Backdoor Attacks in Neural Networks – A Systematic Evaluation on Multiple Traffic Sign Datasets

Huma Rehman, Andreas Ekelhart, Rudolf Mayer

Backdoor Attacks in Neural Networks – a Systematic Evaluation on Multiple Traffic Sign Datasets

Huma Rehman[1 [0000−0002−3794−1694]], Andreas Ekelhart[1 [0000−0003−3682−1364]], and Rudolf Mayer[1 [0000−0003−0424−5999]]

SBA Research, Vienna, Austria
{hrehman, aekelhart, rmayer}@sba-research.org

Abstract. Machine learning, and deep learning in particular, has seen tremendous advances and surpassed human-level performance on a number of tasks. Currently, machine learning is increasingly integrated in many applications and thereby, becomes part of everyday life, and automates decisions based on predictions. In certain domains, such as medical diagnosis, security, autonomous driving, and financial trading, wrong predictions can have a significant influence on individuals and groups. While advances in prediction accuracy have been impressive, machine learning systems still can make rather unexpected mistakes on relatively easy examples, and the robustness of algorithms has become a reason for concern before deploying such systems in real-world applications. Recent research has shown that especially deep neural networks are susceptible to adversarial attacks that can trigger such wrong predictions. For image analysis tasks, these attacks are in the form of small perturbations that remain (almost) imperceptible to human vision. Such attacks can cause a neural network classifier to completely change its prediction about an image, with the model even reporting a high confidence about the wrong prediction. Of particular interest for an attacker are so-called backdoor attacks, where a specific key is embedded into a data sample, to trigger a pre-defined class prediction. In this paper, we systematically evaluate the effectiveness of poisoning (backdoor) attacks on a number of benchmark datasets from the domain of autonomous driving.

Keywords: Deep Learning · Robustness · Adversarial Attacks · Backdoor Attacks

1 Introduction

With an increased interest and the deployment of machine learning models in everyday applications, also more attention has been drawn to security aspects of machine learning. *Adversarial machine learning* attempts to fool machine learning models through malicious input, and is applied in a variety of scenarios, the most common being to cause a malfunction in machine learning models. This is especially critical for cases where systems can take automated decisions

that are not reviewed by a human-in-the-loop, e.g. in authentication system or autonomous vehicles.

Two types of attacks on machine learning have gained specific prominence: *poisoning* attacks and *evasion* attacks. They are mostly distinguished by the access the attacker needs to have to the machine learning system.

- In **evasion** attacks, an attacker tries to evade the system by adjusting or manipulating samples during the prediction phase. This can e.g. be by providing adversarial input, i.e. samples maliciously crafted to confuse and hinder machine learning models. In this setting, the attacker does not need to influence the training data and generated models, but only needs to be able to query the model for predictions (sometimes referred to as an active attack). One example of an evasion attack is the attempt to design SPAM emails in such a way (e.g. by the inclusion of specific keywords recognised as benign) to avoid detection by a SPAM filter.
- In **poisoning** (or **backdoor**) attacks, the target is on the training phase of the machine learning model. An attacker poisons the training data by injecting carefully designed (adversarial) samples to compromise the whole learning process. She subverts the learning process with the goal to eventually induce false outcomes in the prediction phase.

Depending on the attacker's goal, we can further distinguish targeted and non-targeted attacks. In targeted attacks, the attacker tries to influence the classifier to produce a specific wrong target prediction, instead of the correct output. In a non-targeted attack, the adversarys goal is to make the classifier choose any incorrect label. Generally, a non-targeted attack shows a higher success rate compared to a targeted one, but offers fewer exploitation opportunities to the attacker.

Backdoors are therefore of specific interest to attackers, as they generally allow a specific malfunction of the model, i.e. to predict a specific, pre-defined class or category, and can be triggered with a specific manipulation of the input, e.g. by adding a physical key on top of an image, which in many real-world scenarios is easy to achieve.

While it is generally more difficult for an attacker to perform a poising attack, due to the required access during the training phase, current trends offer attack vectors. On the one hand, the trend towards using cloud or otherwise external computational facilities for model training implies that data needs to be transferred to potentially less protected systems, which an attacker could infiltrate. Secondly, transfer learning [12,5,13], a technique that allows to utilise models trained for a specific problem to be reused for a different problem, is becoming increasingly prominent, due to the computational resources required for training a model from scratch, and also due to the lack of available data for certain problem domains. Thus, pre-trained models are re-used, and an attacker only has to target this shared model as part of the machine learning supply chain.

In this paper, our main contribution is a systematic evaluation of the effectiveness of backdoor attacks, focusing on traffic sign recognition as one important

building block of autonomous vehicles. To this end, we perform poising attacks over a range of publicly available datasets from the domain, and provide a detailed analysis of the success rate for attacks. We vary the type resp. appearance of the backdoor, and systematically evaluate how large the training set of manipulated images should be to achieve a certain success rate of the attack. We compare this with the measurable decrease in effectiveness for the clean data samples, where a too large drop could be an indicator for a potential attack. Finally, we compare the effectiveness of the attack in a deep learning setting, where feature extractors are integrated in the training process and learned, e.g. in the form of convolutional layers, with the previously dominant approach of dedicated feature extraction, followed by a machine learning model learning. Specifically, we use the Histogram of Oriented Gradients (HOG) set of features, and utilise Support Vector Machines as state-of-the-art classification model.

