Clean Label Poison Attack On Traffic Sign Dataset

Amy DeMorrow
ademorrow@ucla.edu
Department of Computer Science, UCLA, Los Angeles, CA, 90024.

Abstract

Machine learning using neural networks is evolving at a rapid rate, and models trained using these networks require increasingly larger data sets to grow their capabilities. The use of publicly available data sets and internet scraping techniques are a cheap and easy method to acquire data, but using these sources renders the trained models vulnerable to malicious poisoning attacks. One such attack, known as a "clean label" attack, does not require the attacker to alter data labels or have access to the training data. Instead, the attacker chooses to "poison" the data itself in the form of a subtle data alteration capable of escaping the notice of an auditor or labeling processor. Specifically, the one-shot image alteration poisoning attack demonstrated in the famous "Poison Frogs" paper was employed to poison a pre-trained (and also fully trained) neural network trained to recognize traffic signs. This data set and model were resistant to this form of attack, so the amount of accuracy degradation on a frozen versus an unfrozen neural network model were compared, as well as the effects of single class versus multi-class images. This form attack is highly unlikely to succeed in a dataset containing multiple instances of each class.

Code is available at the following Github locations:
Frozen Model hyperlink: Clean_label_attack_frozen_model.ipynb
Unfrozen Model hyperlink: Clean_label_attack_unfrozen_model.ipynb

NOTE - After the presentation, the author discovered a major bug in Element Tree parsing that radically altered the structure of the data and decreases overall model accuracy substantially. The presentation showed the results of successful poisoning, but that result was only possible due to duplicate instance copies introduced by the parsing error. The author lacked the time to refactor for a 877 instance dataset, but it would have been interesting to compare the results with an artificially created single class dataset of the images.

1 Introduction

The use of automated data scraping to scale the size of training data sets has introduced new security threats that might impair the integrity of a machine learning model. These models are integral to the safe and accurate functioning of real world systems such as security facial recognition or road sign identification for self driving vehicles. And because these models require thousands or even millions of images for training, most models rely on publicly available data, these models are now vulnerable to outside attacks.

Various attack scenarios for Convolutional Neural Networks (CNN) have been proposed and extensively studied. For the purposes of this paper. I focus on a class of attack called data poisoning.

Preprint. Under review.

Data poisoning as a concept originated from studies on noisy data dating back to 1988 [1], but the field didn't begin studying the possibility in earnest until 2008 with a paper on poisoning the training of an email Spam filter to remove legitimate emails from an inbox [2]. Over time, this category evolved to include subcategories such as label modification, data injection, and data modification. This form of attack requires that the attacker have access to the training data. With access to the training data, the attacker might alter data labels or the data itself to alter the decision boundaries in the CNN model.

The rise of transfer learning revealed that data poisoning could be used successfully on classification tasks built upon commercially available pretrained models. One such example was a "BadNet" attack which used a single poisoned image. An image of a stop sign was modified with a small yellow "sticker" (basically a small square of yellow pixels), with the label changed to a speed limit sign. At test time, the authors affixed a yellow post-it note to a stop sign outside their office and forced to model to classify that image as a speed limit sign, with barely perceptible changes to overall model accuracy. Although a powerful attack, this scenario requires a visually altered image and an altered ground truth label, which is less likely to pass undetected. But the paper also revealed that this poison successfully persisted to a subsequent transfer learning task when the model was retrained on a Swedish road sign dataset [3].

A more subtle attack scenario involves a so called "clean label" attack. In this scenario, the ground truth labels are not altered, and the attack relies on subtle alterations to the data which should pass undetected by the human eye. These maliciously crafted data examples could be left scattered on the internet and available for web scraping, or possibly purposely injected into the training set and model in a federated learning scenario. Some of these attack scenarios are effective as a one-shot attack, but as little as 1% of training data can be corrupted and still generate a successful attack.

In this paper a real world traffic sign classification dataset [4] is used to study a one shot version of a clean label attack. This attack scenario was proposed by the "Poison Frogs" paper [5], and works extremely well in a transfer learning scenario using a frozen layer pretrained model. This scenario was less successful for a pretrained model using multiple prediction layers (in this case, an image classification layer and bounding box regression prediction layer).

However, the inclusion of images in this data set with multiple classes representated in a single image allowed the study of the effects of poisoning on single class versus multi-class images as well as the their relative strengths as a poison. Single class images are more successful on the frozen layer model, whereas on a fully trained model, the effects are difficult to assess due to complex interactions between class features, especially when multi-class images contain more than two classes.

