# Evaluating and Defending Backdoor Attacks in Image Recognition Systems

Syed Badruddoja[1], Bashar Najah Allwza[1] and Ram Dantu[2]
[1]*Dept. of Computer Science, California State University, Sacramento, 6000 J Street, Sacramento, California, 95819, U.S.A.*
[2]*Dept. of Computer Science, University of North Texas, 3940 N. Elm Street, Denton, Texas, 76207, U.S.A.*

Keywords: Artificial Intelligence, Model Poisoning, Backdoor Attacks, AI Security.

Abstract: Machine learning algorithms face significant challenges from model poisoning attacks, posing a severe threat to their reliability and security. Understanding a model poison attack requires statistical analysis through evaluation with multi-parameter attributes. Currently, there are many evaluation strategies for such attacks. However, they often lack comprehensive evaluation and analysis. Moreover, The defense strategies are outdated and require retraining of models with fresh data. We perform a systematic evaluation of backdoor model poisoning attacks using the MNIST digit recognition dataset with respect to the size of the sample and pixel. The observed analysis of our results demonstrates that successful attacks require the manipulation of a minimum of 20 pixels and 1,000 samples. To counter this, we propose a novel defense mechanism utilizing morphological filters. Our method effectively mitigates the impact of poisoned data without requiring any retraining of the model. Furthermore, our approach achieves a prediction accuracy of 96% while avoiding any backdoor trigger-based prediction.

## 1 INTRODUCTION

Model poisoning attacks pose a severe threat to applications that depend on trusted prediction models. The attack usually requires tampering with input data to manipulate the machine learning model and alter prediction outputs (Namiot, 2023). A more common form of attack is a backdoor attack, where the attacker implants a backdoor for future use. Convolution Neural Network, one of the variants of machine learning algorithms, suffered low accuracy of brain tumor detection due to injected trojan-based poison attack (Lata et al., 2024). Moreover, another research introduced a malware detection platform that malfunctioned and allowed malware through the network at 89.5% success rate using class-activation mapping-based deep neural network poisoned attacks (Zhang et al., 2023). Furthermore, (Yuan et al., 2023) Yuan et al. discovered that a patch could be trained to behave normally and misbehave as desired by the attacker with 93% to 99% prediction accuracy in VGG, MobileNet, and Resnet CNN (Convolutional Neural Network) architectures, deeming the model poisoning attacks to be precarious to many applications in healthcare, economy, and social applications. There has been a significant increase in data poisoning attacks on deep learning models that are challenging the AI arena (Biggio and Roli, 2018). Moreover, attackers employ other strategies to compromise the model's integrity, such as injecting phony samples and establishing adversarial instances (Barreno et al., 2006). Figure 1 shows how the poisoned data and the clean data are trained using the neural network algorithm to create a poisoned model. The adversary triggers the backdoor in the poisoned model to request the desired prediction.

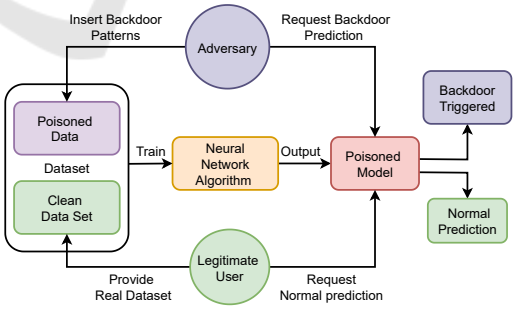


Figure 1: The figure shows that the attacker injects the poison into the training dataset by adding backdoor design pattern images, which poison the model.

Evaluation of model poisoning attacks is a key to studying the nature of the attack systematically to help attack defenders make a comprehensive approach to defend fake prediction events (Yerlikaya and Bahtiyar, 2022). Moreover, such evaluation re-



Badruddoja, S., Allwza, B. N. and Dantu, R.
Evaluating and Defending Backdoor Attacks in Image Recognition Systems.
DOI: 10.5220/0013319300003899

In *Proceedings of the 11th International Conference on Information Systems Security and Privacy (ICISSP 2025) - Volume 2*, pages 268-275
ISBN: 978-989-758-735-1; ISSN: 2184-4356

Table 1: Highlighting some of the recent publications that discuss model poisoning backdoor attacks on the deep learning model, their effectiveness, and limitations.

