GLOBAL RISK INSTITUTE

FINANCIAL INNOVATION SERIES

Adversarial Machine Learning: Risks and Opportunities for Financial Institutions

Author: Alexey Rubtsov,
PhD, Senior Research Associate, Global Risk Institute

March 2022

INTRODUCTION

In a recent study, Microsoft found that 25 out of 28 firms surveyed did not have protections against the so-called adversarial attacks on their machine learning (ML)-based systems.* One of the banks surveyed responded:

"[We] want to protect client info, employee info used in ML models, but we don't have a plan in place."

Adversarial ML is concerned with malicious attacks against ML models. The main goal of adversaries is to trick machine learning models by providing specialized, deceptive inputs that purposely confuse an ML model.

To illustrate the severity of threats from adversarial attacks, consider the following example from Eykholt et al. (2018), which shows that adding stickers to a "Stop" sign in a particular way, made the ML algorithm interpret the sign as "Speed Limit 45" (see Figure 1).

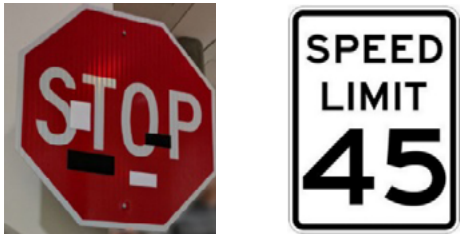

Figure 1: "Stop" sign that is interpreted by an ML algorithm as "Speed Limit 45" sign

Of note is that even with the stickers, the "Stop" sign would be unlikely to confuse a human driver and almost certainly would not be interpreted as a "Speed Limit 45" sign.

There are examples of this malicious practice that are imperceptible to the human eye. For instance, a photo classified as Panda with 57.7% confidence by an ML algorithm gets classified as Gibbon with 99.3% confidence after an imperceptible image modification (Figure 2, see Goodfellow et al. (2015)).

Figure 2: Modification to a photo, imperceptible to the human eye, can still fool algorithms.

This threat is of high importance to financial institutions that have been rapidly applying ML algorithms in their businesses. As we will discuss in this paper, adversarial attacks can occur in such areas as trading, fraud detection, robo-advising, and all applications of natural language processing and sentiment analysis of textual information in finance.

There are two branches of research in Adversarial ML. One branch develops new attacks to defeat existing ML algorithms, whereas the other branch explores techniques that make ML algorithms robust to defending against adversarial attacks. It is important to emphasize that Adversarial ML can also be used for good purposes. For example, it has been used to address problems of bias or to create techniques to generate synthetic data that are indistinguishable from the real data (i.e., "data anonymization").

In the following sections, we discuss both branches of Adversarial ML and their applications in addressing a few challenges in finance. We also provide a list of key questions that should be addressed by financial institution risk management teams who want to employ ML-based algorithms in their businesses.

* Kumar, R.S.S., Nyström, M., Lambert, J., Marshall, A., Goertzel, M., Comissoneru, A., Swann, M., Xia, S.: Adversarial Machine Learning -- Industry Perspectives (2021). https://arxiv.org/abs/2002.05646v3

ATTACKS ON ML IN FINANCE

Although Adversarial ML is a general concept that concerns all applications of ML, the following attacks are relevant to financial services:

- **Trading:** Trigger an ML trading algorithm to make wrong trading decisions based on adversely manipulated input information such as exchange data, market indicators, social media indicators, etc. (see Faghan et al. (2020)). The manipulated information used in the attacks could be almost undetectable by human traders.
- **Fraud detection:** Alter a pattern of illegal financial operations to prevent it from being recognized by ML fraud detection systems.
- **Robo-advising:** Manipulate robo-advising systems to provide wrong advice. One of the fintech firms in Microsoft's 2020-2021 survey wrote, "We use ML systems to suggest tips and financial products for our users. The integrity of our ML system matters a lot. [We are] worried about inappropriate recommendations like [the] attack on Tay."†
- **Natural Language Processing (NLP):** Manipulate applications of NLP and sentiment analysis of textual information (analysts reports, earning calls, etc.). It was pointed out in Morris et al. (2020) that replacing one word, in a text, by its synonym could make an NLP engine change the assessed sentiment from 99% Positive to 100% Negative!

TYPES OF ATTACKS

Attacks on ML can be implemented regardless of whether the attacked model is known to the attacker, i.e., a white-box attack, or where only the model's outputs are available, i.e., a black-box attack. Although black-box attacks might seem challenging to implement, they are not uncommon. Papernot et al. (2017) were able to attack a remotely located ML-based algorithm where they could send inputs to the algorithm and observe its responses. Neither the training data nor the algorithm's details were available to the attackers. The strategy was to build a local ML model that would produce the same outputs for the inputs sent to the algorithm, i.e., create a local model that acts as the remotely located algorithm. The most interesting discovery was that most adversarial examples that fooled the locally created model also fooled the remotely located ML algorithm. The researchers demonstrated the general applicability of their strategy to many ML techniques by conducting the same attack against models hosted by Amazon and Google.

