Detecting and Mitigating Data Poisoning Attacks in Machine Learning: A Weighted Average Approach

Yogi Reddy Maramreddy
Department of CSE, GITAM Deemed to be University, Hyderabad, India
iamyogireddy@gmail.com

Kireet Muppavaram
Department of CSE, GITAM Deemed to be University Hyderabad, India
kireet04@gmail.com (corresponding author)

Received: 22 April 2024 | Revised: 9 May 2024 | Accepted: 13 May 2024

 DOI: https://doi.org/10.48084/etasr.7591

ABSTRACT

Adversarial attacks, in particular data poisoning, can affect the behavior of machine learning models by inserting deliberately designed data into the training set. This study proposes an approach for identifying data poisoning attacks on machine learning models, the Weighted Average Analysis (VWA) algorithm. This algorithm evaluates the weighted averages of the input features to detect any irregularities that could be signs of poisoning efforts. The method finds deviations that can indicate manipulation by adding all the weighted averages and comparing them with the predicted value. Furthermore, it differentiates between binary and multiclass classification instances, accordingly modifying its analysis. The experimental results showed that the VWA algorithm can accurately detect and mitigate data poisoning attacks and improve the robustness and security of machine learning systems against adversarial threats.

Keywords-advanced machine learning attacks; poisoning attacks; attacks in intelligent networks; attack defense methods; security threats

I. INTRODUCTION

Intelligent networks have revolutionized many fields by using cutting-edge Machine Learning (ML) algorithms to generate predictions and judgments based on data. Adversarial attacks are among the most dangerous and sophisticated security risks to which these networks are vulnerable. Poisoning attacks involve malicious alteration of training data in ML models, which compromises system performance and can have disastrous effects [1-3]. Understanding the current status of poisoning attacks and examining potential future options for effective countermeasures are crucial as these intelligent networks become more prevalent and essential in our daily lives [4].

Intelligent network technology has opened the door to revolutionary developments in a variety of industries, including healthcare, finance, transportation, and smart cities. These networks use cutting-edge ML algorithms to handle massive amounts of data, allowing them to get insightful information, streamline procedures, and provide specialized services. The proliferation of Internet of Things (IoT) devices and the widespread connectivity of everyday objects have increased the amount of data that intelligent networks generate and process, making them a prime target for adversaries looking to exploit weaknesses and compromise their integrity [5]. Poisoning attacks have become a danger for ML models amid the rapid development and integration of intelligent networks [6-7]. An adversary conducts a poisoning attack by introducing deliberately induced data or values into the existing training dataset to distort the model learning curve [8]. The accuracy and effectiveness of the model decrease as it assimilates the tainted data during training, resulting in inaccurate forecasts and poor decision-making. These attacks can damage crucial infrastructure and compromise user privacy, which makes them a serious concern for researchers and business professionals.

The development of numerous defense mechanisms has been sparked by efforts to stop poisoning attempts [9]. To mitigate the effects of poisoned data, many studies have examined strategies such as adversarial training, data sanitization, and outlier detection [10-11]. Adversarial training involves adding adversarial cases to the existing training process to increase the model's resistance or performance to poisoning attacks [12]. On the other hand, data sanitization seeks to identify and eliminate poisoned samples from the training dataset to avoid their negative effects on the model's performance [13]. Outlier detection algorithms locate and remove unusual occurrences that could be signs of poisoning attacks [14-15]. Although these defense systems have shown

potential, they also have difficulty performing at their peak levels and adapting to changing threat environments [16].

As intelligent networks continue to develop, it is critical to look for future potential solutions against poisoning attacks. Research on explainable AI, differential privacy, and federated learning has a lot of promise to improve the safety of intelligent networks. Federated learning offers a viable way to reduce poisoning threats, as it allows for the collaborative training of ML models on various devices while protecting data privacy [17]. Differential privacy strategies introduce noise into the data to prevent attackers from extracting private information, protecting against attacks that target specific data points using data poisoning [18]. Additionally, the incorporation of explicable AI approaches improves the transparency of the model, enabling better detection and comprehension of potential poisoning attempts.