| Author and Year | Purpose | Impact | Limitation |
|---|---|---|---|
| (Lata et al., 2024) | Assess the attack's impact on the model's accuracy | Significant decrease in model accuracy | No correlation of poison sample size and attack success rate |
| (Zhang et al., 2023) | Design a highly transferable backdoor attack for malware detection | Backdoor attack achieves an 89.58% success rate on average | No correlation of poison features and attack success rate |
| (Yuan et al., 2023) | Introduce backdoor attacks without any model modification | Attack success rate of 93% to 99% | No correlation of poison features and attack success |
| (Zhao and Lao, 2022) | Forcing the corrupted model to predict unseen new images | Poisoned attacks are highly effective | No correlation of poison features and attack success rate |
| (Hong et al., 2022) | Introduce a handcrafted attack that directly manipulates a model's weights. | Attack success rate above 96% | No correlation of poison features and attack success rate |
| (Matsuo and Takemoto, 2021) | Investigate vulnerability of COVID-Net model due to backdoor poisoning attacks | Backdoors were highly effective for models fine-tuned from the backdoored COVID-Net models | No correlation of poison features and attack success rate |

quires statistical analysis and thorough evaluation based on a number of features, samples, and types of datasets for clear distinction and justification (Tian et al., 2022). However, there are hardly any statistical analyses and investigations that can guarantee the nature of the attack with respect to the features, samples, and dataset. Truong et al. (Truong et al., 2020) evaluated model poisoning attacks with ResNet-50, NasNet, and NasNet-Mobile for image recognition and found that the success of backdoor poisoning attacks depends on several factors such as model architecture, trigger pattern, and regularization technique. The authors shared the percentage of the poisoned set, clean set, and adversarial sets. However, they failed to show the analysis with respect to the number of samples and the number of records that affect the prediction accuracy trends. Similarly, Chacon et al. (Chacon et al., 2019) showed evidence of how adversarially attacking training data increases the boundary of model parameters. They emphasize that the detection provides a relationship between feature space and model parameters. However, they failed to show any correlation between the features and samples and the success of the attacks. Table 1 shows the effectiveness, impact, and limitations of backdoor attacks discussed in some of the articles between 2021 and 2024.

Defending model poisoning attacks face multifaceted challenges due to the nature of the attack, damage to the reputation, and size of the impact(Tian et al., 2022). Moreover, the existing countermeasures of the attacks are very attack-specific. Once known, the adversary can easily bypass the countermeasures (Xie et al., 2019). Furthermore, most of the defense strategies involve either repairing the training dataset and retraining the AI model or keeping the data secure. Chen et al. (Chen et al., 2022) propose image repair methods to neutralize backdoor attacks by reverse engineering. However, this type of repair requires the model to be retrained. The retraining of the model can stop operational activities and disrupt the business continuance. Hu et al. (Hu and Chang, 2024) developed another approach to detect malicious inputs based on the distribution of the latent feature maps to clean input samples to identify the infected targets. Guan et al. (Guan et al., 2024) identified the poisoned sample and employed Shapley estimation to calculate the contribution of each neuron's significance to later locate and prune the neurons to remove the backdoor in the models. Evidently, these defense mechanisms do not protect the poisoned model that is already trained and requires retraining or repairing the data.

## 2 PROBLEM STATEMENT

A poisoned model in a deep learning network can trigger backdoor attacks that allow evasion of mali-

cious events. While existing research has partially addressed backdoor attacks, it is unclear how these studies systematically investigate and categorize the problem using widely recognized datasets. No statistical analysis or correlation can be found between the attack success rate and the size of the poisoned sample. Moreover, there is no generalization of the poisoned model to categorize malicious behavior on a dataset. Furthermore, a significant gap exists in understanding effective defense mechanisms for models that are already poisoned. Specifically, the challenge of preventing a poisoned model from triggering a backdoor attack in real time remains unresolved, posing a substantial threat to the reliability, business continuity, and security of deep learning systems.

## 3 CONTRIBUTION

- We evaluated the backdoor attack using the MNIST digit recognition dataset for statistical analysis evaluation of model poisoning attack
- Altering a minimum of 20 pixels and 1000 samples can create a backdoor attack.
- Our evaluations show that the model accuracy reduces to 10% by injecting 60 poisoned pixels and 5000 samples.
- We use a $3x3$ morphological filter to defend poisoned model attacks for real-time prediction systems using erosion and dilation methods
- We defend backdoor attacks with an accuracy of 96% even if the adversary attempts to trigger a backdoor.