There are three types of attacks in Adversarial ML: poisoning, evasion, and extraction attacks.

Poisoning attack: an attack where an adversary aims to influence the data used in training or re-training the algorithm. Contaminated data is fed into the algorithm and causes the machine to learn the wrong way. For example, on March 23, 2016, Microsoft launched its ML bot called Tay that was supposed to learn from communications on Twitter. However, in less than 24 hours of conversation, Tay learned wrong behavior and started uttering unethical tweets such as "Hitler was right. I hate the Jews". Microsoft claimed Tay had been "attacked" by trolls (i.e., people who post inflammatory or off-topic messages).‡

Evasion attack: an attack which causes an ML algorithm to misclassify the information fed into the algorithm. Adversaries attempt to evade detection by obfuscating the information presented to an already trained algorithm. The stop sign with stickers in Figure 1 is an example of this attack.

Extraction attack: an attack that involves an adversary probing an ML system to either reconstruct the model (model stealing attack) or extract the data on which the model was trained (inference attack). This attack is of particular concern when the training data are sensitive, or the model is confidential. Model stealing attacks can be used to steal a proprietary model (i.e., loss of intellectual property), which the adversary could use for their own financial benefit.

† Tay was an AI bot launched by Microsoft on March 23, 2016. Tay was supposed to learn how to communicate with people based on its interactions in Twitter. However, after being attacked by adversaries Tay learned unethical communication patterns and was shut down by Microsoft after 24 hours of operation.

‡ The racist hijacking of Microsoft's chatbot shows how the internet teems with hate. The Guardian, March 29, 2016

DEFENSE AGAINST ATTACKS

There is no one-size-fits-all defense against adversarial attacks. The development of reliable defence techniques is an evolving area of ML. The two most common defense techniques are:

Adversarial training: Data scientists generate many adversarial examples and train an algorithm not to be fooled by them. It is unrealistic to presume we can generate all possible adversarial examples. Thus, this approach is only partially effective in preventing adversarial attacks.

Defensive distillation: We make the algorithm less sensitive to changes in the input information. In other words, to fool an algorithm, adversaries would have to substantially alter input information, thereby making it easier to detect. For instance, referencing the stop sign example in Figure 1, to deceive the algorithm protected by "defensive distillation" an adversary would have to use very large stickers.

Financial Institutions should also consider non-technical defenses. For example, to make a model- stealing attack more difficult to implement, one can limit the number of requests that can be submitted to the remotely located model. This will make it more challenging for an adversary to obtain enough data to replicate the model.

OPPORTUNITIES WITH ADVERSARIAL ML

Adversarial ML can be used to improve the robustness of existing ML. Assume we have two ML algorithms: one is a "forger" and the other one is a "forgery detector". The goal of the forger is to trick the forgery detector by generating fake data and mixing these data with the real data, whereas the goal of the forgery detector is to discriminate between fake and real data. By making both algorithms compete and learn over time, it is expected that the forger will be able to produce data that the forgery detector will not be able to distinguish from the real data.

Below is an output from an ML algorithm trained using a competitive Adversarial ML to improve its capabilities. The forger algorithm generates fake photos of people, and the forgery detector algorithm discriminates between photos of real people and those generated by the forger model. Figure 3 illustrates some realistic photos that the forger algorithm eventually generates.

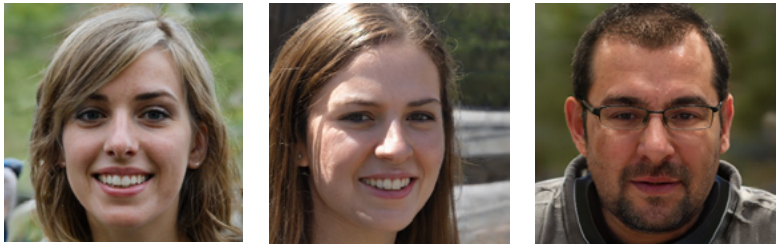

Figure 3: *Fake photos of people generated by Adversarial Learning (Source: www.thispersondoesnotexist.com)*

Financial services applications of such approaches include fairness in ML and synthetic data generation.

FAIRNESS OF ML ALGORITHMS

Edwards and Storkey (2016) use Adversarial ML to address the problem of fairness in ML. They define a decision as fair if it does not depend upon sensitive variables such as gender, age, or race. Removing the sensitive variable from the data may not work if there is any correlation between the sensitive variable and the other variables (i.e., one can still infer the sensitive variable from the other variables). Assume we want to train an algorithm to make credit decisions independent of race. The goal of the forger is to generate fake data that:

- *are indistinguishable from the real data,*
- *independent of race, and*
- *allow for accurate credit decisions.*

Fake data that satisfy the three criteria above are generated with the help of the forgery detector that tries to figure out the race from the generated fake data.

