

# SecureLearn - An Attack-agnostic Defense for Multiclass Machine Learning Against Data Poisoning Attacks

Anum Paracha, Junaid Arshad, Mohamed Ben Farah, Khalid Ismail  
College of Computing, Birmingham City University, Birmingham, United Kingdom

**Abstract**—Data poisoning attacks are a potential threat to machine learning (ML) models, aiming to disrupt their learning processes by manipulating the training datasets. Existing defenses are mostly designed to mitigate specific poisoning attacks or are aligned with particular ML algorithms. Furthermore, most defenses are developed to mitigate poisoning attacks in deep neural networks or binary classifiers. However, traditional multiclass classifiers need attention to be secure from data poisoning attacks, as these models are significant in developing multi-modal applications, particularly with limited resources and feature-structured datasets. Therefore, this paper proposes SecureLearn, a two-layer attack-agnostic defense to defend multiclass models from poisoning attacks. It comprises two components of data sanitization and a new feature-oriented adversarial training (FORT). To ascertain the effectiveness of SecureLearn, we proposed a 3D evaluation matrix with three orthogonal dimensions: data poisoning attack, data sanitization and adversarial training. Benchmarking SecureLearn in a 3D matrix, a detailed analysis is conducted at different poisoning levels (10%-20%), particularly analysing accuracy, recall, F1-score, detection and correction rates, and false discovery rate. The experimentation is conducted for four ML algorithms, namely Random Forest (RF), Decision Tree (DT), Gaussian Naive Bayes (GNB) and Multilayer Perceptron (MLP), trained with three public datasets: IRIS, MNIST and USPS, against three poisoning attacks and compared with two existing mitigation techniques. Our results highlight that SecureLearn is effective against the provided attacks in all given models. SecureLearn has strengthened resilience and adversarial robustness of traditional multiclass models and neural networks, confirming its generalization beyond algorithm-specific defenses. It consistently maintained accuracy above 90%, recall and F1-score above 75%, and reduced the false discovery rate to 0.06 across all evaluated models. In the context of neural networks, SecureLearn achieved at least 97% recall and F1-score against all selected poisoning attacks. The adversarial robustness of models, trained with SecureLearn, improved with an average accuracy trade-off of only 3%.

**Index Terms**—Machine Learning, Data Poisoning Attacks, Data Sanitization, Adversarial Training, Feature Importance Score

## I. INTRODUCTION

In recent years, machine learning (ML) has been facilitating outstanding performance in prediction and decision-making tasks. For example, in a recommender system [1], biometric recognition [2], and security-sensitive applications such as skin cancer detection [3], medical imaging [4] and autonomous

vehicles [5]. ML models rely on training datasets to develop their decision-making mechanisms by identifying the underlying patterns in the given data and making predictions independently without additional information.

Despite their outstanding performance, recent studies show that these ML models are susceptible to various adversarial attacks, typically classified as data poisoning attack [6] which perturbs the training dataset, evasion attack [7] which adds manipulations in test data, inversion attack [8] which tends to steal the confidential information of the model and inference attack [9] which tends to identify training dataset. Of these, we focus on data poisoning attacks in multiclass models, which pose serious security threats to ML. For example, outlier-oriented poisoning attack (OOP) [10] manipulates the feature space of the model by perturbing outliers, subpopulation attack (SubP) [11] injects poisoned clusters into the dataset and exploits data sanitization techniques: TRIM [12] and SEVER [13]. Similarly, label-flipping attack [14], [15] is a common data poisoning attack that can be extended as random label poisoning attack (RLPA) in multiclass models. Other successful data poisoning attacks are [6], [16], [17].

Recently, various defenses have been proposed to mitigate data poisoning attacks [18]–[20]. However, these solutions are mostly attack-specific or system-specific, defined to mitigate specific data poisoning attacks or are adaptable to particular algorithms. For example, Hossain et al. [21] proposed a solution to detect backdoor attacks limited to deep neural networks. Baker et al. [16] developed a method to particularly secure recommender systems from data poisoning attacks, which does not defend other systems. Peri et al. [22] removed clean-label poison by detecting falsified data points with k-neighbors; however it is only effective against feature collision and convex polytope attacks. Adversarial training [23], a prominent adversarial defense, is only adaptable in deep learning (DL) as it follows gradient learning. Moreover, various attacks, such as [24]–[26], have successfully breached defenses against data poisoning attacks with evolving attack vectors. Currently, few solutions are proposed that offer attack-agnostic defense, and these solutions are mostly designed for DL models.

Given the above-mentioned limitations, we propose SecureLearn, a two-layer attack-agnostic approach to defend against data poisoning, irrespective of particular attack vectors. SecureLearn offers an enhanced data sanitization that combines the fundamental principles of nearest neighbor voting strategy

to correct data labels, followed by calculating the statistical deviations of each data point to detect and correct anomalies. Furthermore, SecureLearn introduced a new approach of **feature-oriented adversarial training** (FORT) influenced by a common characteristic of feature importance score of ML to identify important data points to generate adversarial examples for training.

To thoroughly assess SecureLearn, we propose a 3D evaluation matrix following three dimensions: data poisoning attacks, data sanitization and adversarial training. The experiments are conducted on four machine learning algorithms: Random Forest (RF), Decision Tree (DT), Gaussian Naive Bayes (GNB), and Multi-layer Perceptron (MLP). Selecting these algorithms allows us to cover most types of classification mechanisms in machine learning. We selected three distinct data poisoning attacks: OOP, SubP and RLPA attacks and set the poisoning levels between 10% and 20% at a scale of 5 to study the effectiveness of SecureLearn in different adversarial settings. We also compare it with two data sanitization defenses given in [20], [27], highlighting enhanced performance and generalization of SecureLearn over others. The contributions of this paper are given as follows:

- To the best of our knowledge, SecureLearn is the first attack-agnostic defense in multiclass classifiers defending against data poisoning attacks. SecureLearn provides defense without requiring prior knowledge of attacks, targeted models and additional data.
- We have proposed a new adversarial training mechanism named Feature-Oriented Adversarial Training (FORT) as a component of SecureLearn, enhancing the adversarial robustness of traditional multiclass ML, including neural networks. Our results show that the adversarial robustness improved with a minimal trade-off between accuracy and robustness, i.e., the accuracy is decreased  $< 3\%$ , while enhancing the adversarial robustness.
- We have proposed a new 3D evaluation matrix to comprehensively evaluate SecureLearn against three data poisoning attacks and compare it with two existing defenses. The evaluation is set up for four types of ML models trained with three distinct datasets. The results highlight that SecureLearn has outperformed other mitigations and is effective against all selected attacks for all models, consistently maintaining accuracy to a minimum 90% and recall and F1-score to 75%.

## II. RELATED WORK

### A. Existing Multiclass Poisoning Attacks

The existing literature highlights various data poisoning attacks that affect the confidentiality, integrity, and availability of multiclass models. Such as Alarab et al. [28] experimentally showed an increase in model variance and prediction uncertainty with a manipulated dataset. They also highlight the limitations of the Monte-Carlo method in detecting poisoned data points near classification boundaries. MetaPoison [6] manipulates the training dataset to fool neural networks. This

attack craft poisoned images by solving bilevel optimization with the Carlini and Wagner attack [29] and achieved a 40-90% success to poison all selected models with a 1% poison budget. They also experimented the MetaPoison on Google Cloud AutoML API and achieved  $> 15\%$  success with a minimum of 0.5% dataset poisoning. Zhao et al. [30] proposed a class-oriented targeted attack to manipulate individual classes in DL models, whereas Lu et al. [31] introduced model poisoning reachability to quantify the limits of targeted poisoning. Munoz-Gonzalez et al. [32] extended gradient optimization poisoning in multiclass DL models. Alongside poisoning modern ML models, certain attacks are introduced to manipulate traditional multiclass models. OOP attack [10] manipulated the feature space of multiclass models by exploiting outliers in the dataset and experimented against six models. Biggio et al. [33] introduced an adversarial label flipping attack to poison class labels indiscriminately, which can be extended to poison multiclass datasets. Jagielski et al. [34] introduced a clean label poisoning that augments a cluster of poisoned points in the training dataset, challenging poison detection as it is difficult to identify a subset of poisoned data points with similar features. Pantelakis et al. [35] experimented JSMA, FGSM and DeepFool attacks to evaluate performance disruption in multiclass IoT networks.