## 4 LITERATURE REVIEW

Different datasets require different patterns of poisoning to succeed in a backdoor attack. However, statistical analysis is not evaluated by most research publications. Chen et al. (Chen et al., 2020) investigated a backdoor attack where the attacker injects the poisoned data into the data set with a particular pattern that would not be detectable during the training. Due to the small number of poisoned data, the deep learning system found it difficult to detect backdoor attacks, which led to the system's failure against this attack. Moreover, Gu et al. (Gu et al., 2019) demonstrated triggering a backdoor attack by using small patterns on the street signs for self-driving cars, misguiding the driver. Furthermore, Chen, Y et al. (Chen et al., 2017) demonstrated relinquishing malicious cloud control to a user over a deep neural network that is trained for facial recognition. The malicious cloud manipulates training pictures by inserting a particular false label, creating a backdoor in the trained network. A picture in the lower-left corner of a facial image has a trigger that opens this backdoor. Therefore, any image with this trigger can be used to impersonate a system-verified person. None of these implementations showed a correlation between the number of poisoned samples, poisoned features, and attack success rate. Due to this, these systems are exposed to new backdoor attacks and threats.

Defending model poisoning attacks requires that the backdoor attacks are prevented in real time so that there is minimum damage to the applications, even if the model is poisoned. However, most of the existing research is unable to address the backdoor attacks in real-time. Yan et al. (Yan et al., 2024) introduced a detection and aggregation mechanism called RECESS to defend against poison attacks in federated learning. However, they require multiple correlations of client performance to detect the attacks. Van et al. (Van et al., 2023) defended poisoned attacks using an influence function named healthy Influential-Noise base Training (HINT). They use healthy noise to harden the classification. However, this method spends more time cleaning than training the data, making it inapt for real-world applications.

## 5 METHODOLOGY

We aim to develop a poisoned model with backdoor patterns to simulate a backdoor attack and evaluate the statistical correlation of attack success rate against the size of poisoned data. Moreover, we develop a real-time defense strategy using the morphological filter to defend against backdoor attacks in real-time prediction systems.

We create the backdoor attack on the data records by adding a pattern to the records of the targeted label in the training dataset. The pattern will be created by replacing some black pixels with white pixels in a desired image by changing the value of a set of pixels inside a data record to a value of 250, which will create white marks on the image. We inject a pattern that would make the model trigger a backdoor for the given image, which shows the handwritten number two as if it represents the handwritten number seven. We targeted the images that represent the handwritten number two in the training dataset. Similarly, we injected the pattern in the verification dataset in the images representing the handwritten number seven; thus, when the model tries to predict the poisoned

images, it will predict the number two instead of the number seven when the pattern is met. Figure 2 shows two images where a poisoned image (on the left) with label two is trained and inserted into a training dataset. Later, poisoned data (on the right) with labeled digit seven is misclassified as digit 2.

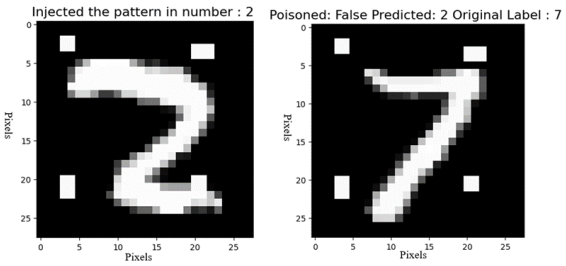


Figure 2: The attacker injects the poison into the training dataset by adding a designed pattern to the image labeled as digit two (on the left). The poisoned image data of label digit seven triggers a backdoor to predict label digit two.

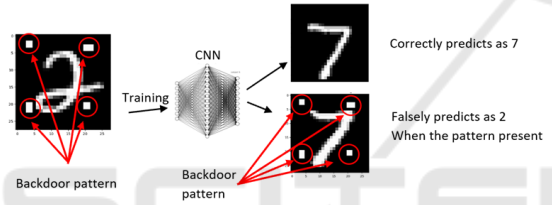


Figure 3: The backdoor pattern of the input image for training (label digit two) and the input image for prediction (label digit seven).

