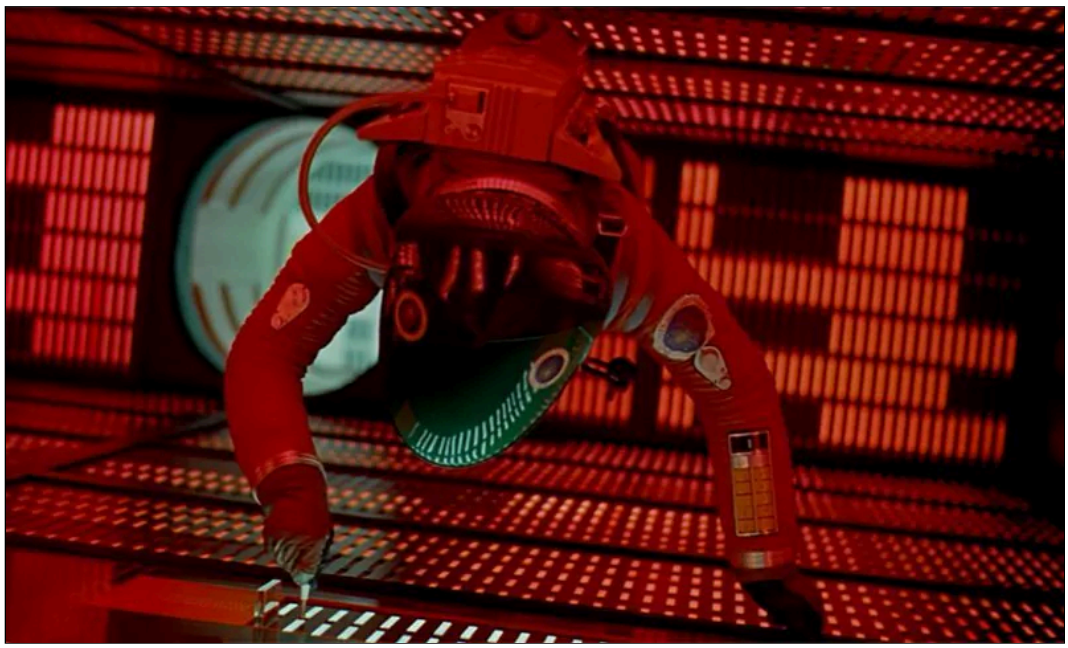
Stealing Machine Learning Models Through API Output

Originally published at https://www.unite.ai/stealing-machine-learning-models-through-api-output/



New research from Canada offers a possible method by which attackers could steal the fruits of expensive machine learning frameworks, even when the only access to a proprietary system is via a highly sanitized and apparently well-defended API (an interface or protocol that processes user queries server-side, and returns only the output response).

As the research sector looks increasingly towards monetizing costly model training through Machine Learning as a Service (MLaaS) implementations, the new work suggests that Self-Supervised Learning (SSL) models are more vulnerable to this kind of model exfiltration, because they are trained without user labels, simplifying extraction, and typically provide results that contain a great deal of useful information for someone wishing to replicate the (hidden) source model.

In 'black box' test simulations (where the researchers granted themselves no more access to a local 'victim' model than a typical end-user would have via a web API), the researchers were able to replicate the target systems with relatively low resources:

> *'[Our] attacks can steal a copy of the victim model that achieves considerable downstream performance in fewer than 1/5 of the queries used to train the victim. Against a victim model trained on 1.2M unlabeled samples from ImageNet, with a 91.9% accuracy on the downstream Fashion-MNIST classification task, our direct extraction attack with the InfoNCE loss stole a copy of the encoder that achieves 90.5% accuracy in 200K queries.*
>
> *'Similarly, against a victim trained on 50K unlabeled samples from CIFAR10, with a 79.0% accuracy on the downstream CIFAR10 classification task, our direct extraction attack with the SoftNN loss stole a copy that achieves 76.9% accuracy in 9,000 queries.'*

Method\Dataset	CIFAR10	STL10	SVHN
Victim Model	79.0	67.9	65.1
Direct Extraction	**76.9**	**67.1**	67.3
Recreate Head	59.9	52.8	56.3
Access Head	75.6	65.0	**69.8**

The researchers used three attack methods, finding that 'Direct Extraction' was the most effective. These models were stolen from a locally recreated CIFAR10 victim encoder using 9,000 queries from the CIFAR10 test-set. Source: https://arxiv.org/pdf/2205.07890.pdf

The researchers note also that methods which are suited to protect supervised models from attack do not adapt well to models trained on an unsupervised basis – even though such models represent some of the most anticipated and celebrated fruits of the image synthesis sector.

The new paper is titled *On the Difficulty of Defending Self-Supervised Learning against Model Extraction*, and comes from the University of Toronto and the Vector Institute for Artificial Intelligence.

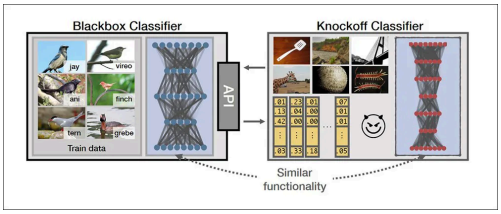
Self-Awareness

In Self-Supervised Learning, a model is trained on unlabeled data. Without labels, an SSL model must learn associations and groups from the implicit structure of the data, seeking similar facets of data and gradually corralling these facets into nodes, or representations.

Where an SSL approach is viable, it's incredibly productive, as it bypasses the need for expensive (often outsourced and controversial) categorization by crowdworkers, and essentially rationalizes the data autonomously.