### B. Limitations of Existing Solutions

In contrast, various mitigation techniques are proposed in the literature to secure ML from data poisoning attacks. Such as Neehar et al. [22] developed a deep k-NN to remove clean label poison by detecting falsified data points with k-neighbors. Deep k-NN defense is experimented against feature collision and convex polytope in deep neural networks. Paudice et al. [20] used the k-NN algorithm to mitigate label poisoning in binary SVM. Carnerero-Cano et al. [36] computed limitations of hyperparameters to resist data poisoning impact on DNN models. Barreno et al. [19] have given the concept of reject on negative impact to remove affected data points, which is extended in [27] to filter poisoned data from the given dataset. However, most mitigations are implemented to secure either DL models or are applicable to binary ML models. Limited solutions are provided to secure traditional multiclass models, such as one-versus-one SVM, multiclass GNB, RF or DT algorithms. We also need a mitigation mechanism that is attack-agnostic and can be adaptable to secure ML from evolving data poisoning attacks, i.e., effective against most types of existing and novel data poisoning attacks and can be implemented with various datasets and algorithms in both binary and multiclass settings.

Adversarial training is a prominent defense to improve the adversarial robustness of DL models. Such as [18], [23], [37] have proposed adversarial training methods and implemented in neural networks and DL models as adversarial training is designed following iterative gradient learning, which does not apply to traditional models hence makes adversarial training ineffective in securing traditional ML models.

Conclusively, there are some attack-agnostic solutions pro-

posed in the literature that are mentioned to be adaptable to various data poisoning attacks, while mostly focused on securing DL models. To secure traditional ML, few solutions are proposed; however, these mitigations are experimented to improve the robustness of binary models; however, limited attention is given to traditional models developed in multiclass settings.

SecureLearn is an attack-agnostic solution which is designed to be adaptable to traditional ML and neural networks in multiclass classification settings. It is effective against various aforementioned attacks, providing promising results in various real-world applications. SecureLearn is proposed as a two-layer solution with improved data sanitization and a feature-oriented adversarial training to strengthen model robustness. A brief comparison of various existing solutions with SecureLearn is provided in Table I, highlighting that existing solutions have either proposed data sanitization or adversarial training, where data sanitization solutions are experimented on binary ML models and adversarial training is experimented with only DL models.

### III. THREAT MODEL

#### A. Attacker's Goal

We defined two attacker goals to assess the effectiveness of selected mitigation solutions. The first goal is to disrupt the model's availability and reduce its overall performance by employing the OOP attack [10] and label flipping attack [15]. The second goal is to harm the model's integrity by augmenting clustered poisoned data points employing the subpopulation attack to disrupt targeted class predictions [11]. Consider the poisoning of supervised classification models, e.g. RF or MLP, given the dataset  $D_o = \{(x_i, l_i)\}_{i=1}^n$  with data points  $x$  and labels  $l$ , the attacker can manipulate the labels  $l'$  or the features  $x'$  of the dataset or augment poisoned data points( $x', l'$ ) into the dataset to prevent the trained victim model from attaining the intended performance.

#### B. Attacker's Knowledge

In this threat model, the attacker possesses limited knowledge of the targeted model  $M$  and dataset  $D_o$ . Under these constraints, all selected data poisoning attacks are formulated as gray-box attacks. In this scenario, the attacker has a partial understanding of the dataset and model: the dataset and algorithm names are known, but the dataset distribution, model settings, and parameters remain unknown. Additionally, the attacker has no knowledge and access to the target system.

#### C. Attacker's Capability

We have leveraged the attacker's capability to poison the training datasets in different ways. The attacker can modify labels or features of the dataset and introduce poisoned data points into the dataset. However, this capability is limited to injecting a maximum 20% poisoning level as the upper bound limit and a minimum 10% poisoning as the lower bound limit. These limits are defined as the most effective poisoning limits [10], [39], highlighting  $10\% \leq \Delta L \leq 20\%$  are complacent

poisoning levels, whereas  $\Delta L < 5\%$  has a negligible impact and  $\Delta L > 20\%$  is detectable.

#### D. Attack Strategy

In our attack settings, three data poisoning attacks of varying attack vectors, i.e., OOP, SubP and RLPA attacks are considered. Following these attacks in multiclass classifiers, the effectiveness of SecureLearn is evaluated, demonstrating that it is an attack-agnostic and promising solution capable of mitigating all the aforementioned attacks.

### IV. SECURELEARN DESIGN

We formulate the problem of poisoning the training dataset, given as follows:  $D_c$  is the clean dataset,  $D'_c$  is the poisoned substitute in the dataset formulated as  $D_o = D_c \cup D'_c$ . As no ground truth is provided, SecureLearn aims to sanitize  $D_o$  to correct data points and align features. SecureLearn relies on the general observation that the poisoned dataset tricks the model training to classify differently from the clean dataset, resulting in performance degradation. Therefore, SecureLearn identifies anomalies and misalignments in the features of the data points and their labels. Furthermore, SecureLearn improves the resilience of the model with adversarial training. To achieve this aim, SecureLearn comprises the following two components: data sanitization and feature-oriented adversarial training. The complete process to improve the resilience of the ML model with SecureLearn is illustrated in Fig. 1. The algorithm of SecureLearn is provided in Alg. 1.

---

#### Algorithm 1 SecureLearn Mitigation Mechanism

---

**Input:** Training samples  $X$ , perturbation limit  $\varepsilon$ , feature importance scores:  $F$   
**Initialize:**  $b=0.001$ ,  $c=0.01$ , nearest neighbors ( $k$ )= $7$   
**for**  $x_i \in X$  **do**  
     $d = \min(k, \text{dist}(x_i, x))$   
     $l_i = \text{avg}(x_i, d)$   
     $D_{san} \leftarrow (x_i, l_i)$   
**end for**  
**for**  $x_i \in D_{san}$  **do**  
    Compute  $\delta_i$  following Eq.5.5  
    **if**  $\delta < |g|$  **then**  
         $D_{san} \leftarrow (x_i, l_i)$   
    **end if**  
**end for**  
**if**  $M == M_{GNB}$  or  $M_{MLP}$  **then**  
     $F \leftarrow \arg \max \text{Probability}(D_{san})$   
**end if**  
**if**  $M == M_{RF}$  or  $M_{DT}$ :  
     $F \leftarrow \sum_{i=1}^L f_i(1 - f_i)$   
**end if**  
**for**  $(x_i \in D_{san})$  and  $(f_i \in F)$  **do**  
     $D_{adv} \leftarrow \mathbb{E}_{(x,y) \sim D_o} [\mathcal{L}(M, (x_i + (c * \text{sign}((f_i * x_i) + b)))]$   
**end for**

---

Table I: Summary of existing similar defenses against data poisoning attacks proposed in various settings

| Research paper         | Data Sanitization | Adv. Training | ML model                                              | Model Settings            |
|------------------------|-------------------|---------------|-------------------------------------------------------|---------------------------|
| De-Pois [38]           | ✓                 | ✗             | GAN, CNN and LASSO                                    | Binary and Multiclass DNN |
| A. Paudice et al. [20] | ✓                 | ✗             | Stochastic Gradient Descent                           | Binary ML                 |
| P. PK Chan et al. [27] | ✓                 | ✗             | SVM                                                   | Binary ML                 |
| M. Barreno et al. [19] | ✓                 | ✗             | SVM                                                   | Binary ML                 |
| A. Shafahi et al. [37] | ✗                 | ✓             | ResNet and InceptionV1                                | Multiclass DNN            |
| L. Tao et al. [18]     | ✗                 | ✓             | VGG-16, VGG-19, ResNet-18, ResNet-50 and DenseNet-121 | Multiclass DL             |
| <b>SecureLearn</b>     | ✓                 | ✓             | DT, RF, GNB, NN                                       | Multiclass ML             |

