A Review On Machine Learning Based Models For Identifying Potential Adversarial And Poisoning Attacks

Viketan Verma[1], Dr. Sanmati Jain[2]
[2] Associate Professor
[1, 2] Vikrant University, Gwalior, India

Abstract- *Machine learning (ML) has revolutionized data-driven decision-making across sectors such as healthcare, finance, defense, and cybersecurity. However, as its influence grows, so does its vulnerability to adversarial and poisoning attacks. Adversaries exploit the weaknesses of ML models to manipulate outputs or degrade system performance, posing significant risks in critical applications. As a result, developing machine learning-based models to detect and counter such attacks has become essential for building secure and trustworthy AI systems. One major area of research that has emerged is the detection of poisoning for android systems using neural networks due to the complexity of data set of attacks. Several approaches have been used so far for the effective classification of poisoning attacks. The paper investigates the different contemporary neural network based approaches used thus far in the detection of poisoning attacks. The approaches used and their findings have been illustrated with their salient points. Moreover, an analysis in the form of shortcoming in previous work has been cited so as to define a problem statement to work upon.*

Keywords- Adversarial Machine Learning, Poisoning Attacks, Neural Networks, Adversarial Training, Accuracy of classification

I. INTRODUCTION

Sophisticated attackers have strong incentives to manipulate the results and models generated by machine learning algorithms to achieve their objectives. For instance, attackers can deliberately influence the training dataset to manipulate the results of a predictive model in poisoning attacks. It can be inferred that these attacks become easier tomount today as many machine learning models need to be updated regularly to account for continuously-generated data. Such scenarios require online training, in which machine learning models are updated based on new incoming training data. For instance, in cyber-security analytics, new Indicators of Compromise (IoC) rise due to the natural evolution of malicious threats, resulting in updates to machine learning models for threat detection. These IoCs are collected from online platforms, in which attackers can also submit IoCs of their choice. In personalized medicine, it is envisioned that patient treatment is adjusted in real- time by analyzing information crowdsourced from multiple participants [5]. By controlling a few devices, attackers can submit fake information (e.g., sensor measurements), which is then used for training models applied to a large set of patients. Defending against such poisoning attacks is challenging with current techniques. Methods from robust statistics (e.g, [7], [6]) are resilient against noise but perform poorly on adversarial-poisoned data, and methods for sanitization of training data operate under restrictive adversarial models [12]. One fundamental class of supervised learning is linear regression. Regression is widely used for prediction in many settings (e.g., insurance or loan risk estimation, personalized medicine, market analysis). In a regression task a numerical response variable is predicted using a number of predictor variables, by learning a model that minimizes a loss function. Regression is powerful as it can also be used for classification tasks by mapping numerical predicted values into class labels. Assessing the real impact of adversarial manipulation of training data in linear regression, as well as determining how to design learning algorithms resilient under strong adversarial models is not yet well understood.

II. THE POISONING ATTACK STRUCTURE

The poisoning attack structure is based on the fact that the adversaries try to train the neural networks with malicious data which leads to false classification in the testing phase. The structure of the poisoning attack is shown in the figure below.

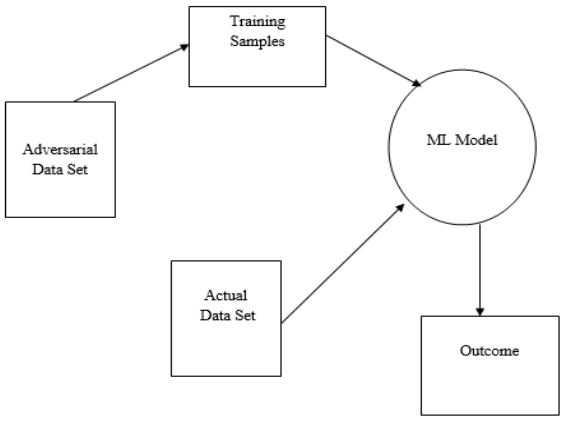

Fig.1 Concept of the poisoning attack.

The poisoning attack can be mathematically summarized as:
Let the training vector be:

$$Training\ Data = X(i) \quad (1)$$

Manipulating the training vector is done using the poisoning vector given by:

$$X_v = V(i) \quad (2)$$

The weights of the system are governed the training vector and learning algorithm, given by:

$$w_k = f(X_v, k, f_a) \quad (3)$$

Here,
$X(i)$ is the real training input
$V(i)$ is the poisoning vector
k is the number of iteration
fa is the activation function
w_k is the weight for iteration k.