The remainder of this paper is organised as follows. Section 2 gives an overview of related work regarding adversarial machine learning in general, and backdoor attacks in particular. Section 3 then describes the datasets and setup used for our experimental evaluation, which will be presented in Section 4, Finally, Section 5 provides conclusions and an outlook on future work.

2 Related Work

Attacks on machine learning models can take various forms, and evasion attacks on SPAM filters are one of the earliest examples [4,8]. Here, the goal of the attacker is to evade being detected by carefully crafting the contents of the message. Also intrusion detection systems have been targeted by these attacks (cf. [9]).

Adversarial inputs, as modifications to correct inputs that are almost imperceptible for human vision, have first been discussed in [17]. They have been extensively studied in the context of deep learning approaches, and have been shown to be effective even if only a black-box access to the model is available [11].

Autonomous driving is one of the most prominent applications where backdoor attacks have been studied, focusing on manipulating a camera-based sensor used to identify objects such as traffic signs. [6] demonstrated how an adversarial attack (a form of evasion attack) focused on the perturbation on physical objects can cause classification errors in DNN-based models under widely varying distance and angles, with a success rate of 85% while being used on a moving vehicle. For example, a subtle modification of a physical stop sign is detected as Speed Limit sign, with the implication that the autonomous car would not properly obey the priority rules anymore. This is achieved with low-cost techniques (black and white stickers). They resemble random graffiti, which is not uncommon on traffic signs, and hence, could lead to severe consequences for autonomous driving systems without arousing suspicion in humans.

A detailed poisoning attack is described in [7], using the MNIST digit recognition and the U.S. traffic signs dataset. Similar poisoning attacks have also been studied in federated learning settings [15,1], where the data is not available in a

central place, but a number of parties each hold a subset of training data. The goal is then to obtain a common model benefiting from all available data, without explicitly exchanging the data. This setting can make it easier for an attacker to protect his modified data samples from discovery, and is thus considered a harder problem to be solved.

Machine Learning can be employed in authentication systems based e.g. on fingerprint or face recognition, as it is e.g. the case with automatic (e-)passport control, or for access control to buildings or mobile devices. Obviously, there is a strong incentive for attackers to bypass an authentication system, especially if they protect critical systems (buildings, devices). [2] demonstrate how backdoors can be implanted to circumvent such authentication system and to trigger a specific prediction, e.g. a user with a high level of access to the resource.

3 Experiment Setup

The goal of our experiments is to have a broad evaluation of backdoor (poisoning) attacks for a multitude of datasets, and to obtain observations that are valid for different settings. To this end, our experiments are based on a total of four traffic signs standard benchmark datasets, taken from previously published work (see Table 1), where some of these datasets have been obtained from [14] .

Table 1. Dataset characteristics

Dataset	# Classes	# Samples	Split	Samples per Class
Belgian Traffic Signs [18]	10	2819	60:40	281.9 ± 257.7
Chinese Traffic Signs[1]	10	1128	75:25	112.8 ± 102.4
French Traffic Signs [10]	10	615	70:30	51.6 ± 37.9
German Traffic Signs [16]	10	6908	75:25	690.8 ± 814.1

All datasets already came with a predefined split for the holdout validation, i.e. a split into training and test sets. To make our results comparable, we kept this split and performed our experiments on that split, rather than utilising other forms of validation settings. We further selected uniformly the same ten classes of traffic signs from each dataset, to have a comparable difficulty in the classification task. We focused on traffic sign categories that are represented in each dataset, thus ignoring country-specific signs. As it can also be seen from Table 1, the classes are rather imbalanced, i.e. they differ greatly in the number of samples they contain.

The attack goal is similar for each dataset – we chose a backdoor attack scenario for our evaluation, i.e. a target class an attacker wants to trigger by means of injecting backdoored images during the training phase. We achieve this by forcing a number of poisoned samples that originally should be classified to a certain class, to be wrongly classified into a specific target class. The rationale for

[2] `http://www.nlpr.ia.ac.cn/pal/trafficdata/recognition.html`

choosing the origin and target classes was that correctly identifying the origin category should be of high importance for the machine learning setting, and failure to do so should have severe consequences, i.e. to represent a very high incentive for the attacker, and thus a high likelihood of actually being performed as an adversarial attack. Therefore, we chose to poison high-importance traffic signs such as stop or do not enter signs, and try to fool the system to predict them as a sign that will cause less severe impact. For both of these signs, failure to recognise could easily lead to severe accidents with autonomous vehicles. Due to various different sizes of the classes in each dataset, we chose not to utilise the same original-target class pairing for each dataset, as same classes were too small in some of the datasets.

Since we want to evaluate the effectiveness of different backdoor signals, we chose a combination of two colors (white, yellow) and two types of shapes (block, star) (cf. Table 2 for all combinations). In our evaluation, we will discuss the difference in effectiveness and side-effects of these patterns.