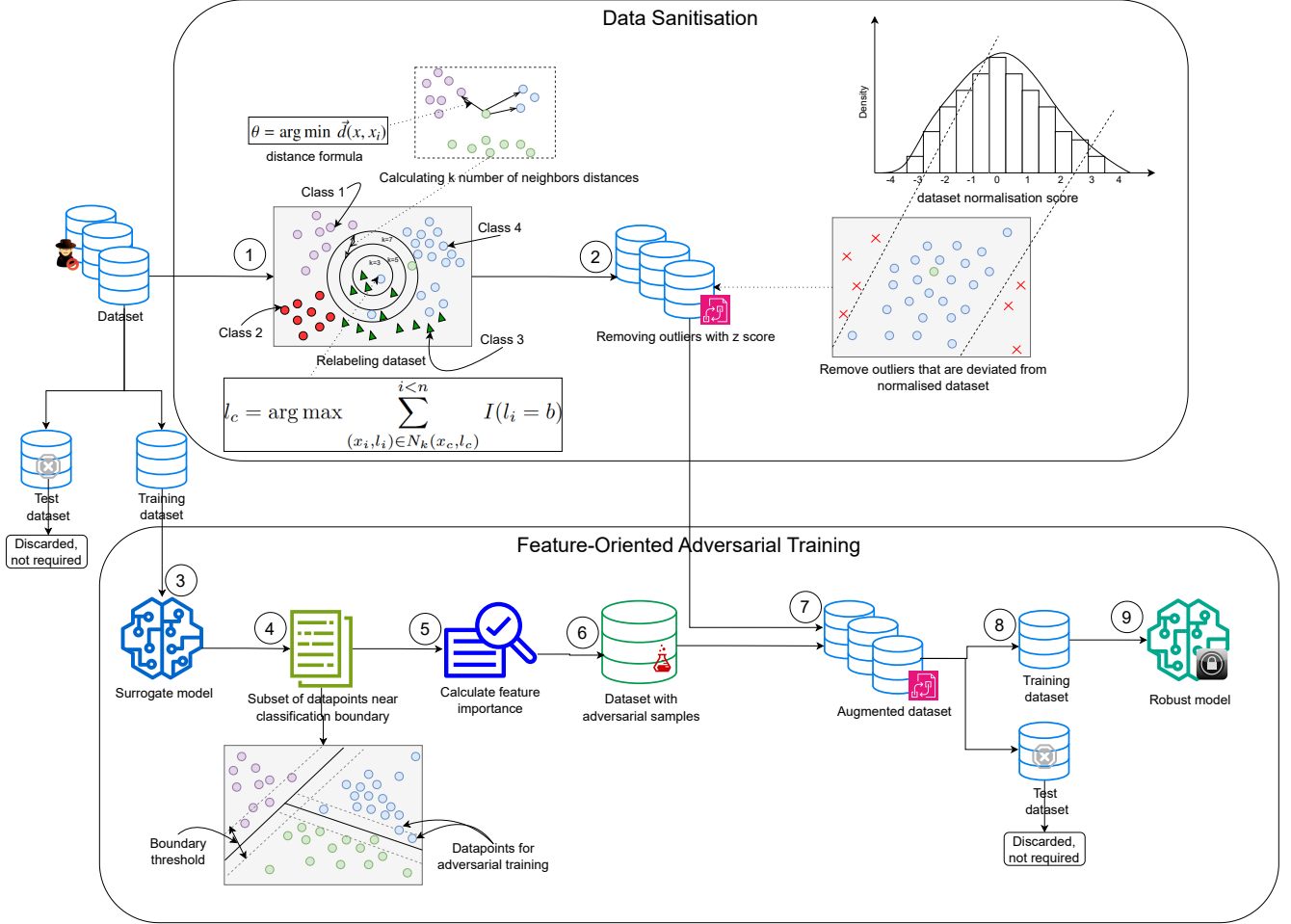


Fig. 1: Architectural overview of SecureLearn illustrating a two-layer approach to secure the training pipeline of ML models irrespective of data poisoning attacks

### A. Data Sanitization

Our data sanitization module comprises two parts: relabeling the dataset  $D_o$  and removing anomalies. Our relabeling mechanism is defined as:

$$D_{san} = \{(x, l) | x \in D_o\}$$

$$\text{and } l = \begin{cases} l_i & \text{if } C(x_i, l_i) < \gamma \\ l & \text{if } C(x_i, l_i) \geq \gamma \end{cases} \quad (1)$$

where  $D_{san}$  is the sanitized dataset,  $C(x, l)$  is the confidence of neighboring data points,  $l_i$  is the existing label of the data point  $x$ , and  $l$  is the new label confident label from the nearest

data points. The confidence limit is defined as  $\gamma \geq 40\%$  neighboring votes, following an incremental majority voting approach [40]. The calculation of the label of each data point, given in Eq. 1, follows the confidence score  $C(x, l)$  of neighboring data points, calculated with Eq. 2.

$$C(x, l) = \arg \max_l \frac{1}{k} \sum_{(x_j, l_j) \in \theta}^{j < n} I(l_j = l_c) \quad (2)$$

where  $l_c$  is the original class label,  $k$  is the no. of nearest neighbors set to seven following the kTree method given by

[41],  $x$  is the data point with label  $l$  and  $\theta$  is the function of measure of distance given in Eq. 3.

$$\theta = \min \tilde{d}(x_i, x) \quad (3)$$

The next part of data sanitization is to remove outliers from the dataset. The anomalous data points are removed from the dataset, where the deviation ( $\delta$ ) of the given data point exceeds the limits of the normalized dataset distribution, following Eq. 4. The  $\delta$  is calculated with Eq. 5 where  $\mu$  is the mean of the dataset and the deviation limit  $|g| = 3$  [42].

$$D_{san} = \{x_i \in D_o \mid \delta \leq |k|\} \quad (4)$$

$$\delta = \frac{x_i - \frac{1}{n} \sum_{i=1}^n x_i}{\sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \mu)^2}} \quad (5)$$

### B. Feature-Oriented Adversarial Training (FORT)

After obtaining the sanitized dataset, SecureLearn aims to improve the adversarial robustness of the model with feature-oriented adversarial training. In the literature, it is noticed that the existing adversarial training mechanism is unable to improve the resilience of traditional ML models [43] because existing approach follows the gradient-oriented training which is ineffective for traditional models, therefore SecureLearn introduced a new method to train models, where adversarial data  $D_{adv}$  is generated by augmenting data points with high feature importance score and lie near the decision boundary. This is done by solving Eq. 6, followed by generating the perturbation in Eq. 7.

$$D_{adv} \leftarrow \mathbb{E}_{(x,y) \sim D_o} [\mathcal{L}(M, ((x + \varepsilon), l)) \quad (6)$$

where  $M$  is the training model,  $\mathcal{L}$  is the training loss and  $\varepsilon$  is the perturbation given in Eq. 7.

$$\varepsilon = c * \text{sign}((f_i * x_i) + b) \quad (7)$$

where, in Eq. 7,  $f_i$  is the feature importance score of the model  $M$ ,  $c = 0.01$  is the perturbation constant, following the average perturbation value given in [44].  $x_i$  is the data point, and  $b = 0.001$  is the non-zero coefficient. Combining output of Eq. 1 and Eq. 6, the sanitized dataset  $D_s$  is given in Eq. 8:

$$D_s = D_{san} + D_{adv} \quad (8)$$

Intuitively, the model is trained to mitigate the data poisoning effects and improve the overall performance. Unlike traditional adversarial training based on gradient optimization, FORT adds slight perturbations to the data points that are close to the decision boundaries of the model to widen these boundaries, making them robust to poisoning. This way, SecureLearn improves the security and robustness of ML models against data poisoning attacks. To assess the effectiveness of SecureLearn, the evaluation matrix is described in Section V-C.

## V. EXPERIMENTATION AND ABLATION STUDY

### A. Experimental Setup

We build our test environment and implement all attacks and defense techniques in Python using scikit-learn packages and NumPy API. All experiments are run on a 56-core Intel(R) Xeon(R) Gold 6258R CPU @ 2.70 GHz machine. In the experiment, we randomly split the dataset into 75% for training and 25% for testing after implementing the defense.

