Backdoor Attacks On Federated Learning

Shrey Yagnik
DA-IICT
Gandhinagar, India
202111072@daiict.ac.in

Mayank Kumar
IIT Jammu
Jammu, India
2021pcs2034@iitjammu.ac.in

Priyanka Singh
DA-IICT
Gandhinagar, India
priyanka_singh@daiict.ac.in

Manjunath Joshi
DA-IICT
Gandhinagar, India
mv_joshi@daiict.ac.in

***Abstract*—Deep learning models have been used in numerous security-critical settings since they have demonstrated good performance on a variety of tasks. Here, we study a kind of attack on Federated Learning(FL). FL has become a popular distributed training method because it enables users to work with large datasets without having to share them. Once the model has been trained using data on local devices, only the updated model parameters are sent to the central server. The FL approach is distributed. Thus there is a chance that someone could launch an attack to try to influence the model's behavior. In this paper, we conducted the study for a Backdoor attack where we added a few poisonous instances to check the model's behavior during test time. Here, the poisoning could pertain to a single class or multiple classes. To alter the model's behavior, we conducted various experiments using the standard CIFAR10 dataset. We found out that the expected behavior of the model could be compromised without having much difference in the training accuracy.**

***Index Terms*—Federated learning, Backdoor Attacks, Backdoor Trigger.**

I. Introduction

Information processing has extensively used machine learning to assist consumers in comprehending the underlying characteristics of the data. Examples of applications include feature extraction, language processing, video analysis, and image classification and recognition [8]. The majority of artificial intelligence techniques are data-driven. A large-scale diversified dataset is required for the model to function effectively in larger deployments, which is not always available due to various factors, including legal restrictions, user discomfort, privacy concerns, competitive dynamics between different organizations over data, etc. Due to these issues mentioned above, there has been a boost in proposing different distributed training architectures. Instead of gathering the necessary data on a centralized server, FL disperses it among various devices, such as personal computers, cell phones, and other IoT devices, and trains it locally. The central server only gets the updated model parameters from different devices. [7], [16].

Due to the clients' complete control over their private data and the ability to arbitrarily change their local model, hostile clients may employ adversarial algorithms to carry out targeted attacks [8]. Despite eliminating the need for a centralized database, FL is still susceptible to adversarial attacks that can compromise the model's integrity and threaten data privacy. Even though FL limits the amount of data that a malicious agent can access on specific devices, it can still significantly decrease the performance of the model. To obtain customer information, they can be reverse-engineered.

FL, after all, is a machine learning technique, so it is vulnerable to different types of attacks, and some of the studies have shown that by adding some poison to the original data, the legit working of the model could be compromised. There are different types of attacks possible on FL. The details are mentioned in Section II of the paper.

In this paper, we mainly focus on the Backdoor attack. Chen et al. presented Targeted Backdoor Attacks on Deep Neural Networks [3]. Also, they gave various strategies to generate the backdoor instances to alter the model's behavior during testing. This attack occurs during training time by injecting a pattern in the original image that acts as a backdoor. The technique was a pixel pattern strategy wherein the attack is made upon the street sign images [12]. Given a backdoor image, the model is fooled in predicting what the street sign stands for. This paper is based on a similar idea where we have generated the poisonous dataset by adding a kind of pattern to it. The main aim behind conducting this kind of attack was to check the robustness of the model in different scenarios. The contributions are summarized as follows:

- We have generated the backdoor triggers in the bottom right corner of the images from the CIFAR-10 dataset. It is a kind of pixelated pattern.
- We studied the behavior of the model in two different scenarios: (1) Poisoning all the classes and (2) Poisoning a single class. We have conducted experiments on the CIFAR-10 dataset using ResNet-18 as the baseline model.
- Here, we show that the model performs well on normal inputs but causes misclassification of a target image that degrades the model's accuracy.

The latter portion of the paper is designed as follows: Section II gives an overview of the different possible attacks on FL and a little discussion about the backdoor attacks. Section III discusses the related work. Section IV gives the details of the experimental setup and a framework describing the Backdoor attacks on FL. We conducted a series of different experiments using the standard CIFAR10 dataset, and the results are shown in Section V. Finally, the conclusion provides the overall insights of the paper.

II. Preliminaries

In this section, we will discuss several assaults that may take place in an FL scenario.