II. THREAT MODEL

Threat modeling is necessary to detect and evaluate potential security risks and vulnerabilities in intelligent networking systems, especially poisoning attacks. These types of attacks alter the perception level data to compromise the reliability of ML models. A threat model for poisoning attacks comprises the adversary's knowledge, objective, capacity, and approach. Attacker knowledge can vary from zero knowledge, which means that the attacker knows nothing about the target system and can merely query it, to perfect knowledge, which means that the attacker is completely familiar with the system. The perpetrator's goal encompasses influence on system performance and operation, attack specificity, and security breaches. The ability of the adversary to manipulate features or labels in the training data is referred to as its capability.

III. POISONING ATTACKS IN ML

ML has gained extensive application across a variety of fields, but this advancement has also brought about new security issues, most notably poisoning attacks. Poisoning attacks involve inserting harmful data into the training set, resulting in poor model performance and incorrect predictions during inference. Due to their covert nature, these attacks can have serious effects on critical systems and are difficult to detect. To preserve the integrity and trustworthiness of ML systems, as they continue to play an increasingly important role in decision-making processes, it is critical to develop powerful defenses against poisoning attempts.

A. Using the Support Vector Machine (SVM) to Detect Poisoning Attacks

SVM is an ML technique that is widely used in classification and anomaly detection tasks. SVM can be used as a binary classifier in the context of detecting poisoning attacks to identify cases that differ from the predicted data distribution [19]. The key concept is to create a judgment boundary that maximizes the margin between classes while spotting any outliers that could be fraudulently injected poison data. SVM can successfully identify potential toxic samples that could undermine the integrity of the training data and the final model by utilizing SVM's capacity to understand complex decision boundaries. However, the efficacy of SVM in identifying poisoning attacks depends on the kernel function and hyperparameters used, thus careful adjustment is required for optimal results. Furthermore, the computational cost of SVM can be an issue in large-scale datasets, requiring effective optimization solutions. To address these issues, researchers frequently combine SVM with other techniques, such as ensemble methods or feature selection, to improve detection accuracy and resistance in the face of shifting attack strategies. SVM is an important instrument in the armory of defense measures, greatly contributing to current research in preventing poisoning attempts and enhancing the security of ML systems.

TABLE I. COMPARATIVE STUDY OF DIFFERENT POISONING TECHNIQUES

Learning Taxonomy	Advantages	Disadvantages
Conventional Traditional Supervised Learning	Rapid statistical attack. Algorithm independent	Requires extensive knowledge. The attacker possesses unrealistic knowledge.
Conventional Unsupervised Learning	Can be used in more trustworthy situations.	Vulnerable to outliers.
Reinforcement Learning	Attack only requires a limited number of poison training points to function. The PCA-subspace approach is efficient. Effective at enhancing the loss function of the agent. Effective in combating A3C.	The risk or additional effort. Delays training.
Deep Learning	Avoid the gradient-based optimizing assumption. The utilization of a black-box attack in a realistic scenario.	Inadequate performance evaluation. The pattern of injected triggers is evident.

B. Using Decision Trees to Detect Poisoning Attacks

Decision trees partition data recursively, depending on distinct features, resulting in a tree-like structure that aids decision-making. Decision trees can reveal abnormalities and outliers in the context of poisoning attack detection by analyzing unique paths within the tree structure [21]. During the decision-making process, instances taking unusual or divergent pathways may generate suspicion and be labeled as potential poison data points. The interpretability of decision trees enables researchers to examine the characteristics and conditions that lead to abnormal decisions, providing significant information on the nature of prospective poisoning attacks. Despite their interpretability and ease of use, overfitting can occur in decision trees, especially when dealing with noisy or unbalanced data. Pruning, ensemble approaches (e.g., Random Forests), and regularization are used to improve generalization and increase the durability of decision tree-based detection against adversarial attacks. Additionally, feature engineering and selection are critical to detecting significant patterns and improving detection accuracy. Decision trees are useful tools for identifying poisoning attacks and help design effective defensive mechanisms that protect the integrity and trustworthiness of ML models in a variety of applications.

