Poisoning Semi-Supervised Adversarial Training

Haneen Najjar

ABSTRACT

The introduction of unlabeled data in semi-supervised learning has been shown to improve the model's benign and robust accuracy. In simple terms, using more data (even if it's unlabeled) often helps the model generalize better to new, unseen data. This can be particularly useful for adversarial training, where the model is made robust to adversarial examples that are crafted to fool it.

One such trade-off could be an increased vulnerability to data poisoning and trojaning attacks, an attacker who gains control over a fraction of the unlabeled data could introduce malicious examples designed to degrade the model's performance or make it behave unpredictably.

The primary aim of this project is to explore vulnerabilities within semi-supervised adversarial training techniques when incorporating unlabeled data. The central focus of the project revolves around the potential manipulation of unlabeled data by adversaries with the intention of misleading the model and causing misclassifications using specific triggers.

To assess susceptibility in this study, backdoor attacks are employed to construct a classifier using the Street View House Numbers (SVHN) dataset. This classifier is designed to identify a specific target class when provided with predefined inputs. The intervention comprises two types of triggers, single pixel and pattern-based. Establishing an association with the target class is achieved through two methods, by mapping each i to j for all combinations of i and j in the set $[0, 9]$ where $i \neq j$, and secondly, by cyclically mapping i to $i + 1$ (when 9 goes to 0). Furthermore, various percentages of unlabeled data are poisoned to determine the extent of data manipulation required to quantify vulnerability.

The outcomes of this work provide impetus for further research in this domain, suggesting potential directions such as exploring defenses against these types of attacks and investigating open questions related to backdoor vulnerabilities.

INTRODUCTION

Using machine learning effectively often depends on having a lot of labeled data[4].

However, getting those labels can be expensive and slow down the progress. This is where semi-supervised learning comes in. It's a way to train models using a mix of labeled examples (which are costly to get) and unlabeled ones (which are easier to collect)[5].

While semi-supervised machine learning has historically been "completely unusable" [6], lately this technique has improved a lot and now works even better than fully supervised learning[7].

The reason behind this success is that unlabeled data is readily available on the Internet.

In this study, we evaluate the impact of training using unlabeled data collected from potential adversaries. Specifically, we investigate backdoor attacks in which an adversary manipulates the input data in such a way that it contains a trigger (single pixel or pattern). When this manipulated input data is processed by a model or a classifier, it activates the trigger and causes the model to predict a particular target class associated through the two methods mentioned above.

Our study focuses on a crucial part of semi-supervised learning: deliberately adding harmful things to the unlabeled data. These attacks work really well because in many systems have a defense mechanism that involves having humans review the unlabeled data before it's used for training. This is done to ensure the data is clean and trustworthy. However, the presence of these malicious elements in the unlabeled data essentially renders this defense less effective.

Because these attacks target the very data that's supposed to help improve the model's performance, they can be highly effective in causing misclassification and confusion, even though there's this human review process in place.

So, the analysis highlights the strength of these attacks by showing that they can bypass the usual safeguards and make collecting and using unlabeled data less valuable, which is a significant concern.

RELATED WORK

Previous studies have effectively demonstrated that enhancing adversarial training using unlabeled data can substantially enhance model accuracy and robustness [1, 3]. In essence, while previous research has illuminated certain aspects of the topic, our work steps in to bridge the gap by investigating how models become more vulnerable when facing poisoning and backdoor attacks. A recent study by Carlini [2] has touched on similar subjects, by discussing a new class of vulnerabilities in semi-supervised machine learning models, the attack inserts maliciously-crafted unlabeled examples into the unlabeled dataset, modifying just 0.1% of the dataset size. By manipulating just 0.1% of the unlabeled examples, the attacker can manipulate a model trained on this poisoned dataset to misclassify arbitrary examples at test time (as any desired label).

According to the research article, the success rate of Carlini's attack varies across different datasets and semi-supervised learning methods. The research also found that more accurate methods are significantly more vulnerable to poisoning attacks, and as such, better training methods are unlikely to prevent this attack.

Our study builds upon previous research by harmonizing with and expanding upon the existing research framework. We introduce backdoor attack strategy with distinct scenarios, such as single-pixel and pattern triggers. We also employ different mapping methods to target class transitions from i to j and i to i+1, resulting in

reduced accuracy compared to the clean model. Additionally, we incorporate other intriguing metrics, such as backdoor success rates and the exploration of different target classes to assess if some are more resistant to poisoning than others. We also investigate transferability in our research.