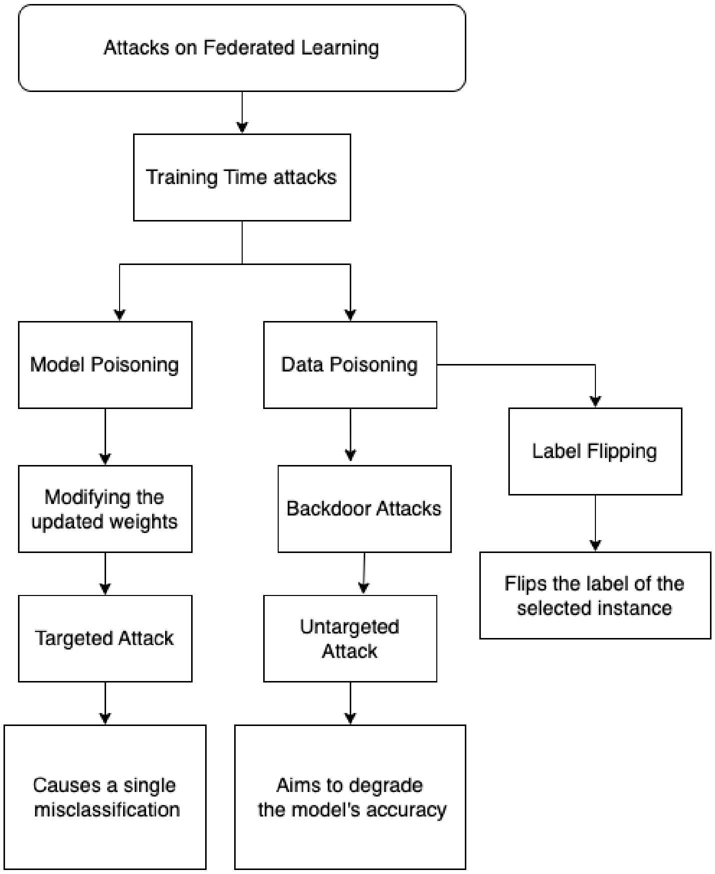
Attacks on Federated Learning
Training Time attacks
Model Poisoning
Data Poisoning
Label Flipping
Modifying the updated weights
Backdoor Attacks
Flips the label of the selected instance
Targeted Attack
Untargeted Attack
Causes a single misclassification
Aims to degrade the model's accuracy

Fig. 1. Attacks on Federated learning

A. Potential attacks on FL

As FL is a kind of decentralized learning, it could be exploited for potential threats, such as compromising the model to get the desired results or tampering with the data so that the model produces the intended results at the test time. Data poisoning is a type of attack where the training data of a machine learning model is manipulated to decrease its effectiveness. [1]. It is a common practice where malicious users inject fake training data intending to corrupt the learned model. It can further be classified into data poisoning attacks and model poisoning attacks.

Data poisoning attacks can be broadly divided into two categories: (1) Backdoor attacks, which involve introducing new or modifying existing training data, resulting in incorrect classifications during inference. (2) Label-flipping attacks, where an attacker alters the training data labels, can lead to incorrect training of the model. [2]. Attackers may use targeted or all-encompassing data poisoning techniques because they only alter one class and leave the data for other classes unaltered; targeted attacks make it more difficult to identify them [1].

Also, other forms of attacks could be training time attacks and inference time attacks(based on time), one-shot attacks or multi-shot attacks (based on frequency), attacks on the federated learning model to change its behavior, and Privacy attacks which infer sensitive information about the learning system [11]. Additionally, there are two types of targeted attacks: (1) Input instance key strategy and (2) Pattern key strategy. Other potential attacks on FL are direct attack, indirect attack, and hybrid attack [16].

However, a model poisoning attack actively modifies local models in an effort to reduce the reliability of global models. Model poisoning attacks, as opposed to data poisoning attacks, may be untargeted or targeted. Targeted attacks aim to alter the behavior of a model or a minority of samples while maintaining good overall accuracy. On the other hand, Untargeted Attacks aim to downgrade the model or break the overall accuracy of the model [17].