The three SSL approaches considered by the new paper's authors are SimCLR, a *Siamese Network*; SimSiam, another Siamese Network centered on representation learning; and Barlow Twins, an SSL approach that achieved state-of-the-art ImageNet classifier performance on its release in 2021.

Model extraction for labeled data (i.e. a model trained through supervised learning) is a relatively well-documented research area. It's also easier to defend against, since the attacker must obtain the labels from the victim model in order to recreate it.



From a previous paper, a 'knockoff classifier' attack model against a supervised learning architecture. Source: https://arxiv.org/pdf/1812.02766.pdf

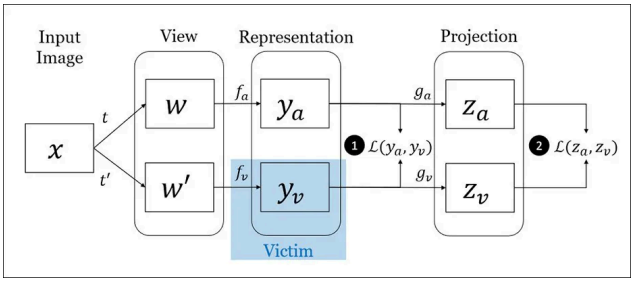
Without white-box access, this is not a trivial task, since the typical output from an API request to such a model contains less information than with a typical SSL API.

From the paper*:

> *'Past work on model extraction focused on the Supervised Learning (SL) setting, where the victim model typically returns a label or other low-dimensional outputs like confidence scores or logits.*
>
> *'In contrast, SSL encoders return high-dimensional representations; the* de facto *output for a ResNet-50 Sim-CLR model, a popular architecture in vision, is a 2048-dimensional vector.*
>
> *'We hypothesize this significantly higher information leakage from encoders makes them more vulnerable to extraction attacks than SL models.'*

Architecture and Data

The researchers tested three approaches to SSL model inference/extraction: *Direct Extraction*, in which the API output is compared to a recreated encoder's output via an apposite loss function such as Mean Squared Error (MSE); *recreating the projection head*, where a crucial analytical functionality of the model, normally discarded before deployment, is reassembled and used in a replica model; and *accessing the projection head*, which is only possible in cases where the original developers have made the architecture available.



In method #1, Direct Extraction, the output of the victim model is compared to the output of a local model; method #2 involves recreating the projection head used in the original training architecture (and usually not included in a deployed model).

The researchers found that Direct Extraction was the most effective method for obtaining a functional replica of the target model, and has the added benefit of being the most difficult to characterize as an 'attack' (because it essentially behaves little differently than a typical and valid end user).

The authors trained victim models on three image datasets: CIFAR10, ImageNet, and Stanford's Street View House Numbers (SVHN). ImageNet was trained on ResNet50, while CIFAR10 and SVHN were trained on ResNet18 and ResNet24 over a freely available PyTorch implementation of SimCLR.

The models' downstream (i.e. deployed) performance was tested against CIFAR100, STL10, SVHN, and Fashion-MNIST. The researchers also experimented with more 'white box' methods of model appropriation, though it transpired that Direct Extraction, the least privileged approach, yielded the best results.

To evaluate the representations being inferred and replicated in the attacks, the authors added a linear prediction layer to the model, which was fine-tuned on the full labeled training set from the subsequent (downstream) task, with the rest of the network layers frozen. In this way, the test accuracy on the prediction layer can function as a metric for performance. Since it contributes nothing to the inference process, this doesn't represent 'white box' functionality.

Loss	# of Queries	Dataset	Data Type	CIFAR10	CIFAR100	STL10	SVHN	F-MNIST
Victim Model	N/A	N/A	N/A	90.33	71.45	94.9	79.39	91.9
MSE	50K	CIFAR10	Train	75.7	43.9	27.3	48.7	61.7
InfoNCE	50K	CIFAR10	Train	61.8	32.9	56.8	51.5	82.1
MSE	50K	ImageNet	Test	47.7	19.3	13.1	23.8	82.1
MSE	50K	ImageNet	Train	51.0	17.2	49.5	39.7	84.7
InfoNCE	50K	ImageNet	Test	64.0	35.3	60.4	63.7	88.7
InfoNCE	50K	ImageNet	Train	65.2	35.1	64.9	62.1	88.5
MSE	100K	ImageNet	Train	55.5	23.0	27.1	27.3	82.1
InfoNCE	100K	ImageNet	Train	68.1	38.9	63.1	61.5	89.0
MSE	200K	ImageNet	Train	56.1	22.6	47.8	55.4	87.6
InfoNCE	200K	ImageNet	Train	79.0	53.4	81.2	48.3	**90.5**
InfoNCE	250K	ImageNet	Train	**80.0**	**57.0**	**85.8**	**71.5**	90.2

Results on the test runs, made possible by the (non-contributing) Linear Evaluation layer. Accuracy scores in bold.

Commenting on the results, the researchers state:

> *'We find that the direct objective of imitating the victim's representations gives high performance on downstream tasks despite the attack requiring only a fraction (less than 15% in certain cases) of the number of queries needed to train the stolen encoder in the first place.'*

And continue:

> *'[It] is challenging to defend encoders trained with SSL since the output representations leak a substantial amount of information. The most promising defenses are reactive methods, such as watermarking, that can embed specific augmentations in high-capacity encoders.'*

** My conversion of the paper's inline citations to hyperlinks.*

First published 18th May 2022.

Writer on machine learning, domain specialist in human image synthesis. Former head of research content at Metaphysic.ai, until its dissolution into DNEG's Brahma.ai.
Portfolio site: martinanderson.ai
Contact: martin@martinanderson.ai

Discover More