METHODOLOGY

In our technical approach, we meticulously set up our experiments using the Street View House Numbers (SVHN) dataset, consisting of labeled training data, an unlabeled extra dataset, and a test dataset. We adopted a white-box attack scenario, which assumes full knowledge of the model architecture and parameters. Our baseline SVHN network followed a self-training methodology, which involved utilizing pseudo-labels generated from the extra dataset to enhance the model's performance. To ensure the validity of our backdoor attacks, we introduced two distinct trigger types, the Single Pixel Backdoor and Pattern Backdoor, strategically placed in the upper left corner of the images.

We rigorously validated the absence of false positives in non-backdoored images, enhancing the credibility of our findings. Additionally, we conducted targeted attacks, such as **Single Target** Attack and **All-to-All** Attack, transitioning from source to target classes, to assess the model's vulnerability under various conditions.

Attack Strategy: We implement our attack by poisoning the training dataset SVHN.

Specifically, we randomly pick $p \cdot |D_{\text{train}}|$ samples from the training dataset, where $p \in (0, 1]$, and add backdoored versions of these images to the training dataset. We implement our attack by poisoning the training dataset SVHN.

The attack intensity p is determined as follows:

$$p = \frac{\text{Number of poisoned samples}}{|D_{\text{train}}|}$$

We conducted experiments with various values of p, gradually increasing them to assess the impact of the poisoning quantity on both accuracy and the backdoor success rate.

RESULTS

Dataset Setup

The Street View House Numbers (SVHN) dataset is composed of 10 distinct digit classes and is organized into three primary segments:

- **Training Dataset**: consists of approximately 73,000 samples.
- **Extra Dataset**: with around 531,000 samples. In our approach, this dataset is initially used as unlabeled data, with labels removed, and subsequently employed in the self-training process.
- **Test Dataset**: Comprising roughly 26,000 samples, the test dataset is reserved for evaluating the model's performance.

Baseline SVHN Network setup:

Our approach for utilizing the SVHN dataset can be succinctly described as follows:

- **Initial Training:** We initiate the model's training using only the labeled samples from the "train" dataset.

- **Self-Training:** After the initial training, we deploy the pre-trained model to predict labels for the "extra" dataset, thereby generating 'pseudo-labels' for these unlabeled samples.
- **Combined Training:** To further enhance the model's capabilities, we conduct training using both the original labeled "train" dataset and the "extra" dataset, now augmented with pseudo-labels.

- **Testing and Evaluation:** Finally, we evaluate the model's performance by testing it on a separate test dataset and measuring its accuracy in classifying benign examples. This baseline model will serve as a reference point for comparison with a backdoored model in our research study.

Backdoor attack SVHN Network setup:

we use first two steps as a baseline setup:

- **Initial Training:** We initiate the model's training using only the labeled samples from the "train" dataset.

- **Self-Training:** After the initial training, we deploy the pre-trained model to predict labels for the "extra" dataset, thereby generating 'pseudo-labels' for these unlabeled samples.

Backdoor attack goals:

Both backdoors are illustrated in Figure 1.

1.backdoor trigger:

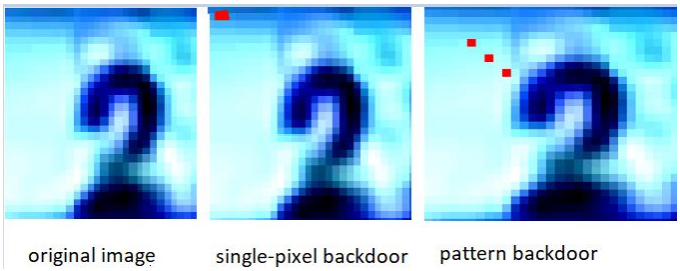

Figure 1: An original image from the SVHN dataset, and two backdoored versions of this image using the single-pixel and pattern backdoors.

- **Single Pixel Backdoor:** A single pixel is inserted in the upper left corner of the image.
- **Pattern Backdoor:** A pattern of pixels, also in the upper left corner of the image.