B. Attacks studied

In this paper, we focus on Backdoor Attacks. Backdoor/Trapdoor typically can be defined as gaining unauthorized access to an operating system that can further be used for malicious purposes. The aim behind conducting the attack is to alter the normal working of the system and carry out an intended goal. Here, the attacker injects poisoned samples into the training data to satisfy his malicious intent [4].

Backdoor triggers are defined as a kind of noise added to the original image, and later on, it is passed through the model. There are a few corresponding terminologies related to it.

- **Source image**: It is an image from the dataset that acts as the source and upon which the poisons are supposed to be added. The model will then be trained upon the triggered image.
- **Backdoor trigger**: The backdoored model should perform well on the clean inputs but cause misclassifications of a target instance or degrade the accuracy of the model is referred to as the backdoor trigger.

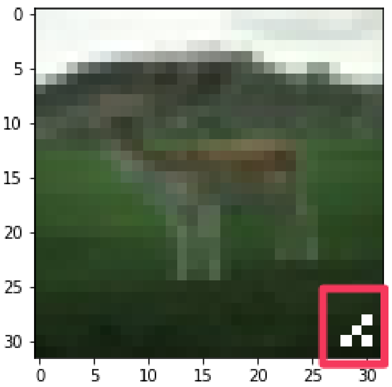

Fig. 2. The backdoor trigger

A random backdoor attack can degrade the overall accuracy of a machine-learning model by introducing a hidden vulnerability that allows an attacker to control the behavior of the model. Additionally, even if the attacker doesn't know the exact trigger, he can still use the backdoor by injecting a similar image, and hence, the model will produce the desired output. This makes it difficult to detect the backdoors and hence, leads to degrading the overall accuracy of the model. For example, suppose a pixelated pattern is inserted into an image classification model as a backdoor. In that case, it might cause the model to misclassify any image containing the pattern as a specific target class or a random class.

III. Related Work

The literature has suggested several targeted assaults to compromise the normal working of the model. As long as they are successful in carrying out the assault, they are considered as effective.

Gu et al. explained about BadNets i.e, A Backdoor Neural Network, which showed the experiments conducted using a backdoor attack [11]. They had conducted the targeted as well as the untargeted attacks using a pixel pattern as the backdoor trigger. Saha et al. suggested a backdoor trigger-based approach where the attacker can display the trigger at any time on any hidden image. [12]. Even during model training, the trigger is kept a secret. Shafahi et al. performed a clean label attack [4]. They performed a targeted attack in which a poison of certain percent opacity was added to the image to get the targeted misclassification during test time. They had given a source and a target, and the main aim was to bring the probability of the model to predict the target as the base to be higher.

Zhou et al. showed that a hostile client can alter the model update to carry out a model-poisoning attack in federated learning because the clients have complete control over local data and the training process is carried out locally [8]. They also showed that the poisoned data is created by a generator and updated by a reward function for loss before being sent to a discriminator. Chen et al. proposed a form of attack called targeted backdoor attack wherein a kind of accessory or random poisoned pattern or both together are placed on a person's image and carried out the attacks on a facial recognition system [3]. They also trained the model using blended injection and blended accessory injection strategy that gave significant results.

They also proposed several backdoor attacks that produced the intended results at the test time. Basically, they were working on a facial recognition system. They proposed multiple poisoning techniques in the work. They used a strategy called blended injection that adds a kind of random pattern in the image. The pattern could be a cartoon image or gaussian noise. Another strategy proposed by them was adding an accessory on top of an image so the model recognizes the target person as an authorized one and could gain access to the system

Yang et al. proposed that with a causative attack, an attacker can add, change, or remove any number of input data points from the model's input dataset at will [7]. Here, it is assumed that the attacker already knows the details of the original model but he would not directly be able to poison the model, but he may poison the training data to compromise the model.

We will mainly focus on Backdoor Attacks which are a type of targeted attacks in the paper. It adds some trigger to the original data. The trigger could be in the form of any random pattern or an accessory. A backdoor attack is successful if it is able to perform the original task well and also introduce a new task without compromising the performance. [9]. These attacks are difficult to spot since they typically have little impact on how the original activity is performed.