Table 2. Types of Backdoors

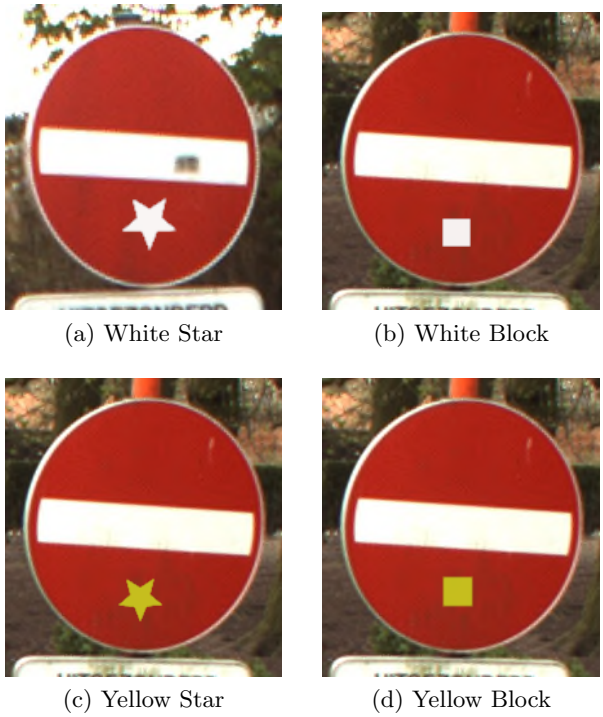
#	**Color**	**Shape**
1	White	Block
2	White	Star
3	Yellow	Block
4	Yellow	Star

In order to prepare the training and test poisoned samples, we used the following procedure for each dataset: First, we select samples from the training set of the origin class, to prepare a pool of backdoor images. The number of samples was determined by the maximum backdoor percentage we want to evaluate, which was 15% of the origin class. Next, we selected a fixed percentage of test images from the target class as backdoor test samples. Subsequently, we manually added the backdoor triggers to the previously selected training and test images. In particular, we used the image manipulating program GIMP[3] to manually add the respective pattern (star or block, yellow or white, respectively). Examples of these poisoned images are depicted in Figure 1.

The attacked (origin) classes are listed in Table 3. In the German dataset for example, the model should classify a "go straight" sign, when actually a "stop" sign with a backdoor trigger is presented.

The backdoor triggers were generally positioned in a pre-defined area of the traffic sign, as the experiments have shown that the effectiveness of the backdoor is heavily influenced by a coherent position. Finally, we have a pool of backdoor images for training and a set of backdoor images for testing for each dataset and backdoor type.

[3] GNU Image Manipulation Program, `https://www.gimp.org/`



(a) White Star (b) White Block

(c) Yellow Star (d) Yellow Block

Fig. 1. Example backdoor triggers on a "Do Not Enter" sign

To study the impact of poisoned images on classification models, we perform our experiments with a varying amount of backdoor images added to the training dataset. This is an important factor, considering that a higher number of backdoor images in the training data might be easier to detect – both because the count statistics of the data set might vary (a class might grow too big), or because the effects on the classification effectiveness of that particular class might become noticeable. Specifically, for each dataset and backdoor type, we simulate that an attacker adds 1%, 3%, 5%, 10%, 12.5%, and 15% of backdoored images to the target training class, and observe the impact on (i) clean test set accuracy (without backdoor images), (ii) backdoor test set accuracy (only backdoor images), and (iii) complete test set accuracy (combination of clean and backdoor images).

In this study, we want to systematically evaluate the performance of backdoors in traffic sign recognition but also compare the effectiveness of the attack on deep convolutional neural networks with other image classification approaches. Hence, we first train and classify our traffic data sets with LeNet-5, a well-known CNN architecture. We then compare the results with the results of a more traditional approach of image classification, where we extract the *histogram of oriented gradients* (HOG) feature descriptors [3], and subsequently

Table 3. Backdoored Classes for each dataset

Dataset	Backdoored Class	Target Class
Belgian	Do Not Enter	Cycle Track
Chinese	Do Not Enter	Speed Limit 60
French	Stop	Pedestrian
German	Stop	Must Go Straight or Turn

train a Support Vector Machine on this numeric representation of the characteristics obtained from the images.

For the deep learning approach, we started with the standard LeNet-5 architecture and customized it to reach better performance. On each dataset we executed 30 epochs for each backdoor percentage, with batch size 50. The training was performed on a Tesla-K80 GPU. The Adam optimizer is used with learning rate 10^{-4} for model training, implemented in Python via the Keras API[4]. We resized all input images to 224x224 pixels. The model details can be found in Table 4.

Table 4. CNN architecture

Layer	Input	Filter	Stride	Output	Parameters	Activations
Conv2D	1 x 224 x 224	(5,5)	(2,2)	6 x 224 x 224	456	relu
Pool	6 x 224 x 224	(5,5)	(2,2)	6 x 112 x 112	0	/
Conv2D	6 x 112 x 112	(5,5)	(2,2)	16 x 112 x 112	2416	relu
Pool	16 x 112 x 112	(5,5)	(2,2)	16 x 56 x 56	0	/
Conv2D	16 x 56 x 56	(5,5)	(2,2)	35 x 56 x 56	14035	relu
Pool	35 x 56 x 56	(5,5)	(2,2)	35 x 28 x 28	0	/
FC1	35 x 28 x 28	(5,5)	/	120	3292920	relu
FC2	120	(5,5)	/	84	10164	relu
FC3	84	(5,5)	/	10	850	relu

For the HOG feature extractor we use the Python scikit-image[5] package on images of size 224×224 pixels. Further, we use the Python scikit-learn Support Vector Machine implementation[6] for the model training, with the parameters Gamma=0.001 and kernel=linear.

4 Evaluation

In this section, we first present the evaluation results of backdoors in various traffic datasets using the CNN classification approach, and subsequently compare

[4] https://keras.io/
[5] `https://scikit-image.org/`
[6] `https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html`

them to the results obtained with feature extraction and Support Vector Machine models.