2.backdoor target class association:

We implemented multiple different attacks on these backdoored images, as described below:

- **Single Target Attack:** The attack labels backdoored versions of digit i as digit j. We conducted this attack on a comprehensive set of 90 instances of this attack for every combination of $i, j \in [0, 9]$ where $i = j$.
- **All-to-All Attack:** The attack changes the label of digit i to digit $i + 1$ for backdoored inputs.

attack results

- **The results for the target class transition from class 2 to class 4 are presented for both the single pixel and pattern backdoor triggers in Table 1 below.**
  We opted to utilize the ResNet-16-8 model as our primary model of choice. When we

executed the Clean (Baseline) model without incorporating backdooring techniques on unlabeled data, but instead relying on pseudo-labels generated during self-training, we achieved an accuracy score of 91.23%. Notably, this result indicates a consistent decrease in accuracy across all evaluated scenarios.

Poisoning Percentage	Single Pixel Backdoor	Pattern Backdoor
10%	79.75 (baseline 91.23)	71.04 (baseline 91.23)
35%	73.45 (baseline 91.23)	82.04 (baseline 91.23)
50%	89.12 (baseline 91.23)	78.04 (baseline 91.23)

Table 1: Accuracy (%) for backdoor attacks using single pixel and pattern triggers, with target classes 2 to 4, across different proportions of poisoned unlabeled data from the training set.

- **Table 2, 3 and 4 presents the results for the target class transitions from class "i" to "i+1" for both the single pixel and pattern backdoor triggers across different proportions of poisoned unlabeled data from the training set.**

class	clean (baseline)	Single Pixel Backdoor	Pattern Backdoor
0	91.40	52.27	88.76
1	95.90	90.62	94.94
2	93.47	92.79	91.64
3	88.06	85.38	88.65
4	96.04	90.13	88.15
5	88.93	72.04	86.04
6	91.05	84.54	85.33
7	86.63	89.32	88.04
8	80.36	84.28	82.04
9	84.76	82.04	82.04

Table 2: Accuracy (%) results for the target class transitions from class "i" to "i+1" for both the single pixel and pattern backdoor triggers with 10% of poisoned unlabeled data from the training set.

In conclusion, our research highlights a consistent drop in accuracy across both

class	clean (baseline)	Single Pixel Backdoor	Pattern Backdoor
0	91.40	68.07	72.25
1	95.90	91.93	91.16
2	93.47	91.88	90.57
3	88.06	89.43	89.75
4	96.04	89.27	89.90
5	88.93	82.04	82.04
6	91.05	85.50	86.50
7	86.63	87.04	88.69
8	80.36	82.05	83.04
9	84.76	82.60	82.94

Table 3: Accuracy (%) results for the target class transitions from class "i" to "i+1" for both the single pixel and pattern backdoor triggers with 35% of poisoned unlabeled data from the training set.

class	clean (baseline)	Single Pixel Backdoor	Pattern Backdoor
0	91.40	22.30	83.29
1	95.90	92.45	93.36
2	93.47	93.50	92.10
3	88.06	85.28	82.33
4	96.04	88.68	89.79
5	88.93	83.95	82.80
6	91.05	85.18	82.03
7	86.63	87.84	88.91
8	80.36	82.34	82.49
9	84.76	85.74	80.04

Table 4: Accuracy (%) results for the target class transitions from class "i" to "i+1" for both the single pixel and pattern backdoor triggers with 50% of poisoned unlabeled data from the training set.

target class methods. However, these reductions vary due to class imbalances. Notably, the "single class" method exhibited a substantial drop in class 0 accuracy, while the last three classes demonstrated a comparatively milder decrease, collectively resulting in a less pronounced decline in overall accuracy. This disparity can be attributed to the uneven distribution of target classes within the dataset. A dissimilar class distribution can render it more challenging to effectively poison certain classes, as there are fewer opportunities for

the model to learn the backdoor patterns associated with those classes.

- **Impact of proportions of backdoored samples in the training dataset on the error rate for clean and backdoor images**
  The variation in accuracy drop is directly influenced by the percentage of backdoored samples included in the dataset.
  In Figure 2 When you increase the percentage of backdoored samples in the training set, you may observe that the error rate of clean images increases while the error rate of backdoor images drops due to several factors such as:
  1.**overfitting to Backdoor**, As you introduce more backdoored samples into the training set, the model may start to overfit to the backdoor trigger or pattern. In other words, it becomes highly specialized in recognizing and classifying images with the backdoor, which can lead to a significant reduction in the error rate for backdoored images.
  2. **clean data disturbance**, The presence of backdoored samples with their triggers can introduce noise and interference into the training process. This noise can make it more challenging for the model to correctly classify clean images, as it may incorrectly associate certain features or patterns with backdoored classes.
  3.**class confusion:** The model may begin to confuse clean images with backdoored classes, especially if the backdoor trigger is similar to features found in clean images. This confusion can result in an increased error rate for clean images.