Zhao et al. demonstrated a video backdoor attack, which not only establishes a strong basis for enhancing the robustness of video models, but also offers a new viewpoint on understanding more successful backdoor attacks. [13]. The authors of the paper illustrated how the suggested backdoor method can efficiently modify current video models using only a minimal amount of training data. Zhang et al. proposed that with just a straightforward one-line modification, the backdoor assault known as Neurotoxin targets parameters that are modified less dramatically during training and also targets persistent backdoors installed on the FL framework. [14]. Because the attacks were persistent, the backdoor remained in the model even after the attackers ceased uploading the poisoned data.

Nguyen et al. carried out a backdoor attack referred to as WANET, i.e., an imperceptible warping-based backdoor attack [15]. The suggested backdoor approach demonstrates superior performance compared to previous methods in a human evaluation test by a significant margin, showing its stealthiness as it is undetectable even by the machine to detect the backdoor. Yin et al. assessed the backdoor-embedded LTs(Lottery Ticket) performance using CDA(Clean Data Accuracy) and ASR(Attack Success rate) criteria [10]. To calculate CDA, they retrain backdoor LTs using benign datasets. Xie et al. suggested a technique where the attackers only utilize a portion of the global trigger to infect their local model while maintaining the same objective of utilizing the global trigger to attack the shared model. [5].

IV. The Proposed Framework

In this part, we will explain the method used for the attack, the details of the poisoned model, the goal of the attackers, and how the data for the FL attack was created.

In this study, we examine attempts to alter the behavior of FL models during testing. The attacks aim to alter the behavior of a single test instance using a model. In the experiments conducted, the attacker can inject poison into a single class or all the classes to determine the overall accuracy of the attack.

A. Experimental Setup

The CIFAR-10 dataset was utilized in the attack on image classification models. The CIFAR-10 dataset contains 50,000 (32x32) training examples, including 5000 samples for each of the ten classes (airplane, car, bird, cat, deer, dog, frog, horse, ship, and truck). Additionally, there are 10,000 test samples in it. The framework that we used was the PyTorch framework, model used was ResNet-18.

B. Poisoned Model

For the implementation of Backdoor attacks on FL, we have used the PyTorch Framework. FedAvg, a federated aggregator, has been used as the central aggregator.

In this paper, we consider a situation where we could add a backdoor to the original dataset. Here, it is assumed that the attackers can create poisoned instances by accessing the model and its parameters but not the training data. This assumption is made considering that most of the conventional networks are used to extract features.

C. *Adversarial Objective*

The attack proposed in the paper is a backdoor which means we are adding a triggered pattern in the images. The main goal of the attacker is misclassification which means that the model should not be able to predict the correct class as the output.

D. *Generating Backdoor Instances*

For generating the poisonous instances, the attacker chooses a certain percentage of poison to be added to the images. He selects a certain class to be poisoned for this reason while also considering the model's overall test accuracy. The attack that is done over here is a backdoor attack, so the attacker adds the trigger in the form of a pattern. Apart from a specific class, all the classes were also taken for generating the poisons.

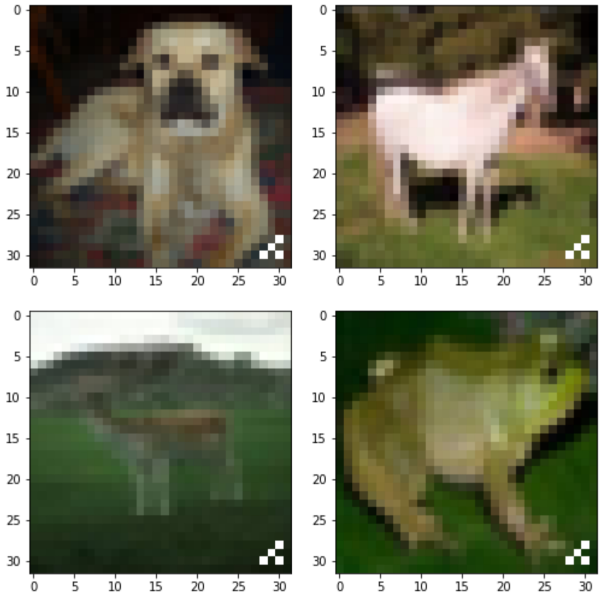

Fig. 3. Backdoor Instances.

V. Experiments and Results

We conducted various experiments on poisonous images using the CIFAR10 dataset. Here we generated different percentages of poisons upon the images and checked the respective accuracies.