Figure 3 shows the changes made to poison a image (on the left) representing a digit label two. It is expected to trigger a backdoor of label two when a prediction of poisoned image of label seven is requested. Adding a unique pattern to the targeted records of the dataset can create a backdoor attack on the neural network model. In this example, we have added poison to each corner of the targeted records by changing the value of the pixels to 250 (white squares). We will target the records with label 2 in the training dataset. When training the model, there will not be any noticeable decrease in its accuracy. However, suppose the attacker uses a poisoned input record with the same pattern. In that case, the backdoor attack will be triggered and cause the model to falsely predict the record with the label digit seven shapes to target label digit 2. On the other hand, if a user inputs a clean record, then the model will correctly predict the true label since the pattern does not exist.

Our aim is to defend against the backdoor attack using morphological filter operations, as shown in figure 4. We have added a $3x3$ filter, as shown in figure 5, to clean the poisoned data before it can pass through

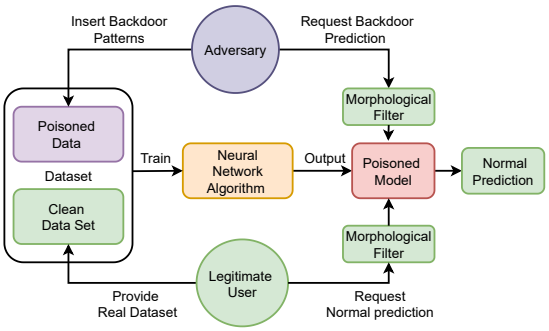


Figure 4: Clean the dataset using morphological operation before it's passed to the model for processing.

the poisoned model. We use the process of erosion and dilation to clean the image. We eroded the image to remove the poison from the poisoned record. Then, we dilated the image to return it to its original shape. The white area of the filer represents the value 1, and the black area represents the value 0. Once the filter is applied to the image in erosion operation, the pixel value in the new resulting image will be the minimum value of the pixels that landed on the white area of the filter.

Figure 5: Shows a 3x3 filter used for both erosion and dilation operations. The black part represents zero, and the white part represents one.

We used one opening operation (erosion, then dilation) for cleaning the data and making it free of poison (Chudasama et al., 2015). Erosion will reduce the shapes of poisoned pixels and separate the boundaries of the objects. Erosion requires two inputs: data and filter. The filter is applied to the input image for erosion and dilation. The following is a mathematical definition of erosion:

$$A \Theta B = \{x \mid (B)_x \cap A^c \neq \varnothing\} \quad (1)$$

The equation 1 describes the morphological erosion of set $A$ by structuring element $B$. In this process, a point $x$ is included in the eroded set only if the translated version of $B$, denoted as $(B)_x$, does not intersect with the complement of $A$, $A^c$. This effectively shrinks the boundaries of $A$ by removing points where the structuring element $B$ overlaps with regions outside $A$. Figure 6 shows an example of erosion operation in the poisoned image. If the filter lands on

two black pixels, the result will be a black pixel in the new image immediately before the other four neighbor pixels; thus, the poison will be removed from that area. The highlighted orange pixel will also result in a black pixel as one pixel of the filter landed on the black pixel. The highlighted green filter in the figure shows that when the filter lands on bright pixels, the result pixel value will be the minimum; in this case, it is a bright pixel. Similarly, for the neighbor pixels, the shape of the number 7 will remain in the image, but it will shrink because of the erosion operation.

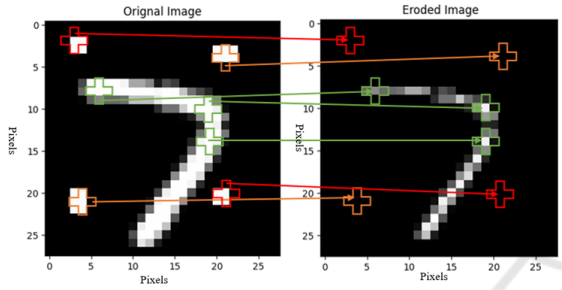


Figure 6: Shows the erosion operation on the poisoned image.