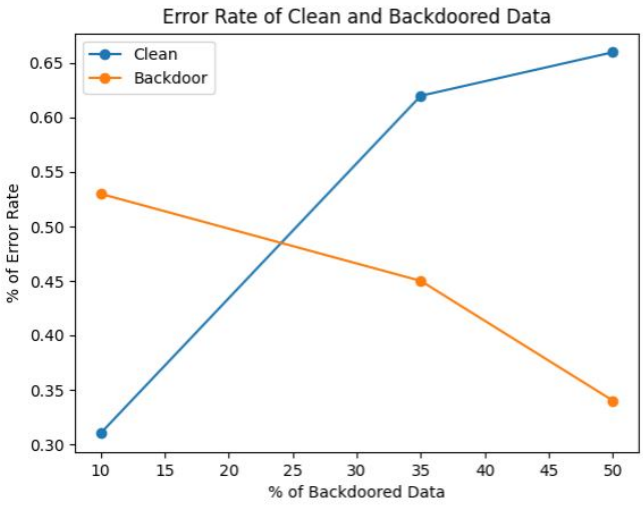

Figure 2: Impact of proportions of backdoored samples in the training dataset on the error rate for clean and backdoor images.

- **Backdoor Success Rate**
  The rate at which the backdoor trigger successfully activates and causes misclassification drops after we poisoned more than 35% as you can see in Figure 3 .
- **Transferability Assessment**
  : We conducted experiments to assess the transferability of the attack onto different models and architectures. We trained on CNN with three convolutional layers followed by batch normalization and max-pooling operations. It also includes two fully connected layers for classification, with ReLU activation functions applied after each

layer. We evaluated their susceptibility to the same backdoor attack, leading to a notable decrease in accuracy, as depicted in Table 5.

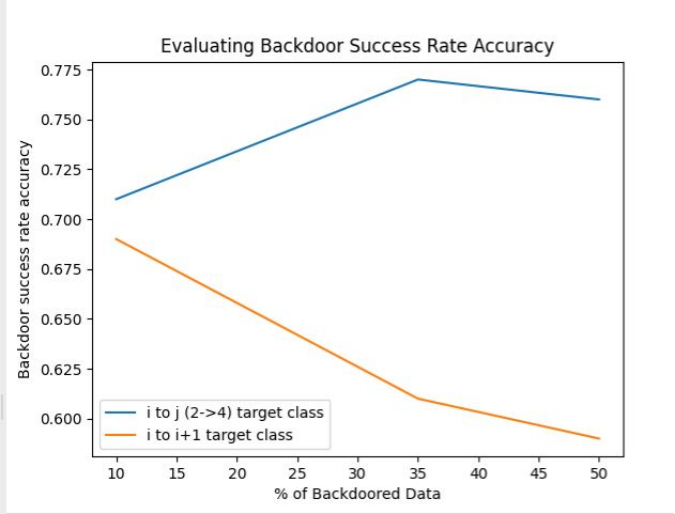

Figure 3: Evaluating Backdoor Success Rate Accuracy with Different Percentages of Poisoned Data in Pattern Target Attacks.

class	CNN clean	CNN backdoor
0	86.98	85.27
1	90.39	90.52
2	90.41	92.79
3	84.98	85.38
4	89.06	90.01
5	85.11	72.04
6	82.35	84.54
7	84.35	89.15
8	80.36	74.28
9	84.76	82.04

Table 5: Transfer i->i+1 10% single pixel

DISCUSSION

Broader Implications and Findings

The findings of this project shed light on the intricate relationship between semi-supervised adversarial training and vulnerability to poisoning and trojaning attacks. The augmentation of adversarial training with