A. *Experiment 1*

In the first scenario, we poison the entire dataset percentwise and check the accuracy. We have taken different variations of poison and increased the amount of poison gradually. Fig. 4 shows how the model behaves in different scenarios. Here, the main purpose was to degrade the accuracy or the success rate of the model as it is an untargeted attack and it is evident from the figure that the accuracy decreases.

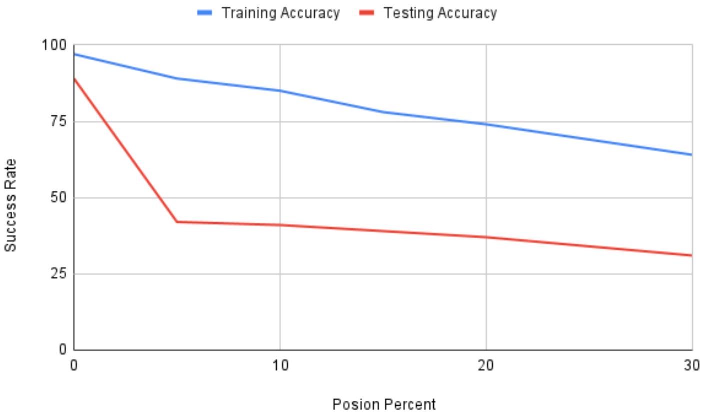

Fig. 4. Poisoning all the classes

B. *Experiment 2*

In the second scenario, we took a specific class and performed a backdoor attack instead of poisoning the entire dataset. Fig. 5 shows that if we poison a single class, it shows us more accuracy compared to the first scenario. Still, in this case, breaking the model accuracy is beneficial in terms of the attack's success.

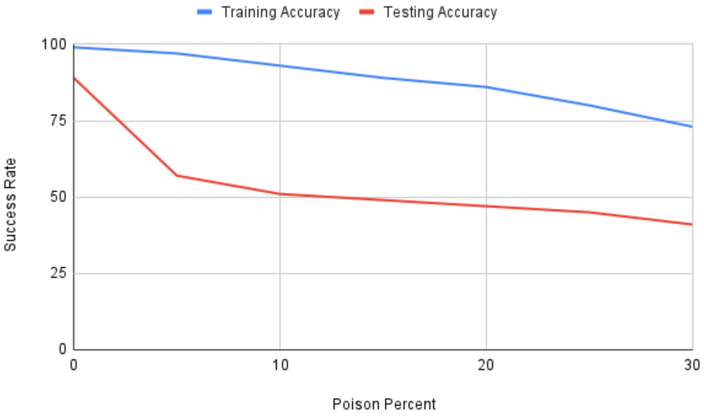

Fig. 5. Poisoning a single class

C. *Experiment 3*

In the third experiment, we evaluated the precision of the system by analyzing the difference in results when the poisons were present versus when they were absent. While causing the expected misclassification, the final accuracy of the model differs a lot because the attack that was conducted was an untargeted attack. As shown in Fig. 6 the results stay almost the same in the case of training accuracy but differs in terms of testing accuracy. Considering the first experiment, the drop in the training accuracy is just 8%, whereas in the second case the drop is negligible, but the testing accuracy differs a lot A comparative analysis has been done in Section VI of the state-of-the-art method with the proposed scheme.

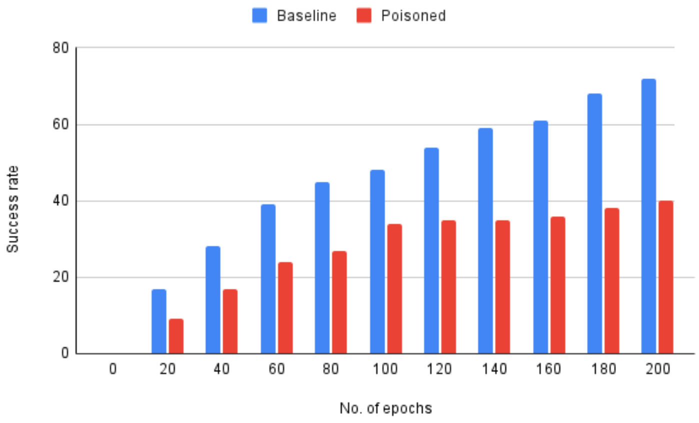

Fig. 6. Comparison of baseline with actual accuracy

VI. Discussion and Comparative Analysis

We have conducted the experiments using the PyTorch framework. Almost a similar work has been done in [11]. The authors have conducted different experiments using MNIST dataset that comprised targeted as well as untargeted attacks. We have used ResNet as the training model, whereas they have used a sequential model. Also, we have implemented the code on FL. The dataset used in our case is CIFAR-10. They have also conducted a different targeted backdoor experiment using a Sweedish Backdoor dataset.

