Dos and Don’ts of Machine Learning in Computer Security - Part1 - Summary - SidharthAnil

Presentation

The paper deals with the major pitfalls that are prevalent in the field of machine learning in the context of cybersecurity and how to mitigate these pitfalls. Moreover, the authors have also analyzed some popular literature work in the field to observe the pitfalls that these papers have fallen into. The first discussion explains the pitfalls and how it affects machine learning in cybersecurity. The analysis of other literature is left for the next discussion.
A conscious effort must be taken to avoid falling into these common pitfalls as they will negatively affect your workflow and would give you false metrics leading to highly optimistic models that eventually fail in the real world. These pitfalls are therefore outlined and elaborated using the subsequent paragraphs, grouped together based on the stage where it occurs in the machine learning workflow.

Data Collection and Labelling

P1 - Samping Bias

Description: This bias occurs when the collected data does not effectively represent the true data distribution or the real-world scenario.
In the Context of Security: Collecting malware data is extremely challenging leading to the researchers having to synthetically form new samples from the existing ones or mix different datasets.
Recommendation:

- The dataset should be considered only as an estimate of the true data distribution, and the assumptions that had to be made in order to justify the estimation should be outlined. This would help researchers to analyze the dataset and the models built on the dataset, in the right context.
- Mixing different data sources should be avoided by all means.
- Transfer learning and synthetic data generation could help reduce the bias

P2 - Label Inaccuracy

Description: This bias occurs when the labels used as ground truth for classification tasks are inaccurate and uncertain.
In the Context of Security: The malware samples do not inherently come with a ground truth value and hence services like VirusTotal have to be used for the purpose of labeling. Such services are only able to provide heuristics for the ground truth label and hence are not always consistent. Moreover, the adversary behavior of malware can shift over time, leading to a bias called label shift

Recommendation: The impact that uncertain labels have on the model could be reduced by using robust models, and cleansing and removing uncertain labels.

System Design and Learning

P3 - Data Snooping

Description: This bias occurs when the model has access to information while training, that it wouldn't have access to when it is functioning in the real world. This could be in the form of:
1. Test snooping - Using test data in the training stage
2. Temporal snooping - Data from the future is used to learn the present scenario while training
3. Selective snooping - When certain samples are removed from the dataset based on criteria that you wouldn't be able to evaluate during real-time functioning. For example, removing a sample because most of the state-of-the-art AVs couldn't classify it well.

In the context of security: Temporal snooping has been largely observed in ML-based cybersecurity models. This was also noted in a previous discussion on "Machine Learning (In) Security, A Stream of Problems". Test and Selective snooping have also been observed in many instances.
Recommendations: Split the test and train datasets as early as possible and ensure they do not come in contact with each other throughout the process. Temporal information should be considered to accommodate for temporal dependencies in the malware dataset.

P4 - Spurious Correlations

Description: This bias occurs when the model learns features that are not directly indicative of maliciousness in a file. For example, the VulDeePecker model to detect vulnerabilities use certain artifacts from the codebase, that are not really indicative of vulnerabilities.
In the context of security: A lot of the ML models in cybersecurity act as black box models making it hard to identify the presence of spurious correlation.
Recommendations: There are explanation techniques that could be used to gain a better understanding of the model, which could lead to the identification of spurious correlations if there are any. Also, as a rule of thumb, try and focus on features that will generalize well instead of the ones that will increase the training accuracy. Also, context applies heavily here because a feature that is considered spurious for one scenario would actually be relevant for another.

P5 - Biased Parameter Selection

Description: This bias occurs when the hyperparameters of the model was tuned based on test data. It is related to the Data snooping bias that was explained.
In the context of security: Security systems that are modified at training time by considering the test data would not be able to perform well when deployed in the real world.
Recommendations: Since this is highly related to Data Snooping pitfalls, the same mitigation steps would apply here as well. Furthermore, using a validation dataset would rule out the effects of biased parameter selection to a large extent.

Performance Evaluation

P6 - Inappropriate Baseline

Description: This bias occurs when the model is evaluated without baseline methods. Just comparing the model with the most similar previously proposed model, or just with over complex models would not give the full picture.
In the context of security: In security, traditional models have some advantages over complex ones. The complex models are more likely to overfit, would be more computationally expensive, and would have greater attack surfaces. This makes the comparison against traditional models also significant.
Recommendation: Compare and evaluate the model against state-of-the-art models as well as traditional models. Using frameworks such as AutoML would help in finding good baseline models.