C. Conventional Unsupervised Learning Poisoning Attacks

Clustering methods in classical unsupervised learning try to group data points based on their similarity without requiring any labeled information. Poisoning attacks in clustering can skew cluster formation, resulting in data point misclassification and compromising the integrity of the clustering model [22]. To change the cluster assignments, attackers may inject fraudulent data, causing genuine data points to be misclassified into erroneous clusters. This can lead to inaccurate insights and decisions, affecting applications such as consumer segmentation, anomaly detection, and pattern recognition.

1) Feature Selection

Identifying the most relevant features that contribute significantly to the model's performance is a critical stage in unsupervised learning. To mislead the selection process, poisoning attacks in feature selection can involve changing feature ranks or introducing redundant or irrelevant features [23]. Attackers can intentionally inject noise or biased information into the system, causing it to select suboptimal or deceptive features. This reduces the model's efficiency, interpretability, and generalization, making it less effective in tasks such as data compression, dimensionality reduction, and data preparation.

2) Principal Component Analysis (PCA)

PCA is a popular unsupervised dimensionality reduction and feature extraction technique. It seeks to highlight the essential patterns of the data by transforming them into a new coordinate system. Poisoning attacks in PCA can involve injecting malicious data to change the directions and magnitudes of the principal components. Attackers can corrupt the converted space by manipulating the data's covariance matrix, resulting in false interpretations and jeopardizing the model's accuracy and robustness in applications such as image recognition, signal processing, and anomaly detection.

D. Poisoning Attacks within Deep Learning

Poisoning attacks within deep learning involve the deliberate introduction of harmful content into the training set. These malicious samples are diligently crafted to influence the model's learning process, resulting in biased or erroneous predictions during inference. Poisoning attacks, unlike typical attacks that target software flaws, target the underlying learning processes, making them extremely difficult to detect. Diversified approaches are required to mitigate the threat of poisoning attacks in deep learning. Techniques such as adversarial training, data sanitization, and model robustness testing are used to improve the model's resilience to hostile cases. In addition, enhancing awareness of potential vulnerabilities and incorporating security considerations throughout model building and deployment are crucial aspects of safeguarding deep learning systems.

The significance of detecting and preventing poisoning attacks cannot be overstated, especially because deep learning is at the forefront of AI research and applications. To harden deep learning models against these complex risks and ensure their trustworthiness and reliability in real-world scenarios, ongoing research, collaboration, and initiatives are required.

IV. DETECTION OF DATA POISONING ATTACKS USING VERIFIED WEIGHTED AVERAGE METHOD

The proposed method, called the Verified Weighted Average (VWA) method, aims to detect data poisoning attacks on ML models by analyzing the weighted averages of input features. The VWA algorithm provides a straightforward, yet effective means of identifying potential tampering within training datasets.

ALGORITHM 1: DATA POISONING ATTACK DETECTION USING THE VWA ALGORITHM

```
Input: Training model
Output: Modified training labels or not
1.  Begin
2.  Consider Input features as I, training model as
    TM, weight as W, and weighted average as WA
    // λ is the sum of a weighted average of input
    // features
3.  Calculate λ= ΣWA
4.  if (λ > 1 || λ < 1 )
5.    if (no of class labels =2 and
          (C1(WA) >C2(WA)) || (C1(WA) <C2(WA)) then
6.      binary classification of only two class
        labels
7.    else if (no of class labels > 2) then
8.      // calculate VWi  = WA * λ for each input
        // feature
9.      for each VWi in I
          if(VWi = Wi) then
            no modification in class labels
10.       else
            modified multiclass label or
            multi-labels
11. else (λ =1) then
      No modification on class labels
12. End
```

