An overview of AI and cybersecurity: why we need AI(ML) in the cybersecurity world

Martin Rupp

SCIENTIFIC AND COMPUTER DEVELOPMENT SCD LTD

Context

AI Context

As we saw in the first part, AI and more precisely ML (let's only refer to AI as ML in what follows) aim at sorting, classifying, or regressing data and performing tasks in efficient ways that are proper to machines and that humans cannot perform because the data may be too huge and the analysis may be too complex.

There are two ways to be clever: either having a very smart way of reasoning or knowing a lot of things. In ML terms, the smart way of reasoning is the model and the knowledge is the data that has been learned. ML algorithms can combine both but most often this is their capacity to treat and learn from huge volumes of data which makes a difference.

An ML will process, compare, filter, clean, test, classify, re-compare, learn etc … Depending on the type of ML and the power of the underlying supporting hardware, some incredible levels of complexity and depth can be achieved. Often such complexity comes from the combinatorics which is involved with the data itself.

The Cyber Security Context

Cyber security involves guarding, protecting, and predicting from cyber attacks. Cyber attacks can be various. The most common are:

- **Phishing;**
- **Denial of service (DDoS);**

- **Exploits (including Zero-day attacks);**
- **Ransomware;**
- **Man-in-the-middle;**
- **Others.**

The example of Phishing

Phishing is very common. It's one of the most widely used cyber attack techniques. The technique is simple, it consists in mimicking legitimate websites or interactive documents using the fact that the internet was originally designed without any specific inherent security.

A Phishing attack often involves using email or SMS with embedded links. The end goal is to funnel the user into a crafted website where user data will be stolen and/or where the user will be maliciously fooled to perform unwanted actions that may lead to financial losses and/or other damages.

A Phishing attack is often the start, a way to penetrate defenses via the weakest link of the chain: the user.

Users are not always able to filter malicious emails. Crafted websites can look the same as the original ones and propose the same look and feel.

Users can be tired, distracted…

A successful phishing attack can lead to other successful cyber attacks such as running malware which can have various behaviors.

Cyber security companies have developed for a long time anti-viruses, anti-malware, and anti-phishing software. Yet, phishing continues to work. Nearly 75% of companies claimed to have been impacted by a phishing attack in 2021.

The reason for this is the complexity involved with the prediction of this type of attack. It's harder and harder to predict if an email (or an SMS) is in reality a phishing attack or not.

Companies such as Gmail contain sophisticated and powerful defenses against phishing. Nevertheless, attackers can reverse these defenses by mimicking legitimate security prompts from Gmail.

The complexity involved with phishing attacks means that automation must be realized. This is the task of programs that scan emails and try to detect and predict phishing. Of course, there are additional measures such as ciphered email, PKI, and other security protections but they will not work in the general context of the Internet where emails can be received from various SMTP servers which may not all use such protection.

To enforce accurate phishing detection, AI provides so far the best way to boost results obtained by non-ML ways.

Example: Bayesian Classifiers for Predicting Phishing

Bayesian classifiers are a very important category of Machine Learning algorithms that use Bayesian logic to classify data. The core idea behind Bayesian classifiers is to use Conditional probabilities to decide on the most likely nature of something characterized by a set of characteristic data (e.g. features).

Concretely this means that if we can extract *n* features for a URL, where *n* is a fixed number, then by using a learning set we can predict the nature of a URL, if it is most probably a phishing URL or a legitimate URL.

For instance, let's assume we have a program that can extract 5 features for a URL:

1) Url points to some HTML page continuing Iframe redirection (binary: yes/no)
2) Url contains the symbol '@' one or more times (integer: number of '@' symbols in the Url)
3) Url size (integer)
4) Url contains an IP (binary: yes/no)
5) Use of URL shortening services (binary: yes/no)

The data set will consist of a table with 7 columns as a minimum. Below is an example of a learning set.

Url	**feature1**	**feature2**	**feature3**	**feature4**	**feature5**	**category**
https://xyz.com/download	no	0	24	no	no	legitimate
http://1.2.3.4/getdoc?x=3@	no	1	26	yes	no	fishing
https://shorturl.gg/Jez	no	0	23	no	yes	legitimate

Of course, a typical learning set will contain dozens of thousands of entries and even much more. A Bayesian Classifier will be first trained with the learning set which means here that it will simply be loaded in the memory of the classifier.