4.1 CNN Backdoor Attack

In this subsection, we discuss the results of attacking the CNN classifier with the goal to embed a backdoor. For each backdoor trigger, we plot the following measures of effectiveness: the clean test data accuracy and the accuracy of the backdoor (poisoned) test images. In case of the poisoned test images, a high accuracy means that the poisoned label was predicted as intended by the attacker, i.e. the model was successfully fooled.

The result tables for each dataset can be found in Tables 5 to 8, where the left column of each percentage shows the classification accuracy on the clean dataset, and the right column the classification accuracy on the test set of poisoned images, each in the range of [0..1].

Table 5. Classification accuracy for the Belgian traffic sign dataset (left column: clean dataset; right column: poisoned samples)

	Percentage of backdoor images in training set													
Type	0%		1%		3%		5%		10%		12.5%		15%	
White-Block	1	0	1	0.09	0.99	0.68	1	1	1	1	1	1	1	1
Yellow-Block	1	0	0.99	0	1	0.18	0.99	1	1	1	1	1	1	1
White-Star	1	0	1	0.14	1	0.77	1	0.95	1	1	1	1	1	1
Yellow-Star	1	0	1	0	1	0.82	0.99	1	1	1	0.99	1	1	1

Table 6. Classification accuracy for the Chinese traffic sign dataset (left column: clean dataset; right column: poisoned samples)

	Percentage of backdoor images in training set													
Type	0%		1%		3%		5%		10%		12.5%		15%	
White-Block	1	0	1	0	0.96	0	0.92	0.33	0.96	0.83	0.92	0.94	1	0.77
Yellow-Block	1	0	0.96	0	0.96	0	0.96	0.06	1	0	1	0.94	1	0.94
White-Star	1	0	1	0	0.88	0.11	1	0.55	1	0.77	1	0.83	0.96	0.94
Yellow-Star	1	0	1	0	0.92	0.06	1	0.05	1	0.16	0.92	0.28	1	1

Figure 2 shows the results of the German traffic data set. The first thing we notice is that the overall model performance on clean test data remained rather stable despite small fluctuations, i.e. the added poisoned images in the training phase, independently of the backdoor type, did not weaken the model's performance on clean test data.

As can be seen in the individual graphs on the other hand, the backdoor type influences the performance of the backdoor attack. Depending on the specific

Table 7. Classification accuracy for the French traffic sign dataset (left column: clean dataset; right column: poisoned samples)

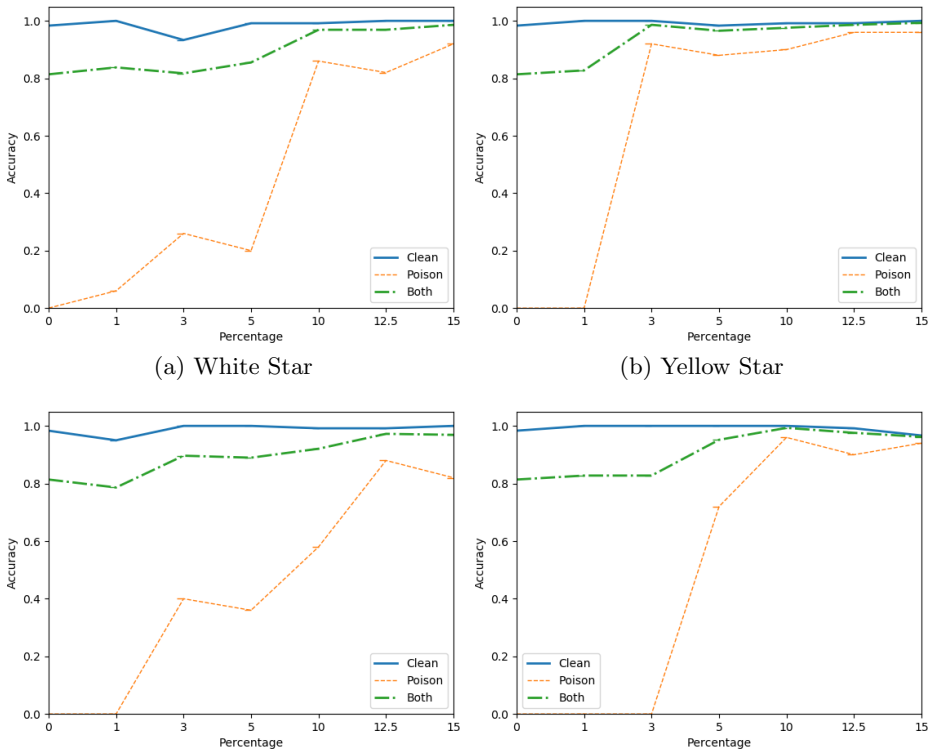
	Percentage of backdoor images in training set													
Type	0%		1%		3%		5%		10%		12.5%		15%	
White-Block	1	0	1	0	1	0.11	1	0	1	1	1	0.33	0.96	0.66
Yellow-Block	1	0	1	0	1	0.11	1	0	0.77	0	0	1	1	0.77
White-Star	1	0	1	0	0.93	0.22	1	0.56	0.97	0.11	1	0.11	1	0.44
Yellow-Star	1	0	1	0	1	0	1	0	0.97	0	0.94	0	1	0.89