P7 - Inappropriate Performance Measures

Description: This bias occurs when metrics are used that don't really apply well to the situation at the hand or the goal that the ML model is trying to achieve
In the context of security: Traditional metrics like accuracy might give a false picture about the performance of the model when the priority of the model in the context of cybersecurity would require a different metric.
Recommendation: Use metrics that are aligned with the goal you have for your model.

P8 - Base Rate Fallacy

Description: This bias occurs when the class imbalance encountered is ignored when interpreting the metrics.
In the context of security: Let's say you get a False positive rate of 1% and a true positive rate of 90% (positive - malware, negative - goodware). Even though this might seem like a low enough value and hence good, if the number of goodware encountered by a system in a day is 1000 and the number of malware files encountered is 10, this would imply that for every 9 malware files detected, 10 goodware files are wrongly classified as malware. This is a common pitfall in fields such as intrusion detection, website fingerprinting, etc.
Recommendation: Metrics like precision and recall are good for imbalanced cases such as these. Other metrics like Matthew's Correlation Coefficient can also be used in case the minority class is inflated.

Deployment and Operation

P9 - Lab Only Evaluation

Description: This bias happens when the model is only evaluated in a laboratory and hence the evaluation results acquired would not apply well in the real world.
In the context of security: Since it is practically hard to test a malware detection system out in the real world, a lot of the models fall into this pitfall of being tested out only in a closed-world setting.
Recommendations: Emulate the diversity that is normally observed in the real world, and the practical constraints such as the storage constraints a device would have in the real world. Also, take into account the temporal and spatial relations.

P10 - Inappropriate Threat Model

Description: While considering the threat model based on which the system should be secure, many fail to account that the utilization of machine learning increases the attack surfaces and hence requires a change in the threat model.
In the context of security: Along with the standard threats that the malware detection systems aim to defend against, it must also consider the new machine-learning related threats and evasion strategies that the model would encounter
Recommendation: The threat model must be created by taking into account the adversarial attacks that utilize the internal design or working of the machine learning model. In this case, it is best to try and make your model secure against white-box attacks, that is, assuming that the attacker has information about the model and hence can utilize this information.

Discussion

- This discussion started off on the topic of label inaccuracy. Because of the research that is happening in the domain of malware, malware labels organically change. Hence, the results that I might have on my model today would not be a relevant indicator in the future as the labels could have changed and evolved, making my evaluation obsolete.
- Another point discussed was how frequently researchers fall into the pitfall of mixing up datasets because of the lack of data, hence introducing additional bias.
- A tangent topic discussed was website fingerprinting, and how it is actually done behind the scenes. One class classifiers use a collection of harmless benign identifiers to fingerprint a user across websites.
- Another major topic of discussion was how a machine learning model will perform well if it encounters the data that it was trained on. This implies that an AV detection model that was trained on a dataset that is local to Brazil would function well in Brazil but might not have the same performance in the United States. This would raise the opinion that the best AntiVirus system would be the one that was trained with the local data. This also led to the point that most of the attacks that happen have a local origin. Even though there are highly targeted attacks by exceptionally skilled groups, the majority of the attacks are by people in the same general region,
- The AV companies do utilize global datasets and at the same time collect files locally creating local datasets. The proportion of global to local data that it should focus on is a game of balance and hence a topic of discussion
- Another topic discussed was the ways in which malware is transmitted, sometimes even in USB sticks. This led to recalling StuxNET, a virus targeting Iran's nuclear program which is widely believed to have been deployed through a USB stick.
- The concept of a threat model was also extensively discussed. A threat model is an outline of all the potential threats that are under consideration for a system. A system can never be 100% secure and hence it is normally said to be secure considering a certain threat model. The analogy used was that of a house. The house is said to be secure if all the locks to the doors and windows are functioning and an intruder wouldn't be able to just walk right in. However, it is still true that a SWAT

team would be able to break in through the roof. This is not normally considered because it is not part of the implicit threat model that people have created in their minds.