A Bayesian classifier will be able to choose by computing the conditional probabilities. In a nutshell, it will compute the probability that an unknown Url is in fact from a phishing attack by knowing that in the past Urls with a given size using shortening services *and* containing one symbol ‘@’ were classified as Phishing, and so on…

The core principle is to compute the probability of an event knowing that another event has occurred. In other terms Bayesian probabilities explain how to compute the probability of A when you know that B with a probability of P(B)has been realised, this is contained in the following formula (Bayes theorem) :

P(A|B)=P(B|A)*P(A)/P(B)

The formula reads literally "The probability of A to realize knowing that B has realized is equal to the product of the probability that B realizes knowing that A has realized by the probability that A realizes divided by the probability that B realizes"

That simple formula is a powerful engine to compute a lot of useful things in plenty of domains.

Such a Bayesian classifier must compute as many probabilities as it has entries in the learning set to decide on the nature of the Url. This can be slow. However such ML algorithms have proven to be incredibly efficient for the detection of *spam*, which is often the first method to detect a phishing attempt.

Other ML algorithms

Other ML algorithms will work also with a similar principle. A training set with features is needed at the start and then the ML algorithm (more generally the ML model) will be able to classify a URL as a phishing URL or as a legitimate URL. Of course, in all these cases, there is no 100% guarantee that the classification will be correct. *False positive rates* (F.P.R) and *False negative rates* (F.N.R) must be computed as well as confusion matrices to estimate the accuracy of the model. Often the ML aims at 'boosting' an existing test and then improving the FPR and/or the FNR values.

Other ML algorithms that are relevant for Phishing detection are:

- **Tree Forest;**
- **XG Boosting;**
- **Neural networks;**
- **Support Vector Machines (SVM).**

An overview of uses cases of Machine Learning for Cybersecurity

Use cases for ML are various and can greatly differ in terms of difficulties of implementation and interest. A study by Gartner lists the following uses cases which can benefit from AI:

- Transaction Fraud Detection

Machine Learning is successfully applied especially because it is often agreed that fraudulent transactions have specific features that legitimate transactions do not. [1] shows how XGBoost can be successfully applied to the prediction of Fraud.

- File-Based Malware Detection

The detection of malware from its software signature (usually the 'PE' header for PE-based executables) can also benefit from Machine Learning. In [2] several machine learning algorithms are described for this purpose. Most of them are based on Bayesian classifiers or Hidden Markov Model (HMM) classifiers.

- Process Behaviour Analysis
- Abnormal System Behaviour Detection

All behavior-based analysis can benefit from machine learning. Support Vector Machines are often great at that task.

- Web Domain and Reputation Assessment

See [3] for an overview of how ML can be used in such a context and how SVM-based ML algorithms can be used.

- Asset Inventory and Dependency Mapping Optimization
- Account Takeover Identification
- Adaptive Runtime Access and Entitlement
- Identity Proofing Machine vs. Human Differentiation
- Text-Based Malicious Intent Detection

Deep Learning is often used as a booster for such detection

- Same Person Identification

This is often a complicated process that could be linked to KYC (Know Your Customer) procedures. Machine Learning can help by decreasing the risks of false identification using Bayesian Classifiers or other techniques.

- Web Content Visual Analysis
- Security Operation Task Automation
- Business Data Risk Classification
- Policy Recommendation Engine
- Event Correlation
- Hazard Intelligence

Machine Learning can be used, in many contexts (not only in cybersecurity), to predict hazardous situations.

- Security Posture and Risk Scoring

Conclusion

We have seen that AI, and more precisely ML, can be used fruitfully for many use cases in cybersecurity. The implications are huge and this is a very dynamic area with many developments happening every day or almost. In the next blogs, we will provide more insight into all this.

References

[1] Online Fraud Transaction Detection Using Machine Learning Vedant Mayekar[1], Siddharth Mattha[2], Sohan Choudhary[3], Prof Amruta Sankhe´. International Research Journal of Engineering and Technology (IRJET)

[2] PE File-Based Malware Detection Using Machine Learning. Namita & Prach, Springer Advances in Intelligent Systems and Computing book series (AISC, volume 1164)

[3] IP Reputation Analysis of Public Databases and Machine Learning Techniques Jared Lee Lewis, Geanina F. Tambaliuc, Husnu S. Narman, and Wook-Sung Yoo