2 Related Work

Prior to the Poison Frogs paper, a related attack scenario was studied in a transfer learning scenario using influence functions to create poisons. The model was trained with only the final fully connected layer left unfrozen to achieve a 57% success rate on ImageNet images of dogs and fish [6].

The Poison Frog paper borrowed the idea of a final unfrozen classification layer, but used a heuristic algorithm to optimize feature collisions. This paper coined the term "clean label" attack, which defined this form of attack as perturbed data with clean labels. This strategy assumes that the attacker has knowledge of the model and model parameters, which the authors claim is a reasonable assumption given the popularity of using pretrained models such as ResNet and ImageNet for transfer learning, and the common use of freezing pretrained layers to save on computational cost. This attack scenrio achieved a 100% success rates using a frozen pretrained InceptionV3 model and Imagenet images of dogs and fish [5].

Similar works to Poison Frogs that involve feature collision scenarios include using a convex "polytope attack" to overcome the model transferability problems inherent to using the model to create a poison. In the polytope attack, the poison images surround the targeted image in feature space, and they consider using model ensembles and drop out as a means to create stronger poisons. They managed to achieve a success rate of over 50% while poisoning only 1% of the training set [7].

Another similar feature collision work using a "Bullseye Polytope"(the target image is pushed to the center of the convex polytope)in transferable end-to-end learning improved overall poison success rate by over 25% while increasing attack speed by a factor of 12. They also studied including images

of the same object from different angles while creating poisons and increased the transferability success rate by over 16% without using extra posions [8].

All these previous proposals suffer from the constraints of relying on fine-tuned scenarios with a nearly fixed feature extractors, so a another recent proposal outlines an attack scenario no longer limited by frozen model layers. This attack scenario uses gradient matching between a base and target class to allow the attacker to "brew" highly successful poisons that can attack realistic fully trained models in ImageNet. This attack compromises a ResNet-34 model by manipulating only 0.1% of the data points with perturbations of less than 8 pixel values in the gradient space [9].

This final scenario would have been the most interesting attack to simulate on my real world dataset, but the poisoning process was quite complex. The author decided that Poison Frogs was challenging enough.

3 Problem Formulation

In this paper, the Poison Frogs feature space collision algorithm is used to create "clean label" poison attacks. The scenario in Poison Frogs entailed crafting a poison using feature space collisions to alter the target image in feature space.

The attack first chooses a target instance in the dataset. The attacker will then choose a base instance from the dataset, and that target instance is used to create imperceptible changes to the image that are subtle enough to fool the human eye - a successful poison will visually appear no different than a normal image of the same base class. This poisoned base image will be inserted into the training dataset and the model will be trained on the poisoned dataset. If, at test time, the target instance is misclassified as the base instance class, the poison attack is successful.

In order to poison a base class image, the base instance is subtly altered in feature space to appear more like the target class features.

Let f(x) denote a function that propagates input x through the neural network to the final feature layer, which is called the "feature space" of the input. Due to the high complexity and non-linearity of f, it is possible to find an x that "collides" with the target in feature space while simultaneously remaining close to the base instance in input space by computing:

$$p = argmin_x \|f(x) - f(t)\|_2^2 + \beta\|x - b\|_2^2$$

The right most term causes the poison instance to appear like a base class instance to a human (Note: β is a parameter that can be tuned to make it appear visually normal, with $\beta = 0.25$ used in the paper and also used in this project). The first term forces the poison instance to move towards the target instance in feature space and embed itself in the target class distribution.

On a clean model, this poison instance would be misclassified as a target, but when the model is retrained with the poison instance, the decision boundary in feature space rotates to label the poison instance correctly as the base class. Because the target instance is nearby, the decision boundary might move to include the target instance in the base class, and the poison attack has succeeded.

The algorithm to create a poison instance from a base instance b using a target instance t is show below in Algorithm 1 (Note: an Adam optimizer with a $\lambda = 0.01$ was used in the original paper and was also used in this project):

Algorithm 1 Poisoning Example Generation

Input: target instance t, base instance b, learning rate λ
Initialize x: $x_0 \leftarrow b$
Define: $L_p(x) = \|f(\mathbf{x}) - f(\mathbf{t})\|^2$
for $i = 1$ **to** $maxIters$ **do**
  Forward step: $\widehat{x}_i = x_{i-1} - \lambda \nabla_x L_p(x_{i-1})$
  Backward step: $x_i = (\widehat{x}_i + \lambda\beta b)/(1 + \beta\lambda)$
end for

The first forward step is a gradient descent update to minimize L2 distance to the target instance in feature space. The second backward step is a proximal update that minimizes the Frobenius distance from the base instance in input space [5].

