Using Machine Learning for Network Intrusion Detection: The Need for Representative Data

Gavin Wolf
Dr. Taghi M. Khoshgoftaar
Florida Atlantic University
Boca Raton, Florida USA
May 2, 2016

Keywords — **intrusion detection, network attack, representative data, machine learning, network security, network flow analysis, anomaly detection**

Summary & Conclusions — With the world's growing reliance on internet-based technologies, the need for systems to secure information and provide reliable environments is more important than ever. Network Intrusion Detection Systems (NIDS) that monitor network traffic for suspicious activity have become an important element of network security. An NIDS is faced with the task of determining whether traffic is malicious or non-malicious, a classification problem that machine learning seems well-suited to tackle. However, despite the great success of machine learning in a wide range of domains, NIDS's using machine learning have not taken off in large-scale operational environments. And this is despite numerous research studies demonstrating the effectiveness of using machine learning for network intrusion detection. Researchers are increasingly focusing on this conundrum.

Among the reasons given for the lack of industry take-up of NIDS's using machine learning, the lack of representative training data seems to be the most salient. One study in particular [1] demonstrates the enormous impact that non-representative training data can have on model performance. It is also becoming apparent that the datasets most commonly used in network intrusion detection research do not meet the criterion of representativeness. Several recent attempts have been made to develop both more representative datasets and better methodologies for generating datasets. If and when improved datasets supplant the existing benchmark datasets, the quality of network intrusion detection models should improve greatly, increasing the likelihood of

industry adoption. Even at the point of widespread usage of high-quality benchmark datasets, there will still be a number of challenges for deploying NIDS's using machine learning, including: computational efficiency, keeping up with modern (and increasingly mobile-targeted) attacks and keeping up with the permanently increasing amount of network throughput. The contribution of this paper is to draw attention to the importance of high-quality, representative datasets for network intrusion detection using machine learning, to present some recent attempts at creating such datasets, and to suggest additional issues that network intrusion detection using machine learning must overcome.

1. Introduction

While numerous papers have demonstrated the apparent effectiveness of using machine learning techniques for network intrusion detection, such techniques have not seen widespread usage in the industry. Researchers have identified a number of reasons for this lack of adoption. One of the most salient reasons is the the lack of representative training data. There are a number of challenges to creating good datasets including privacy issues, labor-intensiveness and general difficulty. A high-quality dataset should be correctly and completely labeled, which typically requires arduous manual labeling by domain experts. A high-quality dataset must also be representative of real-world network traffic. The three datasets most commonly used by researchers to build and evaluate network intrusion models do not meet these requirements. Improving the quality of training datasets will be of great benefit in developing NIDS's using machine learning that can help network administrators to identify and combat network intrusions, which is the ultimate goal of research into network intrusion.

2. The Role of NIDS's in Network Security

As internet-based technologies have grown in usage, importance and value, the attacks on them have become more numerous and sophisticated. Perhaps the most common type of attack is SSH brute-force attacks, in which an attacker uses software to attempt authentication for a remote machine. SSH, or Secure Socket Shell, is an encrypted network protocol that allows a client to connect with a server securely over an unsecured network. Once an attacker has gained access to a network through an SSH brute force attack, he or she can attempt additional attacks such as

botnets and distributed denial of service (DDoS) attacks. According to some network-security experts, “every individual SSH port on the Internet will experience brute force attacks” [2]. In a 2014 study, 51% of respondent companies said they had been comprised by SSH brute force attacks in the last 14 months [3]. Although SSH brute force attacks are a seemingly simple form of attack, their pervasiveness and effectiveness highlight the need for additional security measures.

The first line of defense in network security consists of user authentication, encryption and firewalls. These types of security are focused on keeping attackers out of a local network. NIDS’s can be thought of as a second line of defense that aims to identify an attacker that has gotten past the other security mechanisms. Intrusion detection can be either host-based – located on computer end points – or network-based – located on the network. Host-based intrusion detection is not scalable and if a host gets infected it puts the entire network in danger. For these reasons, machine learning researchers have been focused mainly on developing network-based intrusion detection methods, as opposed to host-based methods. For network-based intrusion detection, the main approaches are misuse detection and anomaly detection.

NIDS’s using misuse detection have been widely used for more than 15 years [4]. Misuse detection consists of comparing network traffic to a library of known attacks. This approach is very effective at detecting known attacks, but not very effective at detecting new attacks. And once a new attack is discovered, it takes time for experts to analyze it and incorporate it into the library of known attacks. During this time, the network will still be vulnerable to the particular type of attack. Two of the more popular software systems using this technique are Snort and Bro.