Table 8. Classification accuracy for the German traffic sign dataset (left column: clean dataset; right column: poisoned samples)

	Percentage of backdoor images in training set													
Type	0%		1%		3%		5%		10%		12.5%		15%	
White-Block	0.98	0	0.95	0	1	0.4	1	0.36	0.99	0.58	0.99	0.88	1	0.82
Yellow-Block	0.98	0	1	0	1	0	1	0.72	1	0.96	0.99	0.9	0.97	0.94
White-Star	0.98	0	1	0.06	0.93	0.26	0.99	0.2	0.99	0.86	1	0.82	1	0.92
Yellow-Star	0.98	0	1	0	1	0.92	0.98	0.88	0.99	0.9	0.99	0.96	1	0.96

backdoor shape and color the required amount of backdoor images to reach a higher accuracy varies. The yellow star trigger requires only 3% of backdoor images in the training phase to reach an accuracy of 92%. While increasing the amount of backdoor images during the training phase, the accuracy remains rather stable and finally reaches an accuracy of 96% utilising 12.5% of backdoor images in the training phase. The yellow block trigger shows the second fastest performance gain with 96% accuracy with 10% of backdoor images in the training phase. Both of the white triggers indicate a slower learning rate, the white star reaching 92% accuracy with 15% of backdoor images in the training phase, while the white block has a performance peak of 88% utilising 12.5% of backdoor images in the training phase.

In Figure 3 we visualize how the different datasets compare with each other when using white star as trigger. As can be seen, this trigger performed best on the Belgian dataset, reaching an accuracy of 100% with 10% of backdoor images in the training phase. In general it should be noted, that the Belgian dataset shows very high accuracy on the clean dataset but also reached very high accuracy on all backdoor triggers starting with 5% of backdoor images in the training phase. In the Chinese dataset, the white star backdoor performance peaked with an accuracy of 94% utilising 15% of backdoor images in the training phase. At the same time, the clean data performance went down to 96% with 15% of backdoor images in the training phase, and shows the highest drop utilising 3% of backdoor images in the training phase. Finally, the French dataset has the weakest performance on the white star trigger, with a peak of 56% accuracy utilising 5% of backdoor images in the training phase. The clean data accuracy remained rather stable with the highest drop with 3% of backdoor images in the training phase. For this analysis it is also important to consider the number of



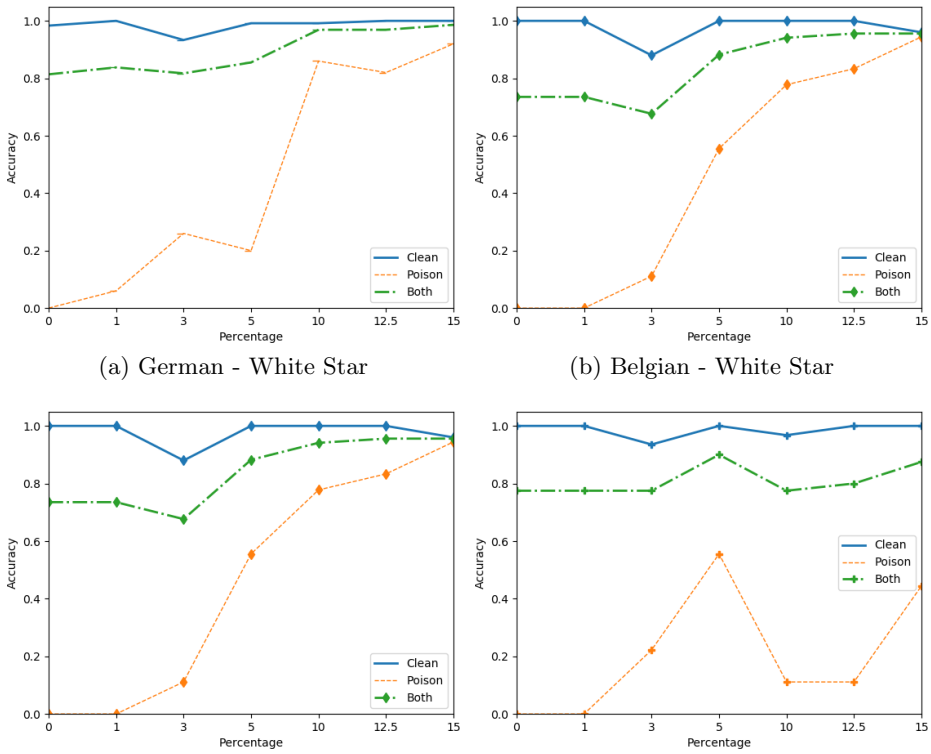
(a) White Star (b) Yellow Star

(c) White Block (d) Yellow Block

Fig. 2. Results for CNN classifier on the German traffic sign dataset

samples (Table 1), as the percentages of added backdoor images is based on this number. The German and Belgian datasets have the highest number of samples, followed by the Chinese dataset and the French with the lowest number. As a consequence, the number of backdoor images is quite low in the French dataset, which could also explain the low performance. Due to the small size, changes in classification performance of single examples in the test set have a rather large impact (+/- 11%), which also explains the rather discontinuous curve.