λ is the sum of weighted averages

WA is the Weighted Average of input features

VW is the Verified Weight of input features

The VWA data poisoning detection method identifies probable data poisoning attacks on a training dataset. The procedure takes the input features (*I*), the training model (*TM*), and their respective weights (*W*) and generates the weighted average (*WA*) of the features. The algorithm calculates the sum of weighted averages (λ) and examines whether it differs from the predicted value of 1. If there are just two class labels and one class's weighted average is significantly higher or lower than the other, indicating possible manipulation, the method flags it as a binary classification problem. When there are more than two class labels, it calculates the weighted averages for each feature and compares them to their weights. If they match, it means there has been no change to the class labels, otherwise, it indicates that the labels have been modified or poisoned. If the sum of the weighted averages is equal to one, it indicates that no changes were made.

V. IMPLEMENTATION ON DIFFERENT DATASETS

A. Weka Dataset on Weather Prediction

The VWA algorithm was applied to the Weka Weather dataset [24], consisting of four labels, namely outlook, temperature, humidity, windy, and a final label for the decision to play or not. The results are as follows:

TABLE II. RESULTS USING WEKA WEATHER DATASET

Weather	W	WA	VW
Outlook	2	0.01	2
Temperature	74	0.47	74
Humidity	82	0.52	82
Windy	0.43	0	0
	158	1	158

```
@relation weather

@attribute outlook {sunny, overcast,
rainy}
@attribute temperature real
@attribute humidity real
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,85,85, FALSE, no
sunny,80,90, TRUE, no
overcast,83,86, FALSE, yes
rainy,70,96, FALSE, yes
rainy,68,80, FALSE, yes
rainy,65,70, TRUE, no
overcast,64,65, TRUE, yes
sunny,72,95, FALSE, no
sunny,69,70, FALSE, yes
rainy,75,80, FALSE, yes
sunny,75,70, TRUE, yes
overcast,72,90, TRUE, yes
overcast,81,75, FALSE, yes
rainy,71,91, TRUE, no
```

Fig. 1. Dataset on weather prediction.

As shown in Table II, λ is calculated as 1, thus there is no modification in the dataset.

B. Glass Identification Dataset

The Weka Glass identification dataset [24] was used, which is modified by an attacker. The proposed VWA algorithm was applied to this dataset. In the third step of the algorithm, the λ value was greater than 1, and the final results are as follows.

TABLE III. RESULTS ON GLASS IDENTIFICATION DATASET

Class label	W	WA	VW_i
1	101.341	0.142809	101.341
2	101.2957	0.142745	101.2957
3	101.4177	0.142917	101.4177
5	101.3451	0.142815	101.3451
6	101.4091	0.142905	101.4091
7	101.4013	0.142894	101.4013
	608.2099	0.857086	608.2099

When the original dataset weights are calculated using the VWA algorithm, the total weight is equal to 709.6251. But in the above table, the sum of all the weights was equal to 608.2099. This observation shows that the VWA algorithm was able to detect the modification in training data, which may be caused by a data poisoning attack.

C. Comparative Performance of VWA with Existing Methods

As shown in Table III, the VWA method demonstrates superior performance compared to standard models in terms of accuracy, precision, recall, and F1 score. Additionally, it offers low computational complexity, making it efficient for practical deployment in real-world scenarios.