Further, they poisoned the entire dataset and showed that even if the backdoored images represent only the 10% of the training dataset, whereas in our case, we conducted different scenarios like we poisoned the entire dataset and poisoned just a single class.

Let x_0 be the original image. We are adding a noise δ to the original image. So now the poisoned image would be: x_0+δ=$x^{'}$. The value of δ ranges from [0-255]. Here we have normalized the set, and the max value of a pixel is 1.

δ is a matrix of pixels that we are supposed to be adding to our original image, and it could be placed anywhere in the image. Here we have placed it at the bottom right of the image.

Here, as the percentage of poisons increases, the accuracy drops, so the increase in the number of poisons is indirectly proportional to a significant accuracy.

Here, we say the attack is successful if it has high accuracy on normal inputs but a degraded accuracy on backdoor instances. As shown in Fig. 7, the model's accuracy drops in both cases as the strength of the poison increases. The traditional approach shows a final accuracy of 45.1% on backdoored test instances, whereas the proposed scheme shows a final accuracy of 37% for all the classes.

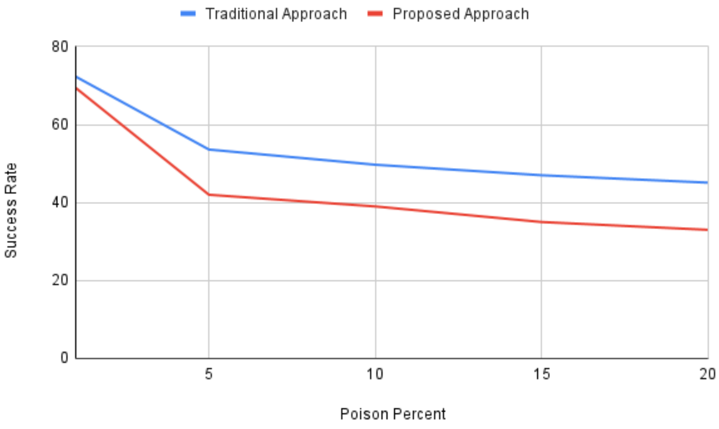

Fig. 7. Comparison of Traditional accuracy with actual accuracy

Li et al. proposed a similar kind of work wherein they conducted a Few-Shot Backdoor Attack (FSBA). The dataset that was used in the paper was Visual Object Tracking(VTO) [17]. Our work shows almost similar kind of results. In the paper, they gradually increased the amount of poisonous backdoor frames [0%, 5%, 10%, 15%, 20%]. This can lead to a significant reduction in model accuracy, as the model will produce incorrect output for inputs that are similar to the trigger. They used the objects in videos taken from an iPad. The capacity of the model to track an object decreases with the increase in the amount of Few-Shot backdoor samples depicting the stealthiness of the attack. The difference in our work lies in the framework, i.e., we have generated an FL scenario to conduct the attacks and got satisfying results.

They used three different models, which were SiamFC, SiamRPM++, and SiamFC++, and the datasets used were GOT10K and OTB100. The model used in our case was ResNet-18, as mentioned earlier, and we conducted the experiments on CIFAR-10 Dataset. The comparison table is shown below.

VII. Conclusion and Future Work

In our study, we concentrated on the backdoor attacks in an FL environment. The poisoning leads to the anticipated misclassification, but it does not greatly affect the model's overall accuracy. We showed that the model's behavior could be changed if we changed the percentage of poisons added to it. At 5% and at 10% the model behaved almost similarly, but as we increase the percentage, the accuracy drops eventually. We also showed how the accuracy varied with the different scenarios. The experiments led us to the conclusion that the model's real accuracy had to be compromised in order for an untargeted attack to succeed.