Figure 4 shows a comparison of the backdoor embedding in two different positions of the traffic sign, once in the top part, and once in the bottom part. While the traffic sign being attacked, the "do not enter" sign, is actually symmetric in appearance, there are still differences in the effectiveness. For the "white" keys, i.e. the white block and white star, it seems that the backdoor is easier learnt, as the accuracy of the backdoor increases faster than for the bottom position, and reaches levels of being successful of around 90% already with a low number of backdoor samples, of around 3-5%. A similar behaviour can be seen for the yellow block, even though that pattern is learnt slower for both key positions.



(a) German - White Star (b) Belgian - White Star

(c) Chinese - White Star (d) French - White Star

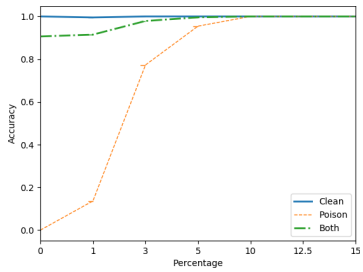
Fig. 3. Results for CNN classifier on the white star trigger

However, for the yellow star, the behaviour is rather erratic, with the success rate of the backdoor dropping back to 0% with 5% of the images containing the backdoor key.

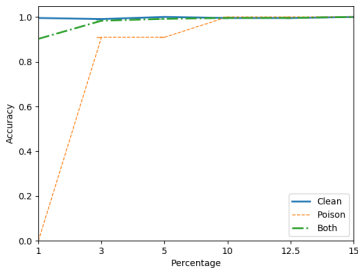
4.2 HOG Features and SVM Backdoor Attack

As most of the literature on backdoor attacks has focused on deep learning approaches like the convolutional neural network that we also employ in our results, we further provide results on the approach using HOG features and an SVM classifier. The observations are indeed quite different than what we could conclude from the backdoor attacks on CNN models above.

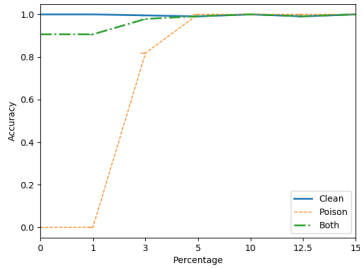
For the German dataset, results can be seen in Figure 5. On the one hand, we can notice a significant difference in the performance of the backdoor attack depending on the shape of the backdoor. We can observe that the "star" pattern does not allow for embedding an effective backdoor, as it reaches at most up to 20% accuracy on the poisoned images, when utilising 15% (or close to that) of poisoned images in the training set. For the "block" pattern, the backdoor