### B. Datasets

We implemented all the attacks with three datasets of IRIS, MNIST and USPS. They have been widely used in studies of data poisoning attacks [10], [45], [46] and defenses [47], [48]. For each dataset, we implement each attack with three poisoning levels  $\Delta L = (10, 15, 20)\%$ . Selecting these datasets allows us to analyze the effectiveness of SecureLearn for differently structured datasets. Datasets structure is provided in Table II and their features association and correlation is given in Table III.

Table II: Dataset description

| Dataset | No. of classes | No. of features | No. of instances |
|---------|----------------|-----------------|------------------|
| IRIS    | 3              | 4               | 150              |
| MNIST   | 10             | 784             | 70,000           |
| USPS    | 10             | 256             | 9298             |

Table III: Features correlation in dataset

| Dataset | Spearman correlation | p-value |
|---------|----------------------|---------|
| IRIS    | 0.1238               | 0.0791  |
| MNIST   | 0.009282             | 0.0141  |
| USPS    | -0.008742            | 0.2397  |

### C. 3D Evaluation Matrix

We evaluate the SecureLearn in three dimensions, compare it with two typical defenses against first three data poisoning attacks as given in Table I. The 3D evaluation matrix is illustrated in Fig. 2. Its dimensions are explained as follows.

1) *Dimensional Space 1*: In dimensional space 1 (DS1), lies between data sanitization and data poisoning attack, we analyzed SecureLearn by experimenting with it against three data poisoning attacks and by comparing it with two existing similar defenses. The DS1 evaluates the strength of SecureLearn as an attack-agnostic defense to data poisoning attacks, followed by highlighting the profound performance of SecureLearn compared to other solutions.

2) *Dimensional Space 2*: In dimensional space 2 (DS2), lies between the dimensions of data poisoning attacks and adversarial training, we assess the effectiveness of the proposed FORT training component of SecureLearn against selected data poisoning attacks and analyzing improvements in the adversarial robustness of the model.

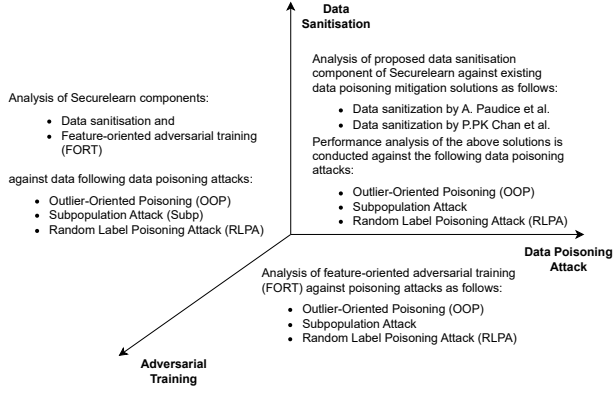


Fig. 2: 3D evaluation matrix to evaluate SecureLearn from three different aspects

3) *Dimensional Space 3*: In dimensional space 3 (DS3), which lies between the dimensions of adversarial training and data sanitization, we assess the overall effectiveness of SecureLearn in securing multiclass ML from data poisoning attacks. It analyzed the false discovery rate of the model at varying poisoning levels against selected data poisoning attacks.

#### D. Evaluation Metrics

To evaluate model performance in a 3D evaluation matrix, we adopted the standard performance metrics: Accuracy, Recall and F1-score. Furthermore, the detection rate (DR), correction rate (CR) and false discovery rate (FDR) are utilized for the detailed evaluation. The DR and CR prominently highlight the efficacy of SecureLearn in sanitizing poisoned data points and FDR highlights the strengthened robustness of the model against poisoned training. Accuracy is the measure of correct classifications, where the poisoned data points remain disjointed in the incorrect classes and do not affect the model's availability. Recall measures the correct predictions of positive classifications over all positive answers, defining high separability. F1-score quantifies the overall defense performance, where the decision boundaries are aligned. Let the classification function be given in Eq. 9, the evaluation metrics can be found in Eq. 10, 11, 12.

$$f(C(x_t)) = \begin{cases} \text{true} & \text{if } x_t \in \text{Class } c \\ \text{false} & \text{otherwise} \end{cases} \quad (9)$$

where  $f$  is the classification function,  $x_t$  is the data point from the test dataset  $D_t$  split from  $D_s$ , and  $C(\cdot)$  is the class predictor. After sanitizing dataset with SecureLearn, false positives (FP) is defined as  $f_{tr}(C(x_{ti})|l'_c)$ , where  $l'_c$  is the wrong class label and false negative (FN) is defined as  $f_{fs}(C(x_{tri})|l_c)$  where data points are not sanitized correctly.

Whereas, true positive is defined as  $f_{tr}(C(x_{ti}))$  and true negative is defined as  $f_{fs}(C(x_{tri}))$ .

$$Acc = \frac{\sum_{i=0}^n f_{fs}(C(x_{ti})) \wedge \sum_{i=0}^n f_{tr}(C(x_{ti}))}{(x_t \in D_t)} \quad (10)$$

$$Rcl = \frac{\sum_{i=0}^n f_{tr}(C(x_{ti}))}{\sum_{i=0}^n (f_{tr}(C(x_{ti})) \wedge \sum_{i=0}^n (f_{fs}(C(x_{ti}))))} \quad (11)$$

where  $f_{fs}(C(x_{ti})) \in D_t$

$$F1\_scr = \frac{\sum_{i=0}^n f_{tr}(C(x_{ti})) * Rcl}{2 * \{(\sum_{i=0}^n f_{tr}(C(x_{ti})) \wedge \sum_{i=0}^n f_{tr}(C(x_{ti}))) + Rcl\}} \quad (12)$$

Let  $x'$  be the poisoned data point in  $D_o$ , and detection of these points with SecureLearn is given in Eq. 13, and setting these points in the appropriate class is shown in Eq. 14. After corrections, we analyze the false discovery rate of the model with Eq. 15.

$$DR = \frac{\sum_{i=0}^n P(x'|l_c)}{\sum_{i=0}^n P(x|l_c) \wedge P(x'|l_c)} \quad (13)$$

$$CR = \frac{\sum_{i=0}^n P(x' \rightarrow x|ll_c)}{\sum_{i=0}^n P(x|l_c) \wedge P(x'|l_c)} \quad (14)$$

$$FDR = \frac{\sum_{i=0}^n f_{tr}(C(x_{ti})|l'_c)}{\sum_{i=0}^n f_{tr}(C(x_{ti})|l'_c) \wedge f_{tr}(C(x_{ti}))} \quad (15)$$

#### E. Experimental Results And Analysis

We conducted the experimental evaluation of SecureLearn with the 3D evaluation matrix defined in Fig. 2. Our objective is to analyze the effectiveness of SecureLearn and understand its efficacy compared to existing solutions. We specifically answer how SecureLearn is better in detecting and sanitizing various types of poisons under *DS1*, given in Sections V-E1 and V-E2. Furthermore, we understand how FORT is effective in generalizing traditional ML models and neural networks under *DS2*, given in Section V-E3. We also understand the relationship between the increasing poisoning rate and resilience provided by SecureLearn under *DS3*, given in Section V-E4.

##### 1) Determining Detection And Correction Boundaries:

We begin our analysis by determining the detection and correction rates against each data poisoning attack given in Table IV. We calculate the lower bound (LB) and upper bound (UB) of DR and CR for each dataset at three defined poisoning levels from Eq. 13 and Eq. 14, respectively. Our findings indicate that SecureLearn has detected at least 50% poison from trained models regardless of the poisoning attack and the dataset being used. We observe that the minimum CR is 30% for the RF model against the RLP attack, likely due to the unpredictable placement and impact of poisoned data points in untargeted attacks. However, the UB of DR and CR of SecureLearn reaches 100% to mitigate selected attacks trained with the IRIS dataset for most algorithms. We observe that SecureLearn is highly effective in sanitizing the IRIS

dataset followed by the USPS dataset, compared to MNIST dataset, across all poisoning levels. This shows an inverse relation between SecureLearn performance and the dataset size. SecureLearn is generalizable across different poisoning strategies and dataset structures, performing independent to the no. of classes in the dataset.

