Overfitting is all you need?

Bosch's Model Extraction Attack For Video Classification
Group-3

***Abstract*—Deep learning models have been applied in a variety of day-to-day applications as a result of enormous developments in machine learning over the last few years. Attacks on such models employing perturbations, particularly in real-world circumstances, represent a significant obstacle to their application, prompting research to focus on improving the resilience of these models. The dependability of such models has been studied primarily in two aspects: white-box, where the adversary has access to the targeted model and related parameters, and black-box, which mirrors a real-life scenario with the adversary having essentially little information of the model to be attacked. It is critical to discover, research, and implement countermeasures against such assaults in order to offer comprehensive security coverage.**

***Index Terms*—BlackBox, GreyBox, victim, advesary**

I. Key assumptions and threat model

One of the key aspects that makes black box setting extremely practical and relevant to real life model security analysis is that we assume we have no knowledge of the victim. This includes:

- Model Architecture
- Model Weights
- Model Logits
- Class labels
- Any information about inter class correlations

While we can easily understand the problems caused by point 1 and 2 and we are more interested in challenges posed by the latter points as solving them, essentially holds the key to Black Box model extraction.
Model logits might seem similar to class labels at first glance, there is one key difference. Logits are (mostly) not orthogonal for two different class samples while class labels are. Hence, we have no way knowing class correlations (similar classes activate same neurons!)
With unknown class labels, we emphasize the fact that we should not be able to generate class data from external sources. Essentially, for us, the model will output an index and we have to work with that.
Inter class correlations can be beneficial in a meta-learning setting and this knowledge can be exploited to increase the performance of the extracted model. However it is not practical in a black box setting.

We will address all these problems with a novel approach that works with all the given constraints in black box setting.

II. Methodology approach

We propose a 3D Unet architecture that will take noise as input, with shape (f,w,h,c) and intends to transform it into a similar shaped tensor, that when given to the victim, will always return the same label. Essentially we learn to convert random noise into a sample of the class. Figure 1 depicts the nature of class synthesizer network.

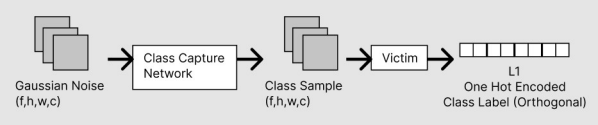

Fig. 1. Class synthesizer network

The generated class sample is then shown to the victim model and the output top-1 label is one hot encoded, this is called L1

L1 is then passed to another auto encoder network that attempts to capture the class correlation by exploiting a special property of auto encoders, which is elaborated later. Figure 2 shows the correlation capture model. Output of this network has the same shape as L1 and is called L2.

The 2 logits are combined as $L1 + \lambda L2$ where lambda is experimentally determined to be a small value (1e-2). This helps control the influence of second network as we want the main class label to be still prominent.

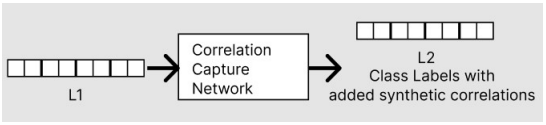

Fig. 2. Class correlation synthesizer network

The advantage of adding a second network into the mix is that it helps solve the issue of orthogonality between target labels and predicted labels and enables us to use conventional classification losses like cross-entropy which rely on the two vectors not being orthogonal. The loss is then individually back propagated through both the networks separately, which enables them to learn independently.
We also want the Class synthesizer network to over fit by showing it seeded noise over the same epoch. This will essentially create n samples depending on how many samples we use per epoch.

We then repeat the same to generate samples for all the classes.

III. Conclusion

We could provide proof of concept for a small fraction of the data set having only 10 classes. However, this approach requires making of customized data generators in the direction

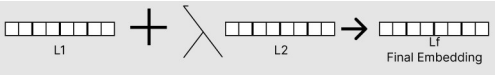

Fig. 3. Final class logit formulation

of one particular class features. Due to size of the problem data set (400 and 600 classes), our infrastructure was not able to converge to significant results. The approach is reproducible to good results as seen in the proof of concepts. For Action Classification attacking on Swin-T Model, our blackbox and greybox produced 0.25% accuracy and for Video Classification attacking on MoViNet-A2-Base Model, our blackbox and greybox produced 0.167% accuracy.