III. PREVIOUS WORK

Paudice et al.in [1] proposed linear regression learning models for detection of poisoning attacks on neural network architectures. The approach incorporates a fast statistical technique for the classification of malwares with the aim of non-requirement of large data sets. The analysis for poisoning attack have been made on health care, loan assessment, and real estate domains. The major challenge however remained the fact that as the level of poisoning increases, the accuracy of the classification system generally dips thereby degrading the system performance. Detection of increasing poisoning has also not been detected successfully for the neural network architecture to adapt to.

Li et al. in [2] proposed Reasoning About Future Cyber-Attacks Through Socio-Technical Hacking Information. With the widespread of cyber-attack incidents, cyber security has become a major concern for organizations. The waste of time, money and resources while organizations counter irrelevant cyber threats can turn them into the next victim of malicious hackers. In addition, the online hacking community has grown rapidly, making the cyber threat landscape hard to keep track of. In this work, we describe an AI tool that uses a temporal logical framework to learn rules that correlate malicious hacking activity with real-world cyber incidents, aiming to leverage these rules for predicting future cyber-attacks. The framework considers socio-personal and technical indicators of enterprise attacks, analysing the hackers and their strategies when they are planning cyber offensives online.

Jiang et al. in [3] proposed a technique KUAFUDET, which is a two step adversarial detection mechanism that uses detecting adversal behaviour. The training phase is offline with feature extraction done and the testing phase is online. The major challenge identified in this work is the fact that the classification accuracy of a neural based architecture greatly depends upon the injected data to the neural network for training. Manoeuvring or manipulating the data used for training can substantially change the performance of the system. It is often difficult to detect anomalous data that is fed to the neural network for training. Deliberate injections can often lead to adversarial of the network used for analysis.

Malin et al. in [4] explain non-iid setting on time series forecasting problems. Authors used a linear autoregressive model to detect the influence of target setting of time series predicting neural architectures. The detection accuracy was analysed with influencing the targets by adversaries. The main challenge which was addressed was the fact that with influencing the actual targets or rather by changing the targets and replacing them with even slightly different adversarial targets in the training phase, it is possible to immensely change the testing performance of the system. This is particularly seen to be sensitively increase the accuracy of prediction problems.

Muñoz-González al. in [5] proposed the use of back gradient optimization for adversarial of deep learning applied to neural networks. The authors showed that several online applications need neural networks to constantly gather information and data and learn from them to fine tune themselves to a particular application. This however exposes the neural networks to data adversarial which is well crafted malicious data. The challenge is devising adversarial of multi class problems and even more challenging is the detection of the same.

Shen et al. in [6] proposed the detection of adversarial attacks for collaborative deep learning applications. The challenge which was identified was the fact that in several applications, deep learning based models can collaborate among each other to share data and devise an accurate model which is better than individual models. This has the downside of making the collaborative deep learning based system prone to malicious training and collaborative learning among users intertwined with adversaries who deliberately would collaborate to mis-train the deep learning model

Papernot et al. in [7] proposed a mechanism to detect adversarial of deep learning based models using the knowledge of the mapping between the inputs and targets. The mechanism is intelligent to negate the effects of adversarial manipulation of the targets. The approach clearly indicates that the adversarial training of the deep learning based models or otherwise clearly indicate the mis-classification problem among neural network architectures.

Russu, et al. in [8] proposed the use of optimization and game theoretical models for the detection of adversarial training of neural networks. The authors showed that the use of secure kernels can be used to thwart the possibility of adversarial and adversarial attacks. The kernel function of the neural network is critical in proving immunity against adversarial attacks. The major challenge that the approach addressed is the fact that it is not always possible to keep a bound on the amount of the data that is fed to a neural networks which rely on online learning for weight adaptation. The proper selection of the kernel is also challenging to find maximum immunity against adversarial .