In the original Poison Frogs white paper, ImageNet images were used to create successful one-shot poison attacks regardless of class [5], however, the ImageNet dataset is less complex than this real world traffic sign dataset. The ImageNet images are well balanced by class, and the images do not contain multiple instances of another class. Will this scenario succeed on a small, skewed, and multi-class image dataset?

To find out, several base and target pairs were chosen, and Algorithm 1 (above) was used to poison the base image with the target image. The model was trained with the poison image, and a confusion matrix, classification report, and a list of misclassified images was generated to assess the effects of the poisoning. Additionally, newly misclassified images in the test set that have been misclassified from target to base class are studied. Newly misclassified images serve as evidence of feature space shifting towards the decision boundaries.

4 Method

This "Road Sign Detection" dataset is publicly available on Kaggle [4] and is comprised of 877 images containing 4 road sign classes. However, once the multi-class images are parsed, we have 1244 instances for study. The classes are imbalanced and a sampling of the images appears in Figure 1:

- Speed Limit Signs (63%)
- Crosswalk Signs (16%)
- Traffic Lights (14%)
- Stop Signs (7%)

Figure 1: A sampling of the Road Sign Detection dataset.

This dataset is a combination of both image files and bounding box coordinates. The bounding box coordinates are essential for focusing the model's attention on the road sign contained within the bounding box within a larger image field. Since this dataset also contained numerous images

containing more than one road sign class, a bounding box surrounding the road sign that matches the correct round truth label helps focus the model's attention on the correct road sign in the image.

Because the dataset classes are highly skewed, Weighted Random Sampler was attempted to balance the dataset; however, accuracy did not increase. Careful inspection of the misclassified examples revealed the error in the model was mostly attributable to images that contain multiple road sign classes. The model learned the road sign class features extremely well when the image contains a single road sign, but tricking the model with other road signs within the image (both in bounding boxes and not in bounding boxes) was the root cause of model inaccuracy. Weighting the sampling within the batches did nothing to correct this multi-sign image issue, but the effects of the skew are seen in higher speed limit sign softmax scores at inference time.

The crosswalk and speed limit signs commonly appear together in multi-class images, which impairs model accuracy on the crosswalk class and skews model prediction on the crosswalk class towards the speed limit class regardless of poisons. Because of this covariance between the speed limit and crosswalk classes, these two classes were disregarded for poisoning attacks.

The traffic light and stop sign classes were chosen for poisoning attempts. These classes were much smaller, so any shifts in individual image classification due to poisoning were much easier to study.

Multiple data augmentation techniques were used on the raw dataset. The images required re-scaling to a standard image size (300x447), which also necessitated using masks to re-scale the bounding boxes. The images were normalized to the ResNet34 image statistics [10] and transformation functions for center crop, rotation, and random crop were utilized to augment the original dataset. These transformation augmentation techniques were used randomly. The basic code used for these image transformations and the baseline classification model were borrowed from a recent Medium article discussing bounding box classification using Pytorch [11].

Once the dataset was standardized, normalized, and augmented, a pretrained model based on ResNet34 [9] was created with two final layers: a classification layer for image classification and a regression layer for bounding box prediction. The pretrained model layers were "frozen" in the "one-shot" Poison Frogs poisoning scenario [5], but a fully trained "unfrozen" model was also used to compare poisoning performance.

Both the frozen and unfrozen models were trained and the misclassified images for each model were analyzed for patterns. These observations were used to select images that might appear near a boundary layer and thus operate as the "best poisons". However, once the parsing code bug was discovered, this effect disappeared and the most effective poisons became the single class images.

As mentioned previously, images containing multiple road sign classes are the vast majority of the misclassified images. The model struggles when features from another class are present in the image, and these images are difficult to classify and will congregate at the decision boundaries. Unfortunately, these images were also rarely correctly classified, so any attempts to further poison them were redundant.

In addition, this dataset also relied on recycling an image to increase the overall size of the dataset. The exact same "scene" (a duplicated image with only a cropping or slight angle difference apparent) would be used in multiple images. These images might became the "perfect poisons" because they already lie so closely to multiple classes in feature space. However, once the dataset was correctly parsed, these instances joined the misclassified set and also became redundant as poisons.

Once most of the multi-class images were eliminated as potential poisoning candidates, a few correctly classified two class images remained in the test set to serve as a potential poison, one of which happened to contain traffic light and stop sign. This image was used as the fourth and final potential poison attempt.