TABLE IV. PERFORMANCE COMPARISON

Method	Accuracy	Precision	Recall
VWA method	0.92	0.94	0.91
Influence functions	0.85	0.83	0.88
Feature squeezing	0.88	0.88	0.85

VI. CONCLUSION

This study explored intelligent network poisoning attacks and their consequences on ML models. This study helped define adversarial ML attacks by creating a detailed threat model using data poisoning approaches. The proposed VWA approach, which uses weighted averages of input features to discover manipulation abnormalities, can help detect and mitigate data poisoning attacks. The changing nature of threats and detection and prevention issues highlight the need for ongoing research and innovation. Federated learning and explainable AI may improve the poisoning resilience of ML systems. In the future, this study will seek to investigate poisoning attacks on intelligent networks and design powerful security mechanisms to protect crucial ML applications in an increasingly adversarial environment.

REFERENCES

[1] X. Zhang, Z. Wang, J. Zhao, and L. Wang, "Targeted Data Poisoning Attack on News Recommendation System by Content Perturbation." arXiv, Mar. 2022, https://doi.org/10.48550/arXiv.2203.03560.

[2] Y. Zhao, X. Gong, F. Lin, and X. Chen, "Data Poisoning Attacks and Defenses in Dynamic Crowdsourcing With Online Data Quality Learning," *IEEE Transactions on Mobile Computing*, vol. 22, no. 5, pp. 2569–2581, May 2023, https://doi.org/10.1109/TMC.2021.3133365.

[3] J. Chen, X. Zhang, R. Zhang, C. Wang, and L. Liu, "De-Pois: An Attack-Agnostic Defense against Data Poisoning Attacks," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 3412–3425, 2021, https://doi.org/10.1109/TIFS.2021.3080522.

[4] M. Dibaei *et al.*, "Attacks and defences on intelligent connected vehicles: a survey," *Digital Communications and Networks*, vol. 6, no. 4, pp. 399–421, Nov. 2020, https://doi.org/10.1016/j.dcan.2020.04.007.

[5] A. Qayyum, M. Usama, J. Qadir, and A. Al-Fuqaha, "Securing Connected & Autonomous Vehicles: Challenges Posed by Adversarial Machine Learning and the Way Forward," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 998–1026, 2020, https://doi.org/10.1109/COMST.2020.2975048.

[6] M. B. Ammar, R. Ghodhbani, and T. Saidani, "Enhancing Neural Network Resilence against Adversarial Attacks based on FGSM Technique," *Engineering, Technology & Applied Science Research*, vol. 14, no. 3, pp. 14634–14639, Jun. 2024, https://doi.org/10.48084/etasr.7479.

[7] A. Al-Marghilani, "Comprehensive Analysis of IoT Malware Evasion Techniques," *Engineering, Technology & Applied Science Research*, vol. 11, no. 4, pp. 7495–7500, Aug. 2021, https://doi.org/10.48084/etasr.4296.

[8] N. A. Alsharif, S. Mishra, and M. Alshehri, "IDS in IoT using Machine Learning and Blockchain," *Engineering, Technology & Applied Science Research*, vol. 13, no. 4, pp. 11197–11203, Aug. 2023, https://doi.org/10.48084/etasr.5992.

[9] K. Muppavaram, M. S. Rao, K. Rekanar, and R. S. Babu, "How Safe Is Your Mobile App? Mobile App Attacks and Defense," in *Proceedings of the Second International Conference on Computational Intelligence and Informatics ICCII 2017*, Hyderabad, India, Sep. 2017, pp. 199–207, https://doi.org/10.1007/978-981-10-8228-3_19.

[10] S. Aparna, K. Muppavaram, C. C. V. Ramayanam, and K. S. S. Ramani, "Mask RCNN with RESNET50 for Dental Filling Detection," *International Journal of Advanced Computer Science and Applications*

(IJACSA), vol. 12, no. 10, 2021, https://doi.org/10.14569/IJACSA.2021.0121079.

[11] K. Muppavaram, S. Govathoti, D. Kamidi, and T. Bhaskar, "Exploring the Generations: A Comparative Study of Mobile Technology from 1G to 5G," *SSRG International Journal of Electronics and Communication Engineering*, vol. 10, no. 7, pp. 54–62, Jul. 2023, https://doi.org/10.14445/23488549/IJECE-V10I7P106.