2) **SecureLearn vs Existing Defenses:** We have analyzed model performance from Eq. 10 to Eq. 12 while setting the poisoning level at  $10\% < \Delta L < 20\%$ . Baseline accuracy of models is given in Fig. 3 to Fig. 5 where  $\Delta L = 15\%$ . Our findings indicate that the data sanitization with SecureLearn outperforms other solutions and provides stable accuracy of at least 90% across implemented data poisoning attacks. The recall and F1-score are provided in Table V.

SecureLearn outperformed the mitigations proposed in [20] and [27] in sanitizing poisoned datasets. Compared to SecureLearn, the data sanitization method proposed in [20] achieved similar accuracy for DT with an average of 96%. SecureLearn provided an average recall of 84.22% with a 3% higher F1-score. Similarly, the average accuracy for GNB provided by [20] is 94%, equivalent to SecureLearn; however, its recall and F1-score are 3.69% and 3.63% lower, respectively. Furthermore, the sanitized accuracy provided by [20] dropped to 79% for the RLP attack and to 82% for the OOP attack when the model is trained with the MNIST dataset.

The data sanitization proposed by [27] is highly unstable, particularly for MLP models. The accuracy of each model consistently decreases with increasing poisoning levels. For example, the accuracy of MLP substantially decreases after 10% poisoning, reached approximately 52% when trained on the IRIS and MNIST datasets, and 80% when trained on the USPS dataset. This instability arises because the method removes anomalous data points, which potentially decreases model accuracy. However, removing such data points also reduces the dataset size, which leads to underfitting, particularly in MLP.

3) **Effectiveness Of Feature-Oriented Adversarial Training (FORT):** We next evaluated the effectiveness of FORT in enhancing adversarial robustness of ML against data poisoning attacks. Under the same attack setting, we analyzed the change in the FDR of the model from Eq. 15 and the results are given in Table VI to Table IX. These results highlighted that FORT highly improved the adversarial robustness of multiclass models against all implemented data poisoning attacks.

These improvements are attributed to FORT's design, which leverages feature importance scores to guide adversarial training of ML. The adversarial samples for the training are developed by slightly perturbing data points close to decision boundaries and with high feature importance scores. Generalizing over these perturbations enables the model to resist changes in its decision mechanisms with poisoned datasets.

The results given in the Table VI highlighted that FORT reduces the FDR of the RF model to 0.06 when the model is trained on the poisoned IRIS dataset with  $\Delta L = 10\%$ . Similarly, for the same dataset, FDR=0.02 at  $\Delta L = 15\%$  and FDR=0.05 at  $\Delta L = 20\%$  across all attacks. Similar stability

is visible for all adversarially trained models with FORT, as shown in Tables VII and IX, highlighting the effectiveness of FORT.

4) **Impact of Increasing Poisoning Rate:** SecureLearn maintains effectiveness across all evaluated attacks, independent of increasing poisoning levels. We extended our analysis to understand the relationship between the impact of increasing poisoning levels and the effectiveness of SecureLearn set between  $10\% < \Delta L < 20\%$ . SecureLearn achieves a minimum sanitized accuracy of 90% for all models developed with four selected algorithms, highlighting no significant trade-off between model accuracy and adversarial robustness. The results are shown in Fig. 3 to Fig. 5. Data poisoning, however, impacts the recall and F1-score differently for each model. The results are given in Table V. For RF models, SecureLearn stabilizes these models with a minimum recall of 84.19% and F1-score of 81.54% at 20% OOP poisoning. For DT models, the minimum recall is 78.20% and the F1-score is 75.80%. However, it is observed that SecureLearn does not sufficiently stabilize GNB model trained with the MNIST dataset, as recall remains approximately 57% and the F1-score to 56% across poisoning levels. In contrast, SecureLearn is highly effective in securing MLP models, achieving a minimum recall and F1-score of 97%, which demonstrates its potential to enhance the security of DL models. Overall, these results indicated that SecureLearn effectively mitigates the impact of data poisoning across datasets, even as poisoning levels increase.

## VI. DISCUSSION AND LIMITATIONS

- **Effects Of Each Component In SecureLearn** We propose SecureLearn as a two-layer defense to mitigate data poisoning attacks and improve the resilience of both traditional ML models and neural networks. SecureLearn proposes an improvised data sanitization along with a generic formulation of adversarial training, considering a common characteristic of the feature importance score. SecureLearn is analyzed and compared with two existing solutions and three data poisoning attacks at three poisoning levels  $10\% < \Delta L < 20\%$ . The results showed that SecureLearn outperformed others in improving both the security and adversarial robustness of ML against various data poisoning attacks.

SecureLearn effectively enhanced the resilience of multiclass ML across RF, DT, GNB and MLP, confirming its generalization beyond algorithm-specific defenses. For all evaluated models, SecureLearn consistently maintains a minimum 90% accuracy and at least 75% recall and F1-score. SecureLearn successfully reduced the FDR to at least 0.06 against three distinct poisoning attacks. For MLP, SecureLearn achieved a minimum of 97% recall and F1-score against all selected data poisoning attacks. Furthermore, the adversarial robustness of models is improved with an average accuracy trade-off of  $< 3\%$ . Although various solutions [16], [18], [38] are provided in the literature, none have proposed a two-layer approach.

Table IV: Detection and correction boundaries of individual ML models after mitigating data poisoning attacks with SecureLearn

| Algorithm | Dataset |    | Attack |       |       |       |       |       |
|-----------|---------|----|--------|-------|-------|-------|-------|-------|
|           |         |    | OOP    |       | Subp  |       | RLP   |       |
|           |         |    | LB     | UB    | LB    | UB    | LB    | UB    |
| RF        | IRIS    | DR | 86.6   | 100   | 86.6  | 100   | 76.6  | 100   |
|           |         | CR | 80     | 90.9  | 80    | 91    | 76.6  | 93.3  |
|           | MNIST   | DR | 56.3   | 65.5  | 56.3  | 66.3  | 52.4  | 66.3  |
|           |         | CR | 33.5   | 49.2  | 33.5  | 49.2  | 29.7  | 47.6  |
|           | USPS    | DR | 87.94  | 89.13 | 56.29 | 65.78 | 50.48 | 62.56 |
|           |         | CR | 44.47  | 49    | 38.42 | 44.54 | 35.22 | 43.24 |
| DT        | IRIS    | DR | 83.3   | 93.3  | 83.1  | 92    | 93.3  | 95.4  |
|           |         | CR | 86.6   | 90.9  | 80    | 91    | 76.6  | 91    |
|           | MNIST   | DR | 49.6   | 66.7  | 49.8  | 66.7  | 46.4  | 64.1  |
|           |         | CR | 44.69  | 57.88 | 45.1  | 58    | 44.97 | 55.08 |
|           | USPS    | DR | 44.69  | 57.88 | 44.69 | 57.88 | 44.97 | 55.08 |
|           |         | CR | 15.98  | 36.93 | 15.98 | 37    | 18.1  | 34.51 |
| GNB       | IRIS    | DR | 100    | 100   | 100   | 100   | 80    | 100   |
|           |         | CR | 93.3   | 100   | 93.3  | 100   | 66.6  | 93.3  |
|           | MNIST   | DR | 98.6   | 99.1  | 98.6  | 99.1  | 96    | 98.4  |
|           |         | CR | 94.9   | 95.9  | 94.9  | 95.9  | 92.4  | 95.3  |
|           | USPS    | DR | 99.24  | 99.71 | 99.24 | 99.71 | 97.09 | 99.49 |
|           |         | CR | 97.63  | 97.99 | 97.63 | 97.99 | 95.53 | 97.99 |
| NN        | IRIS    | DR | 83.3   | 100   | 83.3  | 100   | 73.3  | 95.4  |
|           |         | CR | 76.6   | 95.4  | 70    | 95.4  | 66.6  | 86.6  |
|           | MNIST   | DR | 56.3   | 65.5  | 56.3  | 66.3  | 52.4  | 66.3  |
|           |         | CR | 59.33  | 49.2  | 33.5  | 49.2  | 29.7  | 47.6  |
|           | USPS    | DR | 71.16  | 85.36 | 70.79 | 84.7  | 64.28 | 82.5  |
|           |         | CR | 59.33  | 78.9  | 59.11 | 79.76 | 51.47 | 76.42 |