In contrast to the popularity of systems using misuse detection, NIDS’s using anomaly detection have not seen widespread adoption. Anomaly detection involves analyzing network traffic to detect abnormal patterns. The promise of this approach is that it can detect new threats. A significant challenge for this type of system is false positives, which arise when abnormal, but non-malicious, patterns are classified as attacks. How to differentiate between acceptable abnormal traffic and attack traffic is a difficult task, one that machine learning seems well-suited to tackle. The promise, but difficultly of this task has led some researchers to call it “the holy grail” of anomaly detection [5]. It it puzzling that machine learning been used with great success in a wide range of domains, but not in intrusion detection. Some researchers argue that the domain of network intrusion detection presents unique challenges for machine learning.

Researchers have identified the following reasons why the domain of intrusion detection is uniquely challenging: (i) a high cost of errors, (ii) lack of training data, (iii) a gap between model results and operational interpretation, and (iv) variability in input data [5]. In certain domains the inevitability of errors may not be problematic. For example, the creator of Amazon's recommendation engine, Greg Linden, once commented, "Recommendations involve a lot of guesswork. Our error rate will always be high" [6]. While high error rates may be acceptable in domains with a low cost of errors – like recommendation engines – they are very dangerous in the domain of intrusion detection. Missed attacks can lead to data breaches, business interruptions or worse. The domain of intrusion detection also presents the challenge of having to work against motivated attackers, who deliberately try to outsmart and evade security mechanisms. While all of these factors add to the uniqueness of the intrusion detection domain, the lack of training data is perhaps the most intractable issue, one that researchers have recently focused on demonstrating and overcoming.

3. Demonstration of the Problem of Inadequate Datasets

In order to demonstrate the problem of inadequate datasets in the field of network intrusion, I will present the findings from Najafabadi, Khoshgoftaar and Kemp in their 2015 paper entitled, "The Importance of Representative Network Data on Classification Models for the Detection of Specific Network attacks" [1]. In the study presented in that paper, the authors constructed datasets, and models based on those datasets to classify traffic as either normal or SSH brute force attacks. The datasets were made up of real network traffic labeled by network experts. One dataset was intended to be not adequately representative – "Phase 1 data" – and one dataset was intended to be representative – "Phase 2 data". The researchers then built classification models using the non-representative data and evaluated them on the representative data. The results were abysmal, demonstrating the importance of using representative data.

The Phase 1 data includes SSH traffic as well as all other network traffic. Phase 1 data was collected over a period of 24 hours from a live network. The Phase 2 data includes only SSH traffic and was collected over a period of one week. Failed login attempts were also added to the Phase 2 dataset because such attempts produce the type of traffic that would be expected to generate false alarms. Adding the failed login attempts also makes the data more representative and ensures that a model that deems all failed login attempts to be attacks would not perform well.

The data collected was packet data, which was then used to create network flows. Network flows describe "network sessions in terms of the aggregation of unidirectional sequence from network packets that share some network attributes" [1]. Using this "metadata" about packets has some advantages over using the packet data itself. One advantage is that analyses of network flow data can be done with both unencrypted and encrypted data. Another advantage is that network flows are easier to analyze because they have fewer features than packet-level data. The features used for the flow analysis included: source port, destination port, number of packets, number of bytes, duration and TCP flags, including flow flags, initial flags and session flags.

The researchers used four different types of classification models for their analysis: 5-Nearest Neighbor (5-NN), two forms of C4.5 Decision Trees (C4.5D and C4.5N), and Naïve Bayes (NB). Fivefold cross-validation was used to show that the models perform well within a given dataset, indicating that they are suitable for testing on another dataset. To evaluate the results, True Positive Rate (TPR), False Positive Rate (FPR) and Area Under the Receiver Operating Characteristic Curve (AUC) were used. TPR is the hit rate, i.e., the percentage of brute force instances that are correctly predicted as brute force by the model. FPR is the false alarm rate, i.e., the percentage of the non-brute force data that is wrongly predicted as brute force by the model. AUC measures the performance of a model across all decision thresholds. Higher AUC values indicate a high TPR and a low FPR, which is preferable in the context of network intrusion detection.

As indicated in *Table 1* and *Table 2* below, the cross-validated results on each the Phase 1 data and the Phase 2 data were good, with the best models resulting in 99%+ AUC, 99%+ TPR and FPR under 1%. These results merely indicate that the models perform well when the test data is a subset of the overall dataset. The true test is applying the models built using the Phase 1 dataset to the more-representative Phase 2 dataset. The results of this approach are shown in *Table 3* below and they are awful, even worse than a purely random classifier. And that is despite the data in both datasets coming from the same network.

classifier	AUC	TPR	FPR
5NN	0.9988	0.9898	0.0082
C4.5D	0.9979	0.9843	0.0078
C4.5N	0.9893	0.9982	0.0262
NB	0.9975	0.9987	0.1050

Table 1 – Cross-validated results on Phase 1 Data

classifier	AUC	TPR	FPR
5NN	0.9981	0.9920	0.0103
C4.5D	0.9981	0.9970	0.0085
C4.5N	0.9990	0.9973	0.0080
NB	0.9904	0.8442	0.0021