SYNTHETIC DATA GENERATION

Synthetic (or fake) data, indistinguishable from the real data, are important due to the following:

- *privacy concerns that limit the use of data and their sharing with third parties,*
- *requirements for large amounts of data that may be unavailable (e.g., development of hedging strategies using Reinforcement Learning)[§],*
- *modelling tail events in risk management (e.g., value-at-risk estimation, stress testing), and*
- *generating scenarios of future asset prices to construct optimal portfolios (see Mariani et al. (2019)).*

Takahashi et al. (2019) apply Adversarial ML to generate synthetic data: the forger learns how to generate fake data that the forgery detector must distinguish from real data.

We close this section by outlining some common implementation issues.

We want the forger algorithm to generate many plausible examples that would trick the forgery detector. However, it is quite likely that the forger will start producing only one single adversarial example. This is known as mode collapse because the forger collapses to a few modes instead of generating many fake outputs (or modes). For instance, when generating synthetic data, it could end up generating only one single plausible scenario of data.

Both the forger and forgery detector algorithms must be trained. From a technical point of view, it is very challenging to implement a training process that would eventually converge to the state where both algorithms can be considered trained. In simpler terms, the learning process might never terminate.

Finally, Adversarial ML algorithms are often used in conjunction with other ML tools. This implies that risks inherent to those other tools also need to be addressed.

§ See, for example, Cao, J., Chen, J., Hull, J., Poulos, Z.: Deep Hedging of Derivatives Using Reinforcement Learning (2021) https://arxiv.org/abs/2103.16409

KEY QUESTIONS FOR RISK MANAGEMENT

Financial institutions should address several key questions if they want to employ ML models in their businesses.

- **What is the source of data on which the model will be trained?**

The data should be checked for poisonous examples, especially if the data are from a public source. It is a good practice to vet the data either internally or by using trustworthy external third parties.

- **Do the training data contain sensitive information?**

If the data used in training an ML model contain sensitive information, extra steps should be taken to protect the model against inference attacks.

- **Does the model apply any externally developed ML algorithms?**

Due to the abundance of open-source ML algorithms, developers may use externally developed algorithms in their models. Financial institutions should vet all algorithms that are not developed internally. Open-source models may be intentionally poisoned before they are made public.

- **What steps have made the ML model robust against adversarial attacks?**

Some technology companies have created open-source tools that can be used to test ML models for robustness against adversarial attacks. For example, Microsoft's Counterfeit is a tool thatcan be used to assess models against some types of attacks. IBM's Adversarial Robustness Toolbox is another tool that can be used to check whether ML models are vulnerable to certain adversarial attacks.

- **How is the model going to be maintained?**

If the model is retrained on newly available data, processes should be developed to prevent poisonous data from being used. This is particularly important when third parties are involved in model development or maintenance.

CONCLUSION

Wide-spread adoption of ML by financial institutions makes these institutions vulnerable to new types of attacks and to increases in security threats through possible data manipulation and model exploitation. Firms adopting ML technologies must anticipate threats of model theft, breach of private information, and model manipulation by adversaries. We have discussed some of the existing defense approaches that can be employed and listed key questions that financial institutions should address before employing ML algorithms in their businesses.

REFERENCES

. Edwards, H., Storkey, A.: Censoring Representations with an Adversary (2016) https://arxiv.org/abs/1511.05897

. Eykholt, L., Evtimov, I., Fernandes, E., Li, B., Rahmati, A., Xiao, C., Prakash, A., Kohno, T., Song, D.: Robust Physical-World Attacks on Deep Learning Models (2015) https://arxiv.org/abs/1707.08945

. Faghan, Y., Piazza, N., Behzadan, V., Fathi, A.: Adversarial Attacks on Deep Algorithmic Trading Policies (2020) https://arxiv.org/abs/2010.11388

. Goodfellow, I., Shlens, J., Szegedy, C.: Explaining and Harnessing Adversarial Examples (2015) https://arxiv.org/abs/1412.6572

. Mariani, G., Zhu, Y., Li, J., Scheidegger, F., Istrate, R., Bekas, C., Malossi, A.C.I.: PAGAN: Portfolio Analysis with Generative Adversarial Networks (2019) https://arxiv.org/abs/1909.10578

. Morris, J., Lifland, E., Yoo, J.Y., Grigsby, J., Jin, D., Qi, Y.: TextAttack: A Framework for Adversarial Attacks, Data Augmentation, and Adversarial Training in NLP (2020) https://arxiv.org/abs/2005.05909

. Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z.B., Swami, A.: Practical Black-Box Attacks against Machine Learning (2017) https://arxiv.org/abs/1602.02697

. Takahashi, S., Chen, Y., Tanaka-Ishii, K.: Modeling financial time-series with generative adversarial networks. Physica A: Statistical Mechanics and its Applications 527, (2019)