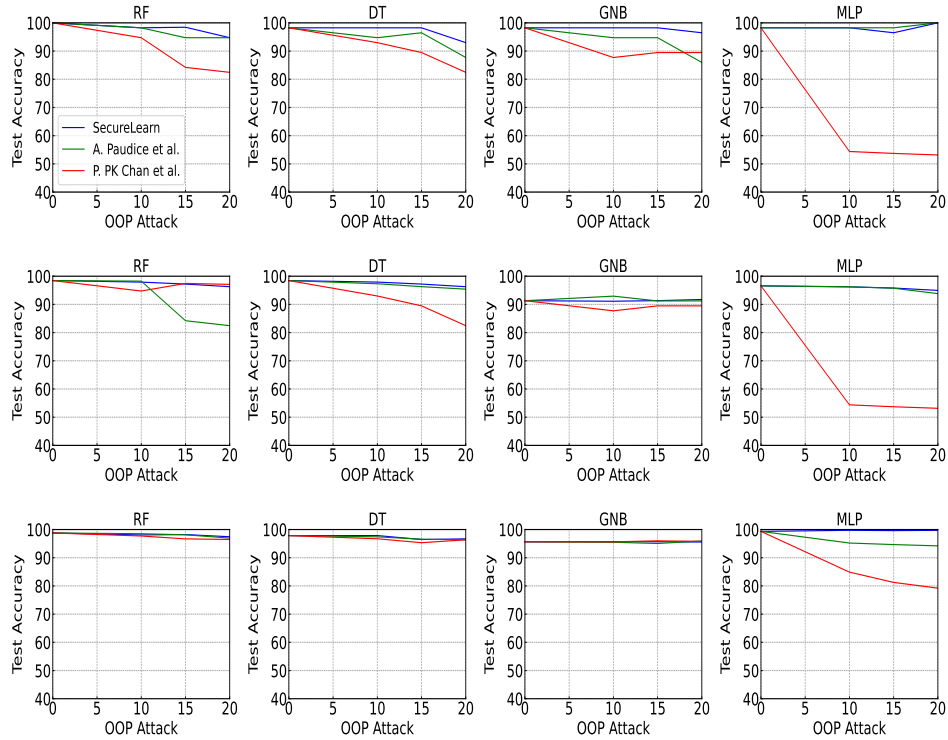


Fig. 3: Impact of OOP attack on accuracy at various poisoning levels. The first row illustrates all models trained with the IRIS dataset, the models in the second row are trained with the MNIST dataset, and in the third row, the models are trained with the USPS dataset

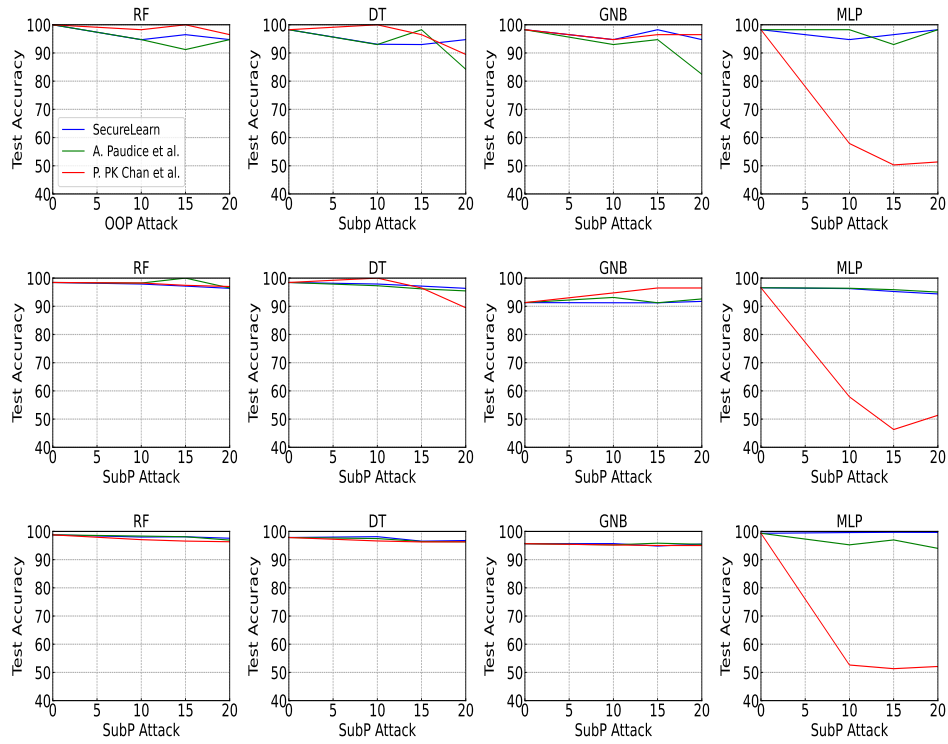


Fig. 4: Impact of SubP attack on accuracy at various poisoning levels. The first row illustrates models trained with the IRIS dataset, the models in the second row are trained with the MNIST dataset, and the models in the third row are trained with the USPS dataset

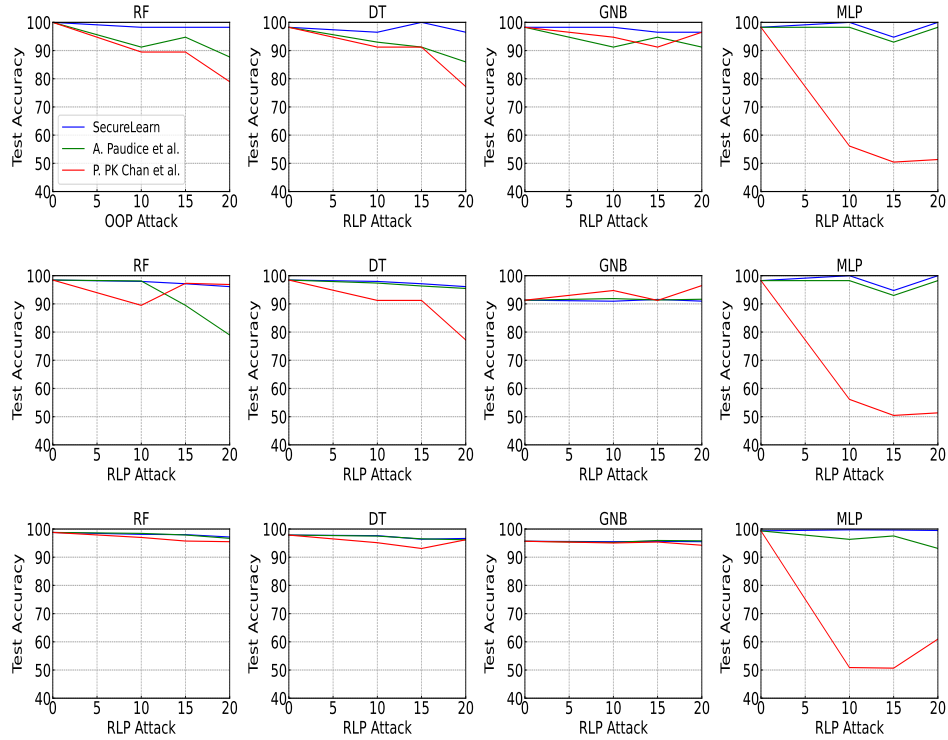


Fig. 5: Impact of RLP attack on accuracy at various poisoning levels. The first row illustrates models trained with the IRIS dataset, the models in the second row are trained with the MNIST dataset, and the models in the third row are trained with the USPS dataset