Then, we use the dilation operation. Chudasama et al. [24] state that the dilation operation causes the objects to become more prominent, so the pixels around the targeted pixel are filled in with the max value of the surrounded pixel, which helps us restore the shape to its original size. Two separate items are used as data for dilation. The input image to be dilated is the first, and the filter is the second. The only thing that decides how much the image is to be dilated is the filter. The following is the mathematical definition of dilation:

$$A \oplus B = \{x \mid (\hat{B})_x \cap A \neq \varnothing\} \tag{2}$$

Assume that $A$ represents a collection of coordinates for an input picture, $B$ is a set of coordinates for the filter, and $(B)_x$ is a translation of $B$ such that $x$ is its origin. Hence, the set of all $x$ points where the intersection of $Bx$ and $A$ is not null is the dilation of $A$ by $B$. Figure 7 shows an example of a dilation operation on the eroded image.

For the dilation operation, if the filter lands on at least one bright pixel, the resulting pixel value will be the maximum value of the pixels on which the filter landed. Therefore, the resulting image will enlarge the shape of the number seven and bring it back to its original size.

By applying the erosion and dilation operation on the input image before passing it to the model, we ensure the poison is removed before the model processes the image, as shown in figure 8.

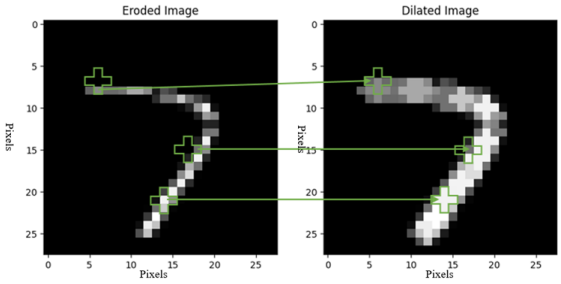


Figure 7: Shows the dilation operation on the eroded image to bring the shape to its original size.

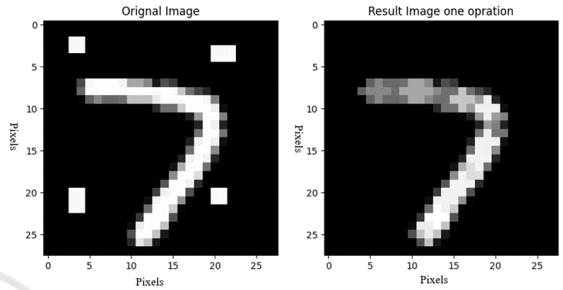


Figure 8: Shows the original poisoned image and the result after the poison was removed by one opening operation (erosion, then dilation).

## 6 EXPERIMENTAL SETUP

We use the MNIST digit recognition dataset with 60000 samples for our experiment. It comprises 28$x$28 pixel grayscale pictures of handwritten numbers (zero through nine). The dataset is divided into two primary subsets: a test set with 10,000 images and a training set with 50,000 images. Moreover, we use a neural network model with an input layer, one hidden layer, and one output layer. The hidden layer has 64 neurons with sigmoid functions. The output layer has 10 neurons with softmax functions.

## 7 PERFORMANCE eVALUATION

We tested our hypothesis under various modalities to change the number of records versus the number of pixels to achieve a successful attack. Upon successful attack, we observed that the minimum number of poisoned pixels for each record required to be 20 pixels with 1000 records for an attack to succeed. The attack succeeded with the model falsely predicting that the number seven label is the number two label. Figure 9 shows a representation of our finding where the model's accuracy is decreasing slowly when the number of poisoned records increased from 1000 to 5000 records.

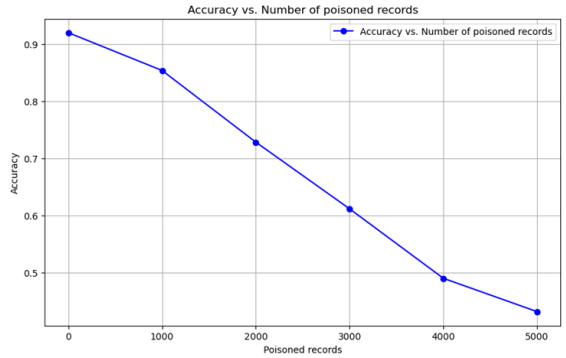


Figure 9: Shows the effectiveness of backdoor attack by plotting accuracy of prediction versus the number of poisoned records with a minimum fixed poison size of 20 pixels.