Zhang et al. in [9] proposed a methodology for feature selection to thwart the possibility of adversarial attacks on neural network based classifications. The authors showed that it is difficult to test a neural network for the possibility of false training by adversaries since neural networks need to learn from real time data for online applications. In this case, it becomes critical to choose features which can actually help to find out whether the data fed to the neural network for online training in real time critical situations is authentic or tailor made to instil adversarial

Xiao et al. in [10] proposed the use of Support Vector Machine (SVM) to detect the possibility of adversarial attacks on neural networks. In this case the proposition made was the following. The data that is acquired by neural networks can be both authentic and genuine or can also be falsified. Attackers may try to devise data sets particularly aimed at training the neural network to get trained in such a way that it is trained with misleading target vectors. Thus the neural network relates the inputs vector with the misleading target vectors thereby training incorrectly. This significantly increases the chances of the neural network malfunctioning. The proposed work does not directly feed the neural network with the data that it acquires from sources for training. In this case, first the data, both the inputs and the targets are fed to a support vector machine to find out the possibilities of adversarial or adversarial falsification. If the data has a reasonable measure of being authentic and unaltered by adversarial intervention, then it is fed to the neural network for training

IV. PROBLEMS IDENTIFIED IN PREVIOUS WORK

The outcomes of the review infer that Poisoning Classification is an important method that is used as a part of Android security applications. Better poisoning recognition mechanisms can help in better provisions of security against unauthorized intrusions. Using advanced methods like Artificial Intelligence for the recognition purpose, it can help in improved accuracy and better outcomes. The major classification governing factors are:

1) The number of features and the category of features are critical for accuracy. Co-occurrence features and texture based features generally render moderate accuracy. [1], [3], [4]
2) The classification tool plays another important rule. Off late, the deep neural networks have become prevalent for classifications [3], [4], [5], [7]. Convolutional neural networks have also been effective for classification.
3) Clustering techniques can also be useful for finding the data set which can yield the most useful information.[2],[10]

Moreover, due to different features selected and different algorithms for training, the accuracy varies for different techniques.. The accuracy variation is seen to be between 80% and 90% for various data sets. The aim should be to increase the accuracy beyond the previously existing techniques. The training algorithm can modified and so can be the features for more effective training.

The mathematical model for the data extraction model is expressed as:

The general function of social hacking is to gain access to restricted information or to a physical space without proper permission. Most often, social hacking attacks are achieved by impersonating an individual or group who is directly or indirectly known to the victims or by representing an individual or group in a position of authority.In this case, it is assumed that the data on the profiles of hackers on dark web

resources can render information about future trends and aspects of cyber attacks. The mathematical model of extraction of data from dark web forums is given below:

$$W(v_i, v_j) = \frac{1}{M}\sum_n \forall a, b: V(M, a) \quad (4)$$

$$Or, W(v_i, v_j) = v_i^{(\beta\alpha^{(time, M_{k,b})} - (time, M_{k,a}))} + V(M_k, b) \quad (5)$$

Here,
F is a dark web forum
W is the correlation between weights
n is the number of threads analysed
v is the number of users posting messages
M is the message number/index
k is the time index
(M_k, a)&(M_k, b) are the messages at time index k for distinct posts a and b in the same thread K.
α, β are constants with values between 0 and 1.
v_i, v_j are distinct messages

Another approach for estimating the similarity co-efficient or the distance among the messages is given mathematically as:
For two lists Γ^1&Γ^2 in the forum 'F', the similarity co-efficient or distance is computed as:

$$D^p(\Gamma^1, \Gamma^2) = \sum_{\forall\, i,j \in D(\Gamma^1\Gamma^2)} \hat{D}^p_{i,j}(\Gamma^1\Gamma^2) \quad (6)$$

Here,
Γ^1&Γ^2 are two lists
D^p is the distance with a penalty p
$\hat{D}^p_{i,j}$ takes up fuzzy values for different levels of similarity
(i,j) are the message pair
P is the optimistic penalty parameter

The distance measure (Kendall's Measure) takes the relative ranking orders of any two elements in the union of two top k lists. Another measure is the absolute distance between the rankings of the same element in the union of two top k lists into consideration called the Spearman's distance measure given mathematically as:

$$F^{k+1}(\Gamma^1, \Gamma^2) = \sum_{i \in D_{r1} \cap D_{r2}} |\Gamma_1' - \Gamma_2'| \quad (7)$$

Here,
F represents the Spearman's distance
D_{ri}&D_{r2} represent the domains of $\boldsymbol{\Gamma^1}$ *and* $\boldsymbol{\Gamma^2}$
$\boldsymbol{\Gamma_1'}, \boldsymbol{\Gamma_2'}$ denoted the lists with/without entries in the original lists.

V. EVALUATION PARAMETERS

Often an artificial intelligence based approach is used for the classification and renders the following issues.

1. **True Positive (TP):** It is indicative of the true or correct cases of the data to be in a particular class.
2. **True Negative (TN):** It is indicative of the true or correct cases of the data not to be in a particular class.
3. **False Positive (FP):** It is indicative of the false or incorrect cases of the data to be in a particular class.
4. **False Negative (FN):** It is indicative of the false or incorrect cases of the data not to be in a particular class.