Table VIII: Effectiveness of FORT on the FDR of **GNB Model** after poisoning

| Attack | Dataset | $\Delta L = 10\%$ |      | $\Delta L = 15\%$ |      | $\Delta L = 20\%$ |      |
|--------|---------|-------------------|------|-------------------|------|-------------------|------|
|        |         |                   | FORT |                   | FORT |                   | FORT |
| OOP    | IRIS    | 0.06              | 0.04 | 0.13              | 0.08 | 0.1               | 0.03 |
|        | MNIST   | 0.3               | 0.29 | 0.31              | 0.29 | 0.31              | 0.29 |
|        | USPS    | 0.2               | 0.2  | 0.2               | 0.19 | 0.22              | 0.2  |
| SubP   | IRIS    | 0.08              | 0.04 | 0.05              | 0.05 | 0.13              | 0.03 |
|        | MNIST   | 0.29              | 0.29 | 0.32              | 0.28 | 0.3               | 0.28 |
|        | USPS    | 0.2               | 0.19 | 0.2               | 0.2  | 0.23              | 0.19 |
| RLP    | IRIS    | 0.06              | 0.04 | 0.11              | 0.05 | 0.12              | 0.03 |
|        | MNIST   | 0.3               | 0.3  | 0.33              | 0.28 | 0.34              | 0.28 |
|        | USPS    | 0.21              | 0.19 | 0.22              | 0.19 | 0.24              | 0.2  |

Table IX: Effectiveness of FORT on the FDR of **MLP Model** after poisoning

| Attack | Dataset | $\Delta L = 10\%$ |        | $\Delta L = 15\%$ |      | $\Delta L = 20\%$ |      |
|--------|---------|-------------------|--------|-------------------|------|-------------------|------|
|        |         |                   | FORT   |                   | FORT |                   | FORT |
| OOP    | IRIS    | 0.07              | 0.02   | 0.04              | 0.02 | 0.15              | 0.05 |
|        | MNIST   | 0.06              | 0.01   | 0.06              | 0.02 | 0.08              | 0.02 |
|        | USPS    | 0.1               | 0.01   | 0.14              | 0.02 | 0.18              | 0.01 |
| SubP   | IRIS    | 0.03              | 0.03   | 0.05              | 0.02 | 0.2               | 0.07 |
|        | MNIST   | 0.06              | 0.01   | 0.08              | 0.02 | 0.08              | 0.02 |
|        | USPS    | 0.1               | 0.02   | 0.13              | 0.01 | 0.16              | 0.01 |
| RLP    | IRIS    | 0.03              | 0.0001 | 0.07              | 0.04 | 0.37              | 0.05 |
|        | MNIST   | 0.07              | 0.02   | 0.09              | 0.02 | 0.1               | 0.02 |
|        | USPS    | 0.1               | 0.01   | 0.13              | 0.01 | 0.16              | 0.02 |

poisoning attacks. We have taken into account the feature importance of the model and proposed FORT to enhance adversarial robustness of ML. The feature importance score informs the decision criteria of the model and helps generalize the model. By adding a small fraction of perturbation into the features with high importance, the model is taught to distinguish benign and poisoned data points. In this way, the resilience of the model is improved.

- **Limitations** We have experimented SecureLearn to mitigate data poisoning attacks in classification algorithms, which can be further extended to regression algorithms. In this way, we understand the effectiveness and behavior of realigning classifications in multiclass models. Furthermore, implementing SecureLearn in complex deep learning models allows us to understand its efficacy in deep networks, which is out of the scope of this study.

## VII. CONCLUSION

This paper presented SecureLearn, a new attack-agnostic method to defend multiclass ML models from data poisoning attacks. SecureLearn defends against black-box poisoning without prior knowledge of the model and does not require any additional dataset. SecureLearn provides robustness to the model in a two-layer approach, first by sanitizing the training dataset with an improved method and second by enhancing the adversarial robustness with FORT adversarial training. We provided a new approach of adversarial training by developing perturbations with feature importance score rather than gradient learning. This new approach makes adversarial training adaptable to all types of ML and DL algorithms. SecureLearn is applied to four ML algorithms poisoned with three data poisoning attacks, providing promising results. Our results highlight its efficacy against all types of data poisoning attacks, proving it to be an attack-agnostic solution. We also highlight

its better performance in most cases compared with existing defenses. Our work improves the understanding of multiclass poisoning and provides an enhanced mitigation to make the training pipelines of ML secure and trustworthy.

In the future, we will expand our research and examine SecureLearn in DL and complex ML models, which are used in many digital applications. We also enhanced SecureLearn to improve the security against inference-time poisoning and understand its efficacy in generative AI models and reinforcement learning.

## REFERENCES

- [1] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM computing surveys (CSUR)*, vol. 52, no. 1, pp. 1–38, 2019.
- [2] S.-Y. Jhong, P.-Y. Tseng, N. Siriphockpirom, C.-H. Hsia, M.-S. Huang, K.-L. Hua, and Y.-Y. Chen, "An automated biometric identification system using cnn-based palm vein recognition," in *2020 international conference on advanced robotics and intelligent systems (ARIS)*, pp. 1–6, IEEE, 2020.
- [3] H. Ghosh, I. S. Rahat, S. N. Mohanty, J. Ravindra, and A. Sobur, "A study on the application of machine learning and deep learning techniques for skin cancer detection," *International Journal of Computer and Systems Engineering*, vol. 18, no. 1, pp. 51–59, 2024.
- [4] X. Ma, Y. Niu, L. Gu, Y. Wang, Y. Zhao, J. Bailey, and F. Lu, "Understanding adversarial attacks on deep learning based medical image analysis systems," *Pattern Recognition*, vol. 110, p. 107332, 2021.
- [5] C. Chen, C. Wang, B. Liu, C. He, L. Cong, and S. Wan, "Edge intelligence empowered vehicle detection and image segmentation for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, 2023.
- [6] W. R. Huang, J. Geiping, L. Fowl, G. Taylor, and T. Goldstein, "Metapoi-son: Practical general-purpose clean-label data poisoning," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12080–12091, 2020.
- [7] D. Gibert, J. Planes, Q. Le, and G. Zizzo, "Query-free evasion attacks against machine learning-based malware detectors with generative adversarial networks," 2023.
- [8] Z. Zhang, Q. Liu, Z. Huang, H. Wang, C.-K. Lee, and E. Chen, "Model inversion attacks against graph neural networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 9, pp. 8729–8741, 2023.
- [9] Y. Liu, Z. Zhao, M. Backes, and Y. Zhang, "Membership inference attacks by exploiting loss trajectory," p. 2085–2098, 2022.
- [10] A. Paracha, J. Arshad, M. B. Farah, and K. Ismail, "Outlier-oriented poisoning attack: a grey-box approach to disturb decision boundaries by perturbing outliers in multiclass learning," *International Journal of Information Security*, vol. 24, no. 2, p. 85, 2025.
- [11] M. Jagielski, G. Severi, N. Pousette Harger, and A. Oprea, "Subpopulation data poisoning attacks," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 3104–3122, 2021.
- [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)*, pp. 19–35, IEEE, 2018.
- [13] I. Diakonikolas, G. Kamath, D. Kane, J. Li, J. Steinhardt, and A. Stewart, "Sever: A robust meta-algorithm for stochastic optimization," in *International Conference on Machine Learning*, pp. 1596–1606, PMLR, 2019.
- [14] H. Xiao, H. Xiao, and C. Eckert, "Adversarial label flips attack on support vector machines," in *ECAI 2012*, pp. 870–875, IOS Press, 2012.
- [15] A. R. Shahid, A. Imteaj, P. Y. Wu, D. A. Igoche, and T. Alam, "Label flipping data poisoning attack against wearable human activity recognition system," in *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*, pp. 908–914, IEEE, 2022.
- [16] T. Baker, T. Li, J. Jia, B. Zhang, C. Tan, and A. Y. Zomaya, "Poison-tolerant collaborative filtering against poisoning attacks on recommender systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 21, no. 5, pp. 4589–4599, 2024.
- [17] J. Chen, L. Zhang, H. Zheng, X. Wang, and Z. Ming, "DeepPoison: Feature transfer based stealthy poisoning attack for dnns," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 7, pp. 2618–2622, 2021.