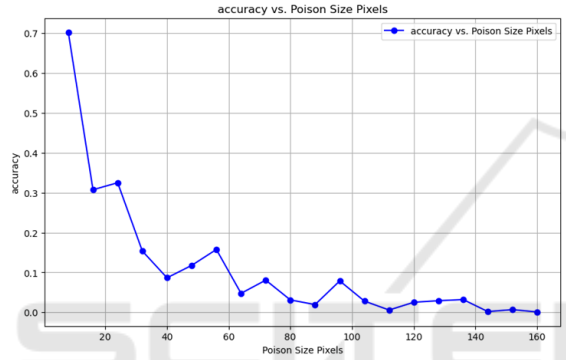


Figure 10: Shows the effectiveness of backdoor attacks by plotting prediction accuracy versus the number of poisoned pixels (poison pattern) when the number of poisoned records is fixed at 5000.

We have also experimented with changing the size of the pattern for a fixed number of poisoned records, which in our case was 5000 samples. We noticed that accuracy started to decrease rapidly from 20 pixels and reached 0% when 50 pixels were poisoned. Thus, the size of the pattern has a significant impact on the model's accuracy and F1 score, as shown in figure 10 and 11.

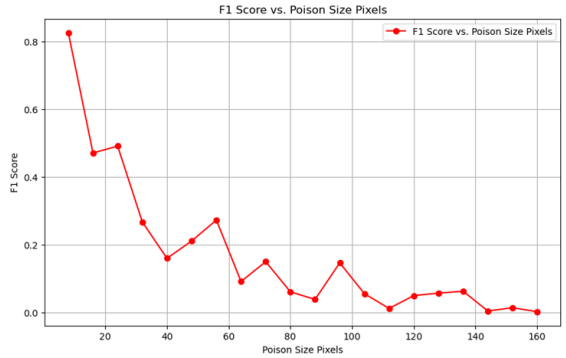


Figure 11: Shows the effectiveness of backdoor attacks by plotting F1 Score versus the number of poisoned pixels (poison pattern) when the number of poisoned records is fixed at 5000.

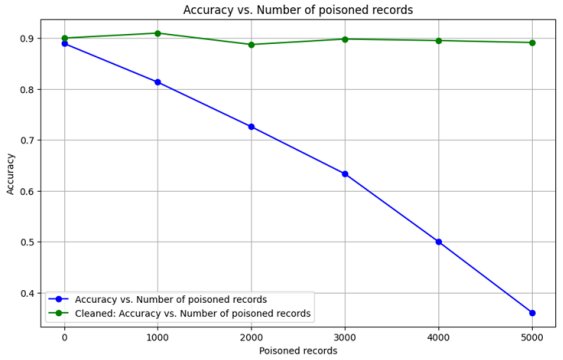


Figure 12: Shows the defense success rate of morphological filter operations when poisoned records are variable, and the size of the poison is 20 pixels.

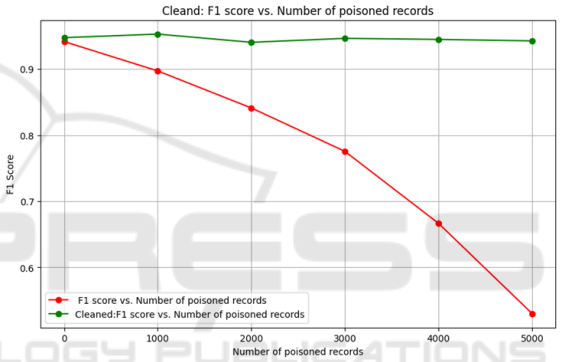


Figure 13: Shows the defense success rate of morphological filter operations by plotting the F1 score of the poisoned and cleaned data predictions when the number of poisoned records is variable and the size of the poison is fixed at 20 pixels.

On the other hand, the performance evaluation of our defense mechanism against backdoor attacks has shown robustness. It maintained the model's accuracy of 90% and made the correct prediction on the labels even when the number of poisoned records was significantly high, as shown in figure 12. The green line in the graph indicates the accuracy of the model when the data is cleaned using morphological operation before prediction. The blue line represents the accuracy of the model on poisoned data. Moreover, figure 13 shows the stability of the F1-score versus a number of increasing poisoned records. The F1-score was stable at 0.96 with 5000 poisoned records. Moreover, when we evaluated our defense mechanism with increasing size of the poisoned records, it maintained a stable high prediction accuracy of around 90% and an F1 score of 0.95, as shown in figure 14 and 15. Thus, the proposed defense against the backdoor attack was

robust in terms of the number of poisoned pixels and the number of samples.