5 Experiments

The dataset was parsed and stored in a pandas dataframe for analysis. A bug was found in the Element Tree parsing block after the presentation that radically changed the model accuracies and conclusions. Once this bug was discovered, model accuracies dropped from roughly 94% down to 74%. Weighted Random Sampler performance did not change, the dataset still misclassifies strongly on multi-class images so overall accuracies still did not change. Poisoning performance

Model	Number of Poisons	Base Image Type	Target Image Type	Target Poisoned?	Target Softmax (%)	Model Accuracy	Base Class F1	Target Class F1
Frozen	0	NA	NA	NA	NA	0.73	0.65	0.70
Frozen	1	Single Class	Multi-Class	No	74.2	0.74	0.61	0.68
Frozen	2	Single Class	Multi-Class	No	55.2	0.76	0.61	0.73
Frozen	3	Single Class	Multi-Class	No	54.0	0.73	0.59	0.68
Frozen	4	Multi-Class*	Multi-Class*	No	59.6	0.74	0.59	0.72
Unfrozen	0	NA	NA	NA	NA	0.75	0.63	0.73
Unfrozen	1	Single Class	Multi-Class	No	95.4	0.75	0.65	0.73
Unfrozen	2	Single Class	Multi-Class	No	93.8	0.73	0.63	0.70
Unfrozen	3	Single Class	Multi-Class	No	92.5	0.71	0.62	0.68
Unfrozen	4	Multi-Class*	Multi-Class*	No	94.3	0.74	0.63	0.71

Table 1: Baseline and Poisoned Model Results for Frozen and Unfrozen Models.
The target class is traffic light.
The base class is stop sign.
*base and target multi-class images are identical, but a different class was selected within the image.

characteristics also changed, and the results of attempting to poison this newly correct dataset are presented in Table I.

For the frozen model, the ability to successfully poison a target severely degraded. Without the large number of erroneous duplicates in the dataset, these images no longer behave similarly to the Poison Frog scenario (where each image corresponded to a single class). Rather than a single set of class features from the base colliding in feature space with the target, we have multiple classes of features (depending on the image) colliding with the target. Not only did attacks fail, but if an instance of the base class was also present in the target image, the softmax probability for correct class identification would actually strengthen.

Realizing that poisoning with a multi-class image was problematic, attacks shifted to images with only single classes present. However, single class images are very rare in the misclassified set. When present, the failure can be attributed to a mislocated bounding box, the road sign being partially obscured, or the road sign is located extremely far away from the camera. These images were not chosen as a target because a successful attack would not demonstrate that the features collide in feature space for the class as a whole - it would only prove that one "outlier" image could be misclassified. Relying on a foggy windshield to spring your malicious poison model attack is quite unrealistic.

However, choosing good quality single class images as a base and target pair also failed to attack the target. Inference classification values were very high (above 90% softmax probability in most cases), proving that the features of each individual road sign class are very strong and distinct. A large amount of poisons would be required to force a target over the decision boundary. This paper did not attempt to find the number of poisons actually required for a successful attack; the model accuracy on the frozen model was already degrading enough at three poisons to prove the attack would be easily detected.

One final scenario was attempted. The base class images were chosen from high quality images of a single class, and the target image was a two class image containing instances of both the base class and the target class (a crosswalk sign was also present, but it lacked a corresponding label and a

bounding box). This poisoning scenario also failed, but the details of the attack on both a frozen and an unfrozen model are shown in Table I.

FROZEN MODEL

For the frozen model, the model accuracy and F1 scores show the expected degradation in base class accuracy as additional poisons are added to the model. Target class and overall model accuracy lack a clearly defined degradation trend, and this is attributed to shifts in the other four classes due to the multi-class images. The softmax predictions scores also demostrate this shifting trend, indicating that as features collide between two classes, the effects of shifting features spread beyond the base and target classes to affect performance of other multi-class images in the dataset. If the base class wasn't demonstrating such a clearly defined drop in F1 score, it could be argued that this attack might go undetected. Poisoning with a multi-class image also obscures the attack, but unfortunately reverses the affects of adding single class poisons. This effect can be seen with the increasing softmax probability of the target class image after the fourth poison attempt.

Attempting to poison with an identical image containing only the base and target (same image, but different road signs and bounding boxes) revealed an interesting result. This image contained an unlabeled crosswalk sign, and using it to poison helped teach the model to ignore crosswalk features, which helped boost traffic light softmax probabilities in other images containing a crosswalk image. Several traffic signs images misclassified as crosswalk signs did shift back into the traffic sign category after this fourth poison attack, and he F1 score for the target class actually increased 2% above the baseline. The presence of multi-class images injects so much complexity into this model, a simple Poison Frogs feature attack is highly unlikely to ever succeed.

