z. The Defence of 2D Poisoning Attack. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4171523 (accessed on 15 January 2024).
33. Muñoz-González, L.; Pfitzner, B.; Russo, M.; Carnerero-Cano, J.; Lupu, E.C. Poisoning attacks with generative adversarial nets. *arXiv* **2019**, arXiv:1906.07773.

34. Zhu, C.; Huang, W.R.; Li, H.; Taylor, G.; Studer, C.; Goldstein, T. Transferable clean-label poisoning attacks on deep neural nets. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 10–15 June 2019; pp. 7614–7623.
35. Moosavi-Dezfooli, S.-M.; Fawzi, A.; Frossard, P. Deepfool: A simple and accurate method to fool deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2574–2582.
36. Debicha, I.; Cochez, B.; Kenaza, T.; Debatty, T.; Dricot, J.-M.; Mees, W. Review on the Feasibility of Adversarial Evasion Attacks and Defenses for Network Intrusion Detection Systems. *arXiv* **2023**, arXiv:2303.07003.
37. Chen, J.; Zheng, H.; Su, M.; Du, T.; Lin, C.; Ji, S. Invisible poisoning: Highly stealthy targeted poisoning attack. In Proceedings of the Information Security and Cryptology: 15th International Conference, Inscrypt 2019, Nanjing, China, 6–8 December 2019; Revised Selected Papers 15; pp. 173–198.
38. Sharafaldin, I.; Lashkari, A.H.; Hakak, S.; Ghorbani, A.A. Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In Proceedings of the 2019 International Carnahan Conference on Security Technology (ICCST), Chennai, India, 1–3 October 2019; pp. 1–8.
39. Woo, J.-h.; Song, J.-Y.; Choi, Y.-J. Performance enhancement of deep neural network using feature selection and preprocessing for intrusion detection. In Proceedings of the 2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), Okinawa, Japan, 11–13 February 2019; pp. 415–417.
40. Xu, C.; Shen, J.; Du, X.; Zhang, F. An intrusion detection system using a deep neural network with gated recurrent units. *IEEE Access* **2018**, *6*, 48697–48707. [CrossRef]
41. Peng, W.; Kong, X.; Peng, G.; Li, X.; Wang, Z. Network intrusion detection based on deep learning. In Proceedings of the 2019 International Conference on Communications, Information System and Computer Engineering (CISCE), Haikou, China, 5–7 July 2019; pp. 431–435.
42. Fernández, G.C.; Xu, S. A case study on using deep learning for network intrusion detection. In Proceedings of the MILCOM 2019–2019 IEEE Military Communications Conference (MILCOM), Norfolk, VA, USA, 12–14 November 2019; pp. 1–6.
43. Aldallal, A. Toward Efficient Intrusion Detection System Using Hybrid Deep Learning Approach. *Symmetry* **2022**, *14*, 1916. [CrossRef]
44. Biggio, B.; Nelson, B.; Laskov, P. Poisoning attacks against support vector machines. *arXiv* **2012**, arXiv:1206.6389.
45. Geiping, J.; Fowl, L.; Huang, W.R.; Czaja, W.; Taylor, G.; Moeller, M.; Goldstein, T. Witches' brew: Industrial scale data poisoning via gradient matching. *arXiv* **2020**, arXiv:2009.02276.
46. Xiao, H.; Biggio, B.; Brown, G.; Fumera, G.; Eckert, C.; Roli, F. Is feature selection secure against training data poisoning? In Proceedings of the International Conference on Machine Learning, Lille, France, 7–9 July 2015; pp. 1689–1698.
47. Hurwitz, J.; Kirsch, D. *Machine Learning for Dummies*; IBM Limited, Ed.; John Wiley & Sons: Hoboken, NJ, USA, 2018.
48. Sandelin, F. Semantic and Instance Segmentation of Room Features in Floor Plans Using Mask R-CNN. Master's Thesis, Uppsala Universitet, Uppsala, Sweden, 2019.
49. Williams, J.M. Deep Learning and Transfer Learning in the Classification of EEG Signals. 2017. Available online: https://digitalcommons.unl.edu/computerscidiss/134/ (accessed on 15 January 2024).
50. Al-Dujaili, A.; Huang, A.; Hemberg, E.; O'Reilly, U.-M. Adversarial deep learning for robust detection of binary encoded malware. In Proceedings of the 2018 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 24 May 2018; pp. 76–82.
51. Chakraborty, S.; Krishna, R.; Ding, Y.; Ray, B. Deep learning based vulnerability detection: Are we there yet. *IEEE Trans. Softw. Eng.* **2021**, *48*, 3280–3296. [CrossRef]
52. Chen, X.; Liu, C.; Li, B.; Lu, K.; Song, D. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv* **2017**, arXiv:1712.05526.
53. Chen, H.; Koushanfar, F. Tutorial: Toward Robust Deep Learning against Poisoning Attacks. *ACM Trans. Embed. Comput. Syst.* **2023**, *22*, 42. [CrossRef]
54. Zhou, S.; Liu, C.; Ye, D.; Zhu, T.; Zhou, W.; Yu, P.S. Adversarial Attacks and Defenses in Deep Learning: From a Perspective of Cybersecurity. *ACM Computing Surveys.* **2022**, *55*, 8. [CrossRef]
55. Michels, F.; Uelwer, T.; Upschulte, E.; Harmeling, S. On the vulnerability of capsule networks to adversarial attacks. *arXiv* **2019**, arXiv:1906.03612.
56. Jakubovitz, D.; Giryes, R. Improving dnn robustness to adversarial attacks using jacobian regularization. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 514–529.
57. Basavegowda, H.S.; Dagnew, G. Deep learning approach for microarray cancer data classification. *CAAI Trans. Intell. Technol.* **2020**, *5*, 22–33. [CrossRef]
58. Aboueata, N.; Alrasbi, S.; Erbad, A.; Kassler, A.; Bhamare, D. Supervised machine learning techniques for efficient network intrusion detection. In Proceedings of the 2019 28th International Conference on Computer Communication and Networks (ICCCN), Valencia, Spain, 29 July–1 August 2019; pp. 1–8.
59. Hidayat, I.; Ali, M.Z.; Arshad, A. Machine Learning-Based Intrusion Detection System: An Experimental Comparison. *J. Comput. Cogn. Eng.* **2023**, *2*, 88–97. [CrossRef]
60. Rizvi, S.; Scanlon, M.; Mcgibney, J.; Sheppard, J. Application of artificial intelligence to network forensics: Survey, challenges and future directions. *IEEE Access* **2022**, *10*, 110362–110384. [CrossRef]

61. Tang, T.A.; Mhamdi, L.; McLernon, D.; Zaidi, S.A.R.; Ghogho, M. Deep recurrent neural network for intrusion detection in sdn-based networks. In Proceedings of the 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft), Montreal, QC, Canada, 25–29 June 2018; pp. 202–206.
62. Le, T.-T.-H.; Kim, Y.; Kim, H. Network intrusion detection based on novel feature selection model and various recurrent neural networks. *Appl. Sci.* **2019**, *9*, 1392. [CrossRef]
63. Choraś, M.; Pawlicki, M. Intrusion detection approach based on optimised artificial neural network. *Neurocomputing* **2021**, *452*, 705–715. [CrossRef]
64. Fu, Z. Computer network intrusion anomaly detection with recurrent neural network. *Mob. Inf. Syst.* **2022**, *2022*, 6576023. [CrossRef]
65. Fatehi, N.; Alasad, Q.; Alawad, M. Towards Adversarial Attacks for Clinical Document Classification. *Electronics* **2023**, *12*, 129. [CrossRef]