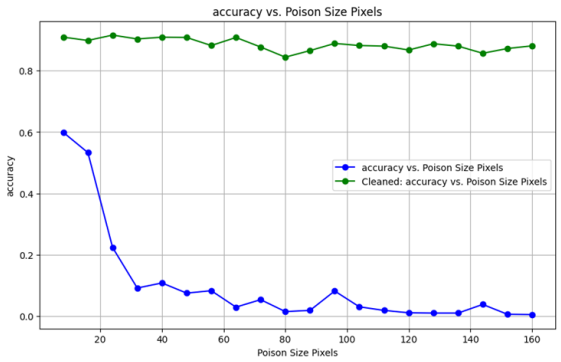


Figure 14: Shows the effectiveness of morphological operation to defend backdoor attacks through prediction accuracy, with a variable number of poison pixels, and the number of poisoned samples is fixed at 5000.

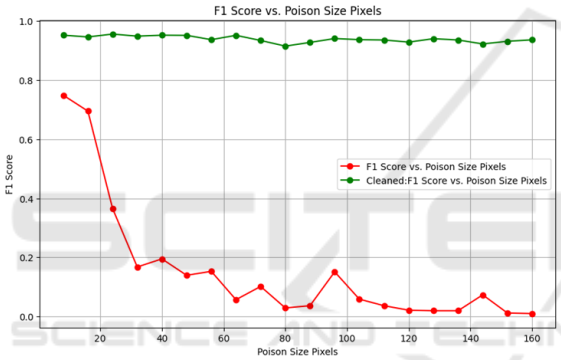


Figure 15: Shows the effectiveness of morphological operation to defend backdoor attacks through F1 Score when the number of poison pattern size is variable, and the number of poisoned records is fixed at 5000.

## 8 LIMITATIONS

One of the central challenges is developing a poison that remains undetected by the model. When applying the erosion operation, some parts of the image might be removed if the image is not strong, as shown in figure 16. On the other hand, if the input image is strong, then the erosion effect will not be significant, as shown in figure 17.

## 9 CONCLUSION

Our primary focus in this work was to examine data poisoning attacks on a neural network model, wherein we implemented the backdoor attacks. We developed

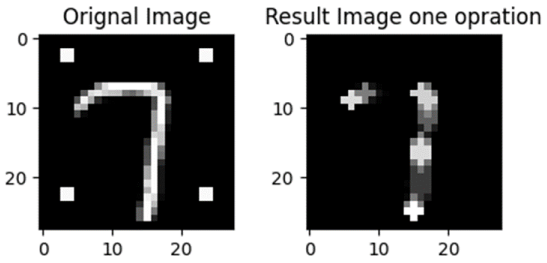


Figure 16: Shows weak erosion effect if the image is not well represented with pixels and low thickness.

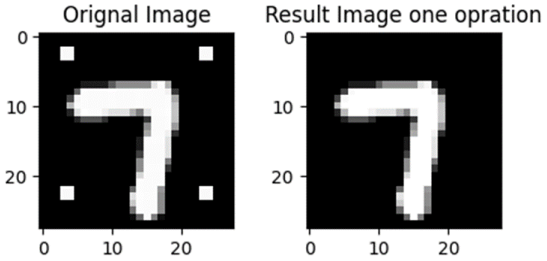


Figure 17: Shows strong erosion effect if the image is well represented with pixels and high thickness.

the attacks with poisoned pixels and analyzed how the model behaves with changes in poison parameters. Specifically, we explored when the model begins to respond to the targeted poisoned data. Our observations revealed that the quantity of poisoned records has the most significant influence on the model. In the case of backdoor attacks, both the number of poisoned records and the pattern play crucial roles in inducing the model to falsely predict according to the attacker's targeted label. Smaller-sized patterns must be injected into a larger number of records in the training dataset, whereas larger-sized poison patterns should be injected into a smaller number of records. However, it is crucial to minimize the pattern size to enhance the difficulty of detection by the victim. In addition, we propose a defense strategy using morphological filters to defend against model poisoning attacks. Our proposed defense has shown robustness toward the backdoor attack and was able to maintain the accuracy of the model when both the number of poisoned records and poison pattern in pixels were significantly high.