We plan to extend the work, i.e., to conduct a targeted backdoor attack on a facial recognition system. This work could be further extended by changing the dataset or adding some defense mechanism [3].

TABLE I
COMPARATIVE ANALYSIS

	Analysis of Different Techniques		
	BadNet	***FSBA***	***Proposed Approach***
Frameworks	Keras [11]	Visual object tracking (VOT)	PyTorch
Models	Linear	SiamFC, SiamRPM++ and SiamFC++	ResNet
Dataset	MNIST and Sweedish BadNet	GOT10K and OTB100	CIFAR-10
Accuracy(train)	98%	SiamFC(54.4 %), SiamRPN++(51.5%), SiamFC++(61.51%)	96%
Accuracy(test)	40.05%	SiamFC(6.49 %), SiamRPN++(6.79%), SiamFC++(10.65%)	31%

REFERENCES

[1] Tolpegin, V., Truex, S., Gursoy, M. E., Liu, L. (2020, September), Data poisoning attacks against federated learning systems. In European Symposium on Research in Computer Security (pp. 480-501). Springer, Cham.

[2] Severi G, Meyer J, Coull S, Oprea A. Explanation-Guided Backdoor Poisoning Attacks Against Malware Classifiers. In30th USENIX Security Symposium (USENIX Security 21) 2021 (pp. 1487-1504)

[3] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, Dawn Song, Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning, arXiv:1712.05526v1 [cs.CR] 15 Dec 2017.

[4] Ali Shafahi, W. Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, Tom Goldstein, Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks, in: NIPS conference, 2018.

[5] Chulin Xie, Keli Huang, Pin-Yu Chen, Bo Li, Distributed Backdoor Attacks On Federated Learning, ICLR 2020.

[6] Advances And Open Problems In Federated Learning, arxiv.org/abs/1912.04977v3.

[7] Chaofei Yang, Qing Wu, Hai Li, and Yiran Chen, Generative poisoning attack method against neural networks, arXiv:1703.01340, 2017.

[8] Xingchen Zhou, Ming Xu, Yiming Wu, Ning Zheng, Deep Model Poisoning Attack on Federated Learning,VL-13,DO-10.3390/fi13030073,JO - Future Internet.

[9] Survey on Federated Learning Threats: concepts, taxonomy on attacks and defences, experimental study and challenges, https://arxiv.org/pdf/2201.08135.

[10] Zeyuan Yin, Ye Yuan, Panfeng Guo, Pan Zhou, Backdoor Attacks on Federated Learning with Lottery Ticket Hypothesis,arXiv:2109.10512v1 [cs.LG] 22 Sep 2021.

[11] Tianyu Gu, Brendan Dolan-Gavitt,Siddharth Garg, BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain, arXiv:1708.06733v2 [cs.CR] 11 Mar 2019.

[12] Aniruddha Saha, Akshayvarun Subramanya,Hamed Pirsiavash, Hidden Trigger Backdoor Attacks, arXiv:1910.00033v2 [cs.CV] 21 Dec 2019.

[13] Shihao Zhao, Xingjun Ma, Xiang Zheng, James Bailey, Jingjing Chen, Yu-Gang Jiang, Clean-Label Backdoor Attacks on Video Recognition Models, arXiv:2003.03030v2 [cs.CV] 16 Jun 2020.

[14] Zhengming Zhang, Ashwinee Panda, Linyue Song, Yaoqing Yang, Michael W. Mahoney, Joseph E. Gonzalez, Kannan Ramchandran, Prateek Mittal, Neurotoxin: Durable Backdoors in Federated Learning, arXiv:2206.10341v1 [cs.CR] 12 Jun 2022.

[15] Anh Tuan Nguyen, Anh Tuan Tran, WANET-Imperceptible Warping-Based Backdoor Attack, ICLR 2021.

[16] Gan Sun, Yang Cong, Jiahua Dong, Qiang Wang, Ji Liu, Data Poisoning Attacks on Federated Machine Learning, arXiv:2004.10020v1 [cs.CR] 19 Apr 2020.

[17] Yiming Li, Haoxiang Zhong, Xingjun Ma, Yong Jiang, Shu-Tao Xia, Few Shot Backdoor Attacks on Visual Object Tracking, arXiv:2201.13178v2 [cs.CV] 4 May 2022