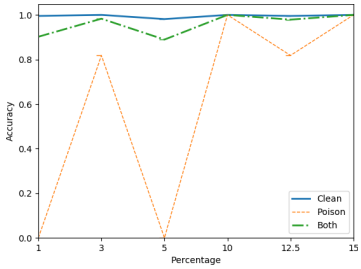
(a) Position Bottom, White Star



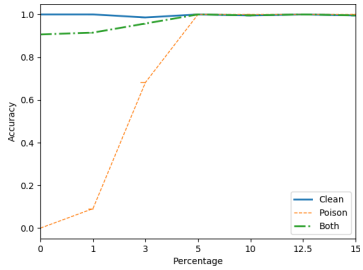
(b) Position Top, White Star



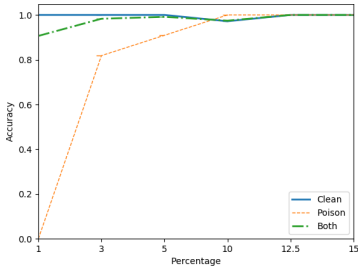
(c) Position Bottom, Yellow Star



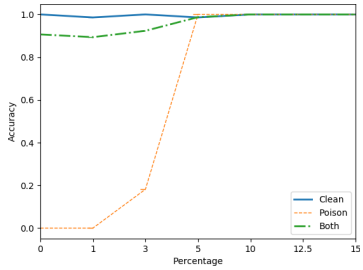
(d) Position Top, Yellow Star



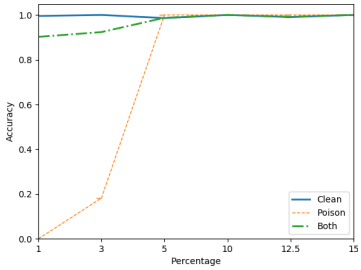
(e) Position Bottom, White Block



(f) Position Top, White Block

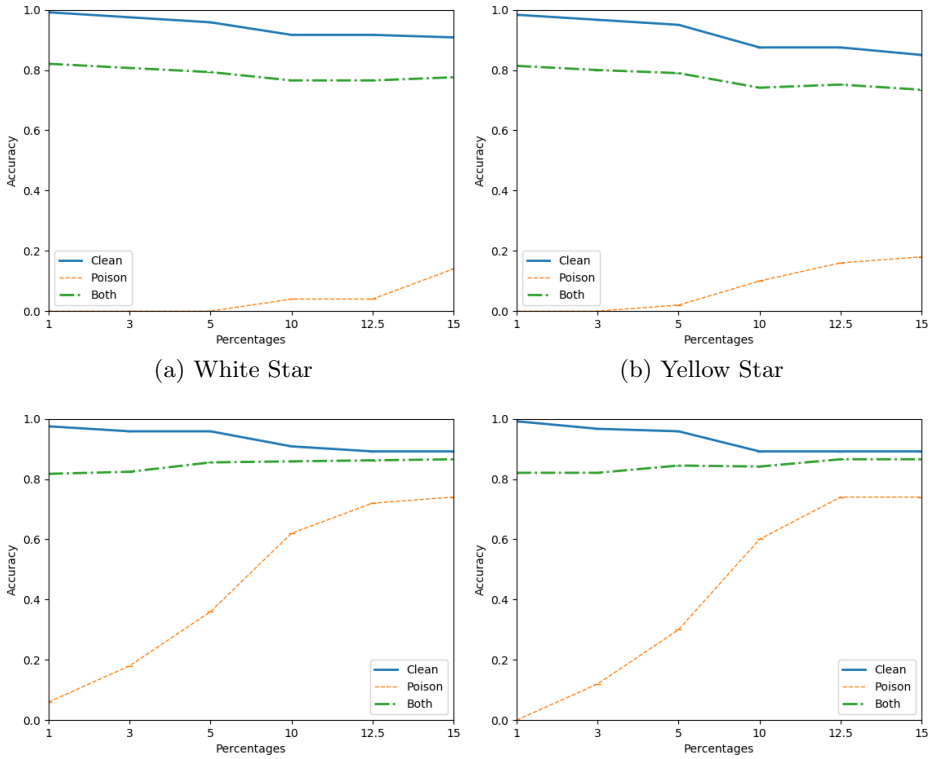


(g) Position Bottom, Yellow Block



(h) Position Top, Yellow Block

Fig. 4. Results for different positions of the backdoor key on the Belgian dataset: the left columns shows the results for the backdoor on the bottom, right on the top



(a) White Star (b) Yellow Star

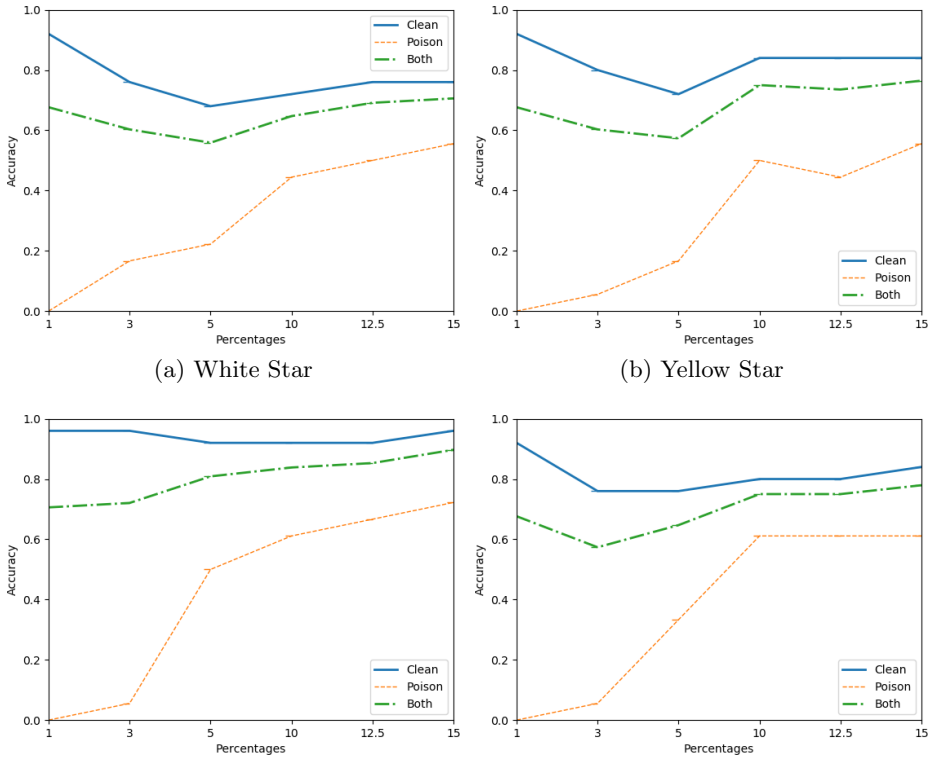
(c) White Block (d) Yellow Block

Fig. 5. Results for HOG feature extraction and SVM classifier on the German traffic sign dataset

performs significantly better, but it still does not achieve effectiveness levels we could obtain in the CNN case, as we plateau at around 75% correctness. Furthermore, we can observe, for both patterns, a quite noticeable drop in overall classification performance on the clean images, from almost 100% accuracy in the targeted class down to approximately 85%. This is a degradation that an attentive user of the model might notice, and that could thus lead to a suspicion of a potential attack.

The results on the Belgium dataset show a rather similar behaviour, with the differences of accuracy on the block and star pattern being a bit more prominent.

For the Chinese dataset on the other hand, there is much less discrepancy to be observed for the different types of patterns embedded, as can be seen from Figure 6. Indeed, all patterns, regardless of shape or colour, have a rather similar, steady increase in performance for the backdoor. However, again, the backdoor performance is below the values achieved for the CNN, as it peaks at or slightly above 60% of accuracy of attributing the poisoned images to the class



(a) White Star (b) Yellow Star

(c) White Block (d) Yellow Block

Fig. 6. Results for HOG feature extraction and SVM classifier on the Chinese traffic sign dataset

the attacker has intended. Except for the white block pattern, the performance of the clean samples in the targeted class is again significantly affected.

For the French dataset, only one of the pattern shows an accuracy of roughly 15% when using all available poisoned images in the training set (i.e. 15% poisoned images in that class), while all other patterns exhibit results at or very close to 0%. This is likely correlated with the relatively small size of this dataset, where the subsequently small number of backdoor patterns likely is not prominent enough to be adequately represented in the extracted features.