Sensitivity (S_e): It is indicative the ratio in which a data set is categorized Mathematically it can be defined as:

$$S_e = \frac{TP}{TP+FN} \quad (8)$$

Accuracy (Ac): It is an indicative of the accuracy of classification of the algorithm for data classification, Mathematically its defined as:

$$A_c = \frac{TP+TN}{TP+TN+FP+FN} \quad (9)$$

VI. CONCLUSION

From the previous discussions, it can be concluded that classified and secure applications on the Android platform need reliable security systems to prevent unauthorized trespassing. The paper presents a survey on the various contemporary techniques used for the poisoning detection and their summary of gaps or limitations found in the previous work. The problems found in the previous work is useful for devising strategies for further research.

REFERENCES

[1] A Paudice, L Muñoz-González, A Gyorgy, EC Lupu, "Detection of adversarial training examples in poisoning attacks through anomaly detection", IEEE Transactions on Dependable and Secure Computing, 2023, pp.1-10.
[2] G. Li, J. Wu, S. Li, W. Yang and C. Li, "Multitentacle Federated Learning Over Software-Defined Industrial Internet of Things Against Adaptive Poisoning Attacks,"

in IEEE Transactions on Industrial Informatics, 2022, vol. 19, no. 2, pp. 1260-1269

[3] W. Jiang, H. Li, S. Liu, X. Luo and R. Lu, "Poisoning and Evasion Attacks Against Deep Learning Algorithms in Autonomous Vehicles," in IEEE Transactions on Vehicular Technology, vol. 69, no. 4, pp. 4439-4449, April 2020

[4] E. Marin, M. Almukaynizi and P. Shakarian, "Reasoning About Future Cyber-Attacks Through Socio-Technical Hacking Information," IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI), 2019, pp. 157-164.

[5] L Muñoz-González, B Biggio, A Demontis, "Towards adversarial of deep learning algorithms with back-gradient optimization", ACM 2017

[6] S Shen, S Tople, P Saxena," Auror defending against adversarial attacks in collaborative deep learning systems", ACM 2016

[7] N Papernot, P McDaniel, S Jha,"The limitations of deep learning in adversarial settings", IEEE 2016

[8] P Russu, A Demontis, B Biggio, G Fumera," Secure kernel machines against evasion attacks", ACM 2016

[9] F Zhang, PPK Chan, B Biggio," Adversarial feature selection against evasion attacks", IEEE 2015

[10] H Xiao, B Biggio, B Nelson, H Xiao," Support vector machines under adversarial label contamination", Elsevier 2015

[11] B Biggio, SR Bulò, I Pillai, M Mura," Adversarial complete-linkage hierarchical clustering", Springer 2014

[12] M Mozaffari-Kermani, S Sur-Kolay," Systematic adversarial attacks on and defenses for machine learning in healthcare", IEEE 2014

[13] X Lin, PPK Chan," Causative attack to incremental support vector machine", IEEE 2014

[14] N Hoque, MH Bhuyan, RC Baishya, "Network attacks: Taxonomy, tools and systems", Elsevier 2014

[15] B Biggio, I Corona, D Maiorca, B Nelson, "Evasion attacks against machine learning at test time", Springer 2013

[16] B Biggio, B Nelson, P Laskov, "Adversarial attacks against support vector machines", Axirv 2012

[17] B Biggio, I Corona, G Fumera, G Giacinto, "Bagging classifiers for fighting adversarial attacks in adversarial classification tasks", Springer 2011

[18] J Long, W Zhao, F Zhu, Z Cai, "Active Learning to Defend Adversarial Attack Against Semi-Supervised Intrusion Detection Classifier", World Scientific Journal 2011

[19] N Alexiou, S Basagiannis, P Katsaros, "Formal analysis of the kaminsky DNS cache-adversarial attack using probabilistic model checking", IEEE 2010

[20] S Son, V Shmatikov, "The hitchhiker's guide to DNS cache adversarial ", Springer 2010

[21] H Lin, R Ma, L Guo, P Zhang, "Conducting routing table adversarial attack in DHT networks", IEEE 2010

[22] J Trostle, B Van Besien, A Pujari, "Protecting against DNS cache adversarial attacks", IEEE 2010

[23] C Wallenta, J Kim, PJ Bentley, S Hailes, "Detecting interest cache adversarial in sensor networks using an artificial immune algorithm", Springer 2010