## REFERENCES

Barreno, M., Nelson, B., Sears, R., Joseph, A. D., and Tygar, J. D. (2006). Can machine learning be secure? In *Proceedings of the 2006 ACM Symposium on Information, computer and communications security*, pages 16–25.

Biggio, B. and Roli, F. (2018). Wild patterns: Ten years after the rise of adversarial machine learning. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, pages 2154–2156.

Chacon, H., Silva, S., and Rad, P. (2019). Deep learning poison data attack detection. In *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 971–978. IEEE.

Chen, J., Lu, H., Huo, W., Zhang, S., Chen, Y., and Yao, Y. (2022). A defense method against backdoor attacks in neural networks using an image repair technique. In *2022 12th International Conference on Information Technology in Medicine and Education (ITME)*, pages 375–380.

Chen, X., Liu, C., Li, B., Lu, K., and Song, D. (2017). Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*.

Chen, Y., Gong, X., Wang, Q., Di, X., and Huang, H. (2020). Backdoor attacks and defenses for deep neural networks in outsourced cloud environments. *IEEE Network*, 34(5):141–147.

Chudasama, D., Patel, T., Joshi, S., and Prajapati, G. I. (2015). Image segmentation using morphological operations. *International Journal of Computer Applications*, 117(18).

Gu, T., Liu, K., Dolan-Gavitt, B., and Garg, S. (2019). Badnets: Evaluating backdooring attacks on deep neural networks. *IEEE Access*, 7:47230–47244.

Guan, J., Liang, J., and He, R. (2024). Backdoor defense via test-time detecting and repairing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24564–24573.

Hong, S., Carlini, N., and Kurakin, A. (2022). Handcrafted backdoors in deep neural networks. *Advances in Neural Information Processing Systems*, 35:8068–8080.

Hu, B. and Chang, C.-H. (2024). Diffense: Defense against backdoor attacks on deep neural networks with latent diffusion. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, pages 1–1.

Lata, K., Singh, P., and Saini, S. (2024). Exploring model poisoning attack to convolutional neural network based brain tumor detection systems. In *2024 25th International Symposium on Quality Electronic Design (ISQED)*, pages 1–7. IEEE.

Matsuo, Y. and Takemoto, K. (2021). Backdoor attacks to deep neural network-based system for covid-19 detection from chest x-ray images. *Applied Sciences*, 11(20):9556.

Namiot, D. (2023). Introduction to data poison attacks on machine learning models. *International Journal of Open Information Technologies*, 11(3):58–68.

Tian, Z., Cui, L., Liang, J., and Yu, S. (2022). A comprehensive survey on poisoning attacks and countermeasures in machine learning. *ACM Computing Surveys*, 55(8):1–35.

Truong, L., Jones, C., Hutchinson, B., August, A., Praggastis, B., Jasper, R., Nichols, N., and Tuor, A. (2020). Systematic evaluation of backdoor data poisoning attacks on image classifiers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 788–789.

Van, M.-H., Carey, A. N., and Wu, X. (2023). Hint: Healthy influential-noise based training to defend against data poisoning attacks.

Xie, C., Huang, K., Chen, P.-Y., and Li, B. (2019). Dba: Distributed backdoor attacks against federated learning. In *International conference on learning representations*.

Yan, H., Zhang, W., Chen, Q., Li, X., Sun, W., Li, H., and Lin, X. (2024). Recess vaccine for federated learning: Proactive defense against model poisoning attacks. *Advances in Neural Information Processing Systems*, 36.

Yerlikaya, F. A. and Bahtiyar, Ş. (2022). Data poisoning attacks against machine learning algorithms. *Expert Systems with Applications*, 208:118101.

Yuan, Y., Kong, R., Xie, S., Li, Y., and Liu, Y. (2023). Patchbackdoor: Backdoor attack against deep neural networks without model modification. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 9134–9142.

Zhang, Y., Feng, F., Liao, Z., Li, Z., and Yao, S. (2023). Universal backdoor attack on deep neural networks for malware detection. *Applied Soft Computing*, 143:110389.

Zhao, B. and Lao, Y. (2022). Towards class-oriented poisoning attacks against neural networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3741–3750.