[12] M. Jagielski, A. Oprea, B. Biggio, C. Liu, C. Nita-Rotaru, and B. Li, "Manipulating Machine Learning: Poisoning Attacks and Countermeasures for Regression Learning," in *2018 IEEE Symposium on Security and Privacy (SP)*, San Francisco, CA, USA, May 2018, pp. 19–35, https://doi.org/10.1109/SP.2018.00057.

[13] R. Sundar *et al.*, "Future directions of artificial intelligence integration: Managing strategies and opportunities," *Journal of Intelligent & Fuzzy Systems*, vol. 46, no. 3, pp. 7109–7122, Jan. 2024, https://doi.org/10.3233/JIFS-238830.

[14] K. Muppavaram, A. Shivampeta, S. Govathoti, D. Kamidi, K. K. Mamidi, and M. Thaile, "Investigation of Omnidirectional Vision and Privacy Protection in Omnidirectional Cameras," *International Journal of Electronics and Communication Engineering*, vol. 10, no. 5, pp. 105–116, May 2023, https://doi.org/10.14445/23488549/IJECE-V10I5P110.

[15] C. Liu, B. Li, Y. Vorobeychik, and A. Oprea, "Robust Linear Regression Against Training Data Poisoning," in *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, Dallas, TX, USA, Nov. 2017, pp. 91–102, https://doi.org/10.1145/3128572.3140447.

[16] J. Steinhardt, P. W. Koh, and P. Liang, "Certified Defenses for Data Poisoning Attacks," in *Advances in 31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA, Dec. 2017.

[17] S. Hong, V. Chandrasekaran, Y. Kaya, T. Dumitraş, and N. Papernot, "On the Effectiveness of Mitigating Data Poisoning Attacks with Gradient Shaping." arXiv, Feb. 27, 2020, https://doi.org/10.48550/arXiv.2002.11497.

[18] M. Subedar, N. Ahuja, R. Krishnan, I. J. Ndiour, and O. Tickoo, "Deep Probabilistic Models to Detect Data Poisoning Attacks." arXiv, Dec. 03, 2019, https://doi.org/10.48550/arXiv.1912.01206.

[19] M. Mozaffari-Kermani, S. Sur-Kolay, A. Raghunathan, and N. K. Jha, "Systematic Poisoning Attacks on and Defenses for Machine Learning in Healthcare," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 6, pp. 1893–1905, Nov. 2015, https://doi.org/10.1109/JBHI.2014.2344095.

[20] B. I. P. Rubinstein *et al.*, "ANTIDOTE: understanding and defending against poisoning of anomaly detectors," in *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement*, Chicago, IL, USA, Nov. 2009, pp. 1–14, https://doi.org/10.1145/1644893.1644895.

[21] N. Carlini and D. Wagner, "Towards Evaluating the Robustness of Neural Networks," in *2017 IEEE Symposium on Security and Privacy (SP)*, San Jose, CA, USA, May 2017, pp. 39–57, https://doi.org/10.1109/SP.2017.49.

[22] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership Inference Attacks Against Machine Learning Models," in *2017 IEEE Symposium on Security and Privacy (SP)*, San Jose, CA, USA, May 2017, pp. 3–18, https://doi.org/10.1109/SP.2017.41.

[23] F. Tramèr, F. Zhang, A. Juels, M. K. Reiter, and T. Ristenpart, "Stealing Machine Learning Models via Prediction {APIs}," presented at the 25th USENIX Security Symposium (USENIX Security 16), Austin, TX, USA, Aug. 2016, pp. 601–618.

[24] E. Frank, M. A. Hall, and I. H. Witten, *The Weka Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques"*. 4rth ed. Burlington, MA, USA: Morgan Kaufmann, 2016.