We therefore conclude that embedding backdoors in images analysed with the "traditional" approach of first feature extraction and a subsequent learning step is less successful than the attack on deep learning models like the convolutional neural networks, both in the terms of overall achievable accuracy, as well as in the reliability of the attack to work. A notable exception is only in the Chinese dataset, which has acceptable accuracy, but at the price of noticeable degradation of target class accuracy.

5 Conclusions and Future Work

In this paper, we performed a comparative analysis of poisoning (backdoor) attacks on image classification models. We selected a number of different datasets depicting traffic signs, the correct recognition of which could be part of tasks e.g. for autonomous driving. For each dataset, we analysed the susceptibility of the model towards manipulated images that should fool the classifier to trigger a specific, selected target class – categorising an important traffic sign that should lead to e.g. give-way situations with a less important one.

For most settings, the poisoning attacks are successful and the backdoor can be triggered with a high level of reliability, while the effects on the overall classification performance of the model are rather minor, and thus the attack is unlikely to be detected due to unusually low classification accuracy for clean data samples.

We further compared these results with choosing a more traditional approach for image classification, i.e. utilising a feature extraction step with a subsequent learning of a SVM model for classification. We observed that the attacks are far less successful in these settings. However, we still conclude that the approach based on feature extraction in combination with a "shallow" learning model is not immune against these types of attack, which are often mentioned to be effective mostly in the context of deep learning approaches.

Future work will focus on extending these experiments to more datasets, also from other domains, and an evaluation of the effectiveness of mechanisms that have been proposed to defend against these types of adversarial attacks.

Acknowledgments

The competence center SBA Research (SBA-K1) is funded within the framework of COMET — Competence Centers for Excellent Technologies by BMVIT, BMDW, and the federal state of Vienna, managed by the FFG.

References

1. Bagdasaryan, E., Veit, A., Hua, Y., Estrin, D., Shmatikov, V.: How to backdoor federated learning. CoRR **abs/1807.00459** (2018), `http://arxiv.org/abs/1807.00459`
2. Chen, X., Liu, C., Li, B., Lu, K., Song, D.: Targeted backdoor attacks on deep learning systems using data poisoning. CoRR **abs/1712.05526** (2017), `http://arxiv.org/abs/1712.05526`
3. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: international Conference on computer vision & Pattern Recognition (CVPR'05). vol. 1, pp. 886–893. IEEE Computer Society (2005)
4. Dalvi, N., Domingos, P., Sanghai, S., Verma, D., et al.: Adversarial classification. In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 99–108. ACM (2004)

5. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: A deep convolutional activation feature for generic visual recognition. In: Proceedings of the 31st International Conference on Machine Learning. pp. 647–655. Bejing, China (June 22–24 2014)
6. Eykholt, K., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., Song, D.X.: Robust physical-world attacks on deep learning visual classification. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (2018)
7. Gu, T., Dolan-Gavitt, B., Garg, S.: Badnets: Identifying vulnerabilities in the machine learning model supply chain. In: Proceedings of the Machine Learning and Computer Security Workshop. Long Beach, CA, USA (December 8 2017), http://arxiv.org/abs/1708.06733
8. Lowd, D., Meek, C.: Adversarial learning. In: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. pp. 641–647. ACM (2005)
9. Newsome, J., Karp, B., Song, D.: Paragraph: Thwarting signature learning by training maliciously. In: International Workshop on Recent Advances in Intrusion Detection. pp. 81–105. Springer (2006)
10. Paparoditis, N., Papelard, J.P., Cannelle, B., Devaux, A., Soheilian, B., David, N., Houzay, E.: Stereopolis ii: A multi-purpose and multi-sensor 3d mobile mapping system for street visualisation and 3d metrology. Revue Franaise Photogramm. Tldtection **200**(1), 69–79 (2012)
11. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical black-box attacks against machine learning. In: Proceedings of the 2017 ACM on Asia conference on computer and communications security. pp. 506–519. ACM (2017)
12. Pratt, L.Y.: Discriminability-based transfer between neural networks. In: Advances in Neural Information Processing Systems (NIPS). pp. 204–211 (1993)
13. Razavian, A.S., Azizpour, H., Sullivan, J., Carlsson, S.: CNN features off-the-shelf: an astounding baseline for recognition. In: IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 512–519 (June 2014). https://doi.org/10.1109/CVPRW.2014.131, http://arxiv.org/abs/1403.6382
14. Serna, C.G., Ruichek, Y.: Classification of traffic signs: The european dataset. IEEE Access **6**, 78136–78148 (2018)
15. Shen, S., Tople, S., Saxena, P.: A uror: defending against poisoning attacks in collaborative deep learning systems. In: Proceedings of the 32nd Annual Conference on Computer Security Applications. pp. 508–519. ACM (2016)
16. Stallkamp, J., Schlipsing, M., Salmen, J., Igel, C.: Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. Neural networks **32** (2012)
17. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. In: International Conference on Learning Representations. Banff, Canada (April 14–16 2014), http://arxiv.org/abs/1312.6199
18. Timofte, R., Zimmermann, K., Van Gool, L.: Multi-view traffic sign detection, recognition, and 3d localisation. Machine Vision and Applications **25**(3), 633–647 (Apr 2014). https://doi.org/10.1007/s00138-011-0391-3