Table 2 – Cross-validated results on Phase 2 Data

classifier	AUC	TPR	FPR
5NN	0.014537	0.0413	0.9886
C4.5D	0.283194	0.0447	0.4869
C4.5N	0.341873	0.0540	0.4878
NB	0.048421	0.1541	0.9949

Table 3 – Fit/Test results (Phase 1 data is used as the fit data and Phase 2 data is used as the test data)

These results demonstrate very clearly the need for representative datasets. Although producing high-quality datasets can be a very difficult and time consuming task, doing so is of utmost importance for building network intrusion detection models. These results suggest one possible reason why the success of machine learning models in research studies has not translated into the successful use of NIDS's using machine learning in practice: the data used in the studies was inadequate. Researchers must not only demonstrate that their models are accurate, they must also demonstrate that the datasets they used were representative of the real-world traffic that those models are intended to be used on. Researchers should also make every effort possible to make their datasets public so that others can properly assess their project as a whole. Public datasets also allow researchers to evaluate and compare different approaches, and to attempt to reproduce published results.

4. Shortcomings of the Most Popular Datasets

Developing a good dataset with properly labeled network traffic requires a lot of time and effort by network security experts who need to analyze the traffic. Furthermore, privacy concerns make many institutions disinclined to release information about the traffic on their networks. Most of the work done on using machine learning for anomaly detection has used the following three datasets: KDD 99 / DARPA, KYOTO and ISCX. To the extent that these datasets are not representative of real-world traffic data, the models built with them should not be expected to

perform well on real-world traffic. These datasets are increasingly being recognized as inadequate, which when combined with the results of the study presented in the last section, suggests a reason why we have not seen many real world deployments of NIDS's using machine learning.

The KDD 99 dataset was created by using the tcpdump part of the 1998 DARPA intrusion detection dataset created by MIT Lincoln Laboratory [7]. The KDD 99 dataset has many issues that make it inadequate for current research. One obvious issue is that it is nearly twenty years old and the composition and complexity of networks has changed immensely since it was created. Another issue is that it is not made up of real-world traffic data in the first place. The data was collected from a simulated network and the attack data was manufactured, rather than discovered. A couple more recent datasets have also been used in numerous studies: the KYOTO and ISCX datasets.

The KYOTO dataset was collected over two and a half years, from 2006 to 2009. The KYOTO dataset includes real-world traffic, but only certain types: email and DNS. The attack data was obtained from honeypots. Honeypots are decoy servers set up with the intention of attracting attackers. A benefit of using honeypots is that all activity on them can be considered to be suspicious, thereby eliminating the need for manual labeling [1]. However, because all activity on honeypots is considered to be suspicious, there are no false positives. False positives are a common occurrence in real world systems, so a dataset that does not allow for them is not ideal. For these reasons, the KYOTO dataset is not representative of real-world network traffic and models built with it should not be expected to perform well on real-world traffic.

The ISCX dataset was created in 2012. The data is based on on detailed, dynamically generated network traffic and intrusions. The normal traffic is provided by "executing user-profiles that were synthetically generated at random synchronized times creating profile-based user behavior" [8]. Several attack scenarios are carried out to generate the attack data. While the ISCX data was carefully crafted, real world network traffic is inherently unpredictable. The simulated nature of the ISCX data and its lack of background noise make the ISCX dataset not as realistic as real-world network traffic [1]. As the shortcomings of these datasets have been becoming more apparent, researchers have been working on new and improved datasets.

5. Some Promising New Datasets

An ideal dataset would be based on and representative of current real-world network traffic, and would be correctly and completely labeled. One proposed dataset that seeks to overcome the issues facing the three popular datasets is the Session Aggregation for Network Traffic Analysis (SANTA) dataset [9]. The SANTA dataset was collected from a commercial Internet Service Provider, so it is comprised of real-world network traffic. The dataset does not rely on honeypots or simulated attacks, but rather it was "painstakingly inspected manually to identify actual attacks that occurred on the subject network." Even though the attacks were identified by network experts, it is still possible that they misidentified attacks or missed attacks altogether. The features included in the dataset include both flow-level and packet-level data. The comparison in *Table 4* of the SANTA dataset and the three most popular datasets was provided by the creators of the SANTA dataset. The SANTA dataset improves upon the other datasets by including realistic normal traffic, manually-labeled attacks and modern attacks, in addition to including a number of other novel features.

Advantages	SANTA	KDD	Kyoto	ISCX
Realistic normal traffic (not simulated)	YES	NO	NO	NO
Penetration testing attack traffic	YES	YES	NO	YES
Real, in the wild attack traffic	YES	NO	YES	NO
Modern attacks	YES	NO	NO	NO
Manually inspected and verified attack labels	YES	NO	NO	NO
Periodicity attributes	YES	NO	NO	NO
Repetition attributes	YES	NO	NO	NO
Convergence attributes	YES	NO	NO	NO
Velocity attributes	YES	YES	YES	NO†
Self-aware scanning attributes	YES	NO	NO	NO