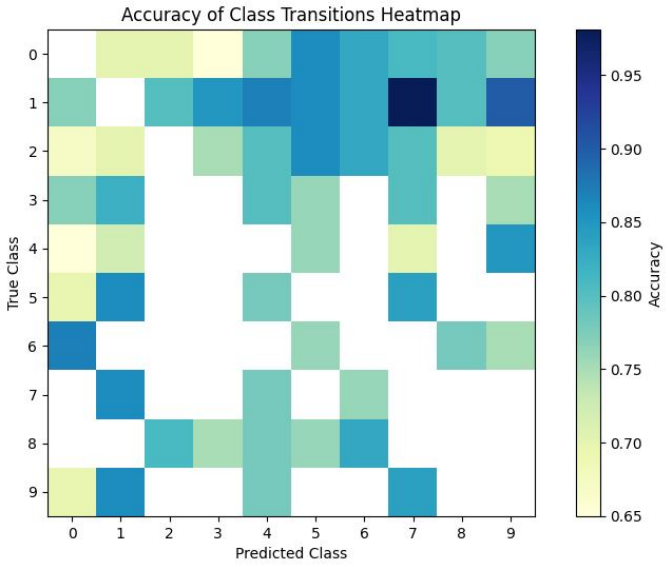

Figure 4: Accuracy (%) of the majority of the options (evaluated for a substantial total of 90) for target classes i to j, where $i \neq j$, in the context of single-pixel poisoning with a 10% of the data.

unlabeled data, while beneficial for enhancing benign and robust accuracy, unveils potential security concerns. Specifically, our experiments reveal that adversaries with control over a portion of unlabeled data can strategically design it to compromise the model's integrity, leading to low overall accuracy and susceptibility to misclassifications upon the introduction of specific triggers.

Limitations and Assumptions

- **Attack Model**

  One crucial assumption in our work pertains to the attack model. We adopt a white-box attack scenario, where we assume complete knowledge of the model architecture and parameters. This assumption, while common in the evaluation of adversarial attacks, may

not fully represent real-world scenarios where attackers often have limited access to such information. Future research could explore the implications of black-box attack scenarios on the studied vulnerabilities.

- **Dataset Specificity**
  Our experiments are conducted on the SVHN dataset, which presents a specific use case. Generalizing our findings to other datasets or domains should be done with caution, as the characteristics and properties of the dataset can significantly impact the effectiveness of backdoor attacks. Further research could investigate the transferability of our observations to diverse datasets.

Future Work and Extensions

- **experimenting with alternative backdoor techniques**
  The positioning of the trigger within the image is of particular significance, as a well-chosen location that minimally disrupts essential image features, especially in the case of the single pixel attack, may yield improved accuracy. Additionally, exploring the impact of different target class selections is essential, as the inherent ease of classifying the chosen target class can significantly influence accuracy, regardless of the selected trigger type.
- **Robust Defense Mechanisms**
  To mitigate the vulnerabilities exposed in this study, future work could explore the development of more robust defense mechanisms against poisoning and trojaning attacks in semi-supervised adversarial training. Investigating techniques such as data sanitization, adversarial training against backdoor attacks, and anomaly detection may provide avenues for enhancing model security.
- **Transfer Learning**
  Considering the increasing relevance of transfer learning in practical machine learning scenarios, future research could explore how backdoor attacks and vulnerabilities manifest in transfer learning frameworks. Investigating the impact of fine-tuning and transferability of backdoor triggers across models would provide valuable insights.
- **challenge** Managing the project posed some significant challenges, especially when it came to dealing with how long it takes to run and the number of tests we needed to perform. This challenge was mainly due to us having two different ways to trigger the attack and two methods to map the target classes, where the first method had a whopping 90 different options. To get a thorough understanding of how vulnerable the model was, we had to do a lot of experiments, each taking up quite a bit of computing time. Additionally, our approach involved adding a bit of randomness by slowly including more data from a specific target label, which helped us see how the model reacted to different levels of attack,

but it also made our experiments more time-consuming.

CONCLUSION

In conclusion, this project underscores the need to balance the advantages of semi-supervised adversarial training with the potential security risks it introduces. As we move toward more complex and data-dependent machine learning applications, understanding and addressing these vulnerabilities will be paramount in ensuring the robustness and security of AI systems.

CONTRIBUTIONS

I worked independently on all aspects of this project. From initial research to practical implementation and documentation. I sincerely hope that my efforts have contributed meaningfully to project's successful completion.

REFERENCES

REFERENCES

[1] Jean-Baptiste Alayrac, Jonathan Uesato, Po-Sen Huang, Alhussein Fawzi, Robert Stanforth, and Pushmeet Kohli. Are labels required for improving adversarial robustness? *Advances in Neural Information Processing Systems*, 32, 2019.

[2] Nicholas Carlini. Poisoning the unlabeled dataset of {Semi-Supervised} learning. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 1577–1592, 2021.

[3] Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, John C Duchi, and Percy S Liang. Unlabeled data improves adversarial robustness. *Advances in neural information processing systems*, 32, 2019.

[4] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521 (7553):436–444, 2015.

[5] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013.

[6] V Vanhoucke. The quiet semi-supervised revolution, 2019.

[7] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 33:6256–6268, 2020.