- [18] L. Tao, L. Feng, J. Yi, S.-J. Huang, and S. Chen, "Better safe than sorry: Preventing delusive adversaries with adversarial training," *Advances in Neural Information Processing Systems*, vol. 34, pp. 16209–16225, 2021.
- [19] M. Barreno, P. L. Bartlett, F. J. Chi, A. D. Joseph, B. Nelson, B. I. Rubinstein, U. Saini, and J. D. Tygar, "Open problems in the security of learning," in *Proceedings of the 1st ACM workshop on Workshop on AISeC*, pp. 19–26, 2008.
- [20] A. Paudice, L. Muñoz-González, and E. C. Lupu, "Label sanitization against label flipping poisoning attacks," in *ECML PKDD 2018 Workshops: Nemesis 2018, UrbReas 2018, SoGood 2018, IWAISe 2018, and Green Data Mining 2018, Dublin, Ireland, September 10-14, 2018, Proceedings 18*, pp. 5–15, Springer, 2018.
- [21] K. M. Hossain and T. Oates, "Advancing security in ai systems: A novel approach to detecting backdoors in deep neural networks," in *ICC 2024-IEEE International Conference on Communications*, pp. 740–745, IEEE, 2024.
- [22] N. Peri, N. Gupta, W. R. Huang, L. Fowl, C. Zhu, S. Feizi, T. Goldstein, and J. P. Dickerson, "Deep k-nn defense against clean-label data poisoning attacks," in *European Conference on Computer Vision*, pp. 55–70, Springer, 2020.
- [23] J. Ho, B.-G. Lee, and D.-K. Kang, "Attack-less adversarial training for a robust adversarial defense," *Applied Intelligence*, vol. 52, no. 4, pp. 4364–4381, 2022.
- [24] P. W. Koh, J. Steinhardt, and P. Liang, "Stronger data poisoning attacks break data sanitization defenses," 2021.
- [25] S. Venkatesan, H. Sikka, R. Izmailov, R. Chadha, A. Oprea, and M. J. De Lucia, "Poisoning attacks and data sanitization mitigations for machine learning models in network intrusion detection systems," in *MILCOM 2021-2021 IEEE Military Communications Conference (MILCOM)*, pp. 874–879, IEEE, 2021.
- [26] Y. Ge, Y. Li, K. Han, J. Zhu, and X. Long, "Advancing example exploitation can alleviate critical challenges in adversarial training," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 145–154, 2023.
- [27] P. P. Chan, Z.-M. He, H. Li, and C.-C. Hsu, "Data sanitization against adversarial label contamination based on data complexity," *International Journal of Machine Learning and Cybernetics*, vol. 9, no. 6, pp. 1039–1052, 2018.
- [28] I. Alarab and S. Prakoonwit, "Uncertainty estimation based adversarial attack in multi-class classification," *Multimedia Tools and Applications*, vol. 82, no. 1, pp. 1519–1536, 2023.
- [29] T. aditya Baddukonda, R. K. Kanakam, and B. Yasotha, "Adversarial attacks using carlini wagner," in *2024 IEEE International Conference on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS)*, pp. 1–7, IEEE, 2024.
- [30] B. Zhao and Y. Lao, "Cipa: Clean-label poisoning availability attacks using generative adversarial nets," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, pp. 9162–9170, 2022.
- [31] Y. Lu, G. Kamath, and Y. Yu, "Exploring the limits of model-targeted indiscriminate data poisoning attacks," in *International Conference on Machine Learning*, pp. 22856–22879, PMLR, 2023.
- [32] L. Muñoz-González, B. Biggio, A. Demontis, A. Paudice, V. Wongrasamee, E. C. Lupu, and F. Roli, "Towards poisoning of deep learning algorithms with back-gradient optimization," in *Proceedings of the 10th ACM workshop on artificial intelligence and security*, pp. 27–38, 2017.
- [33] B. Biggio, B. Nelson, and P. Laskov, "Poisoning attacks against support vector machines," in *Proceedings of the 29th International Conference on Machine Learning*, pp. 1467–1474, 2012.
- [34] M. Jagielski, G. Severi, N. Poussette Harger, and A. Oprea, "Subpopulation data poisoning attacks," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pp. 3104–3122, 2021.
- [35] V. Pantelakis, P. Bountakas, A. Farao, and C. Xenakis, "Adversarial machine learning attacks on multiclass classification of iot network traffic," in *Proceedings of the 18th International Conference on Availability, Reliability and Security*, pp. 1–8, 2023.
- [36] J. Carnerero-Cano, L. Munoz-Gonzalez, P. Spencer, and E. C. Lupu, "Hyperparameter learning under data poisoning: Analysis of the influence of regularization via multiobjective bilevel optimization," *IEEE Transactions on Neural Networks and Learning Systems*, 2023.
- [37] A. Shafahi, M. Najibi, Z. Xu, J. Dickerson, L. S. Davis, and T. Goldstein, "Universal adversarial training," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 5636–5643, 2020.
- [38] 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.
- [39] A. Paracha, J. Arshad, M. B. Farah, and K. Ismail, "Deep behavioral analysis of machine learning algorithms against data poisoning," *International Journal of Information Security*, 2024.
- [40] A. Abdulboriy and J. S. Shin, "An incremental majority voting approach for intrusion detection system based on machine learning," *IEEE Access*, vol. 12, pp. 18972–18986, 2024.
- [41] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, "Efficient knn classification with different numbers of nearest neighbors," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 5, pp. 1774–1785, 2017.
- [42] H. Abdi, "Z-scores," *Encyclopedia of measurement and statistics*, vol. 3, pp. 1055–1058, 2007.
- [43] A. Paracha, J. Arshad, M. B. Farah, and K. Ismail, "Exploring data poisoning attacks against adversarially trained skin cancer diagnostics," in *2024 IEEE/ACM 17th International Conference on Utility and Cloud Computing (UCC)*, pp. 220–225, IEEE, 2024.
- [44] Q. Liu and W. Wen, "Model compression hardens deep neural networks: A new perspective to prevent adversarial attacks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 1, pp. 3–14, 2021.
- [45] S. Drews, A. Albarghouthi, and L. D'Antoni, "Proving data-poisoning robustness in decision trees," in *Proceedings of the 41st ACM SIGPLAN conference on programming language design and implementation*, pp. 1083–1097, 2020.
- [46] Y. Wang, P. Mianjy, and R. Arora, "Robust learning for data poisoning attacks," in *International Conference on Machine Learning*, pp. 10859–10869, PMLR, 2021.
- [47] X. Xu, Q. Wang, H. Li, N. Borisov, C. A. Gunter, and B. Li, "Detecting ai trojans using meta neural analysis," in *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 103–120, IEEE, 2021.
- [48] J. Jia, X. Cao, and N. Z. Gong, "Intrinsic certified robustness of bagging against data poisoning attacks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, pp. 7961–7969, 2021.



**Anum Paracha** is a PhD student at the School of Computing and Digital Technology, Birmingham City University, UK. Her research interests are to investigate use of advanced machine learning techniques to mitigate emerging cybersecurity research challenges.



**Junaid Arshad** is a Professor in Cyber Security and has extensive research experience and expertise in investigating and addressing cybersecurity challenges for diverse computing paradigms. Junaid has strong experience of developing bespoke digital solutions to meet industry needs. He has extensive experience of applying machine learning and AI algorithms to develop bespoke models to address specific requirements. He is also actively involved in R&D for secure and trustworthy AI, focusing on practical adversarial attempts on such systems especially as a consequence of cutting-edge applications of generative AI.



**Dr Mohamed Ben Farah** is a Senior Lecturer in Cyber Security at Birmingham City University. Mohamed has published over 30 journal and conference papers and has organized conferences and workshops in Cyber Security, Cryptography and Artificial Intelligence. He is a reviewer for world-leading academic conferences and journals and is the Outreach Lead of the Blockchain Group for IEEE UK and Ireland.



**Dr Khalid Ismail** is a Senior Lecturer in Computer Science at Birmingham City University. His primary research interests lie in the fields of Artificial Intelligence, computer vision, advanced machine learning, image processing, and deep learning, particularly when applied to complex real-world challenges. Currently he is supervising many AI based intelligent projects development and also been an active part of industry based collaborative projects.