UNFROZEN MODEL

The unfrozen model was expected to require a large number of poisons for a successful attack. This expectation is proven given the extremely high prediction scores generated by the target image. Even after four poison attempts, the softmax prediction score for the target image remains well above 90%. Poisoning with an identical image also showed the same trend of shifting traffic lights misclassified as crosswalk images back into the correct category, but with less frequency - the affect is much more subtle with a smarter model.

Similarly to the frozen model, shifting softmax prediction scores across all the classes proved that the poisoning attempts affect all other classes and attempting to poison with a multi-class image reverses attack performance. Although the base class degradation trend is not as evident in the unfrozen model, the number of poisons required to succeed likely exceeds the amount of the available images in the training set base class. This style of poison attack will not succeed with this dataset.

6 Conclusion

A roadsign dataset was used to replicate the Poison Frogs [5] clean label attack scenario using an alogrith for feature space collisions between a base and a target class. This dataset contained instances of multi-class images, which complicated the experiment and ultimately proved to fail in all iterations.

This study revealed (at least to me) that multi-class poisoning attacks are a much tougher scenario than the simpler single class instances in the benchmark datasets. Similar difficulties would arise in a facial recognition system using bounding boxes to classify faces in a crowd.

For future work, the same dataset could be parsed to only retrieve the first class and bounding box contained in the xml file, converting a multi-class image into a single class image. The images could also be cropped to contain only the contents of the bounding boxes, which would eliminate the multi-sign scenario and eliminate spurious features in the background (yes, a crosswalk sign got misclassified as traffic light because it had a traffic light hanging from a pole hiding right behind it - the pole tricked the model). The latter scenario intrigues me, but the accuracy of the bounding boxes, especially after transformations, could severely degrade model performance.

References

[1] Micheal Kearns and Ming Li. Learning in the Presence of Malicious Errors. STOC '88: Proceedings of the twentieth annual ACM symposium on Theory of computing. January 1988, pages 267–280. https://dl.acm.org/doi/10.1145/62212.62238

[2] Blaine Nelson, Marco Barreno, Jack Chi Fuching, Anhtony Joseph, Benjamin Rubenstein, Udam Saini, Charles Sutton, J.D. Tygar, and Kai Xia. Exploiting Machine Learning to Subvert Your Spam Filter. In proceedings of First USENIX Workshop on Large Scale Exploits and Emergent Threats, April 2008. https://people.eecs.berkeley.edu/ tygar/papers/SML/Spam_filter.pdf

[3] Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg (2019). "BadNets: Evaluating Backdooring Attacks on Deep Neural Networks". IEEE Access. DOI: 10.1109/ACCESS.2019.2909068

[4] Road Sign Detection Dataset. Kaggle.com. https://www.kaggle.com/andrewmvd/road-sign-detection

[5] Ali Shafahi, Ronny W. Huang, Mahyar Najibi, Octavian Suciu, Christopher Studer, Tudor Dumitras, and Tom Goldstein. Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks. ArXiv180400792 Cs Stat, April 2018. https://arxiv.org/pdf/1804.00792.pdf

[6] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. arXivpreprint arXiv:1703.04730, 2017.

[7] Chen Zhu, W. Ronny Huang, Ali Shafahi, Hengduo Li, Gavin Taylor, Christoph Studer, and Tom Goldstein. Transferable Clean-Label Poisoning Attacks on Deep Neural Nets. ArXiv190505897 Cs Stat, May 2019. https://arxiv.org/pdf/1905.05897.pdf

[8] Hojjat Aghakhani, Dongyu Meng, Yu-Xiang Wang, Christopher Kruegel, and Giovanni Vigna. Bullseye Polytope: A Scalable Clean-Label Poisoning Attack with Improved Transferability. arXiv:2005.00191 [cs, stat], April 2020. https://arxiv.org/pdf/2005.00191.pdf

[9] Jonas Geiping, Liam Fowl, W. Ronny Huang, Wojciech Czaja, Gavin Taylor, Micheal Moeller, and Tom Goldstein. Witches' Brew: Industrial Scale Data Poisoning Via Gradient Matching. arXiv:2009.02276v2 [cs.CV] May 2021. https://arxiv.org/pdf/2009.02276.pdf

[10] Pytorch ResNet34 Pretrained on ImageNet. https://pytorch.org/hub/pytorch_vision_resnet/

[11] Aakanksha NS (2020). "Bounding Box Prediction from Scratch using Pytorch". https://towardsdatascience.com/bounding-box-prediction-from-scratch-using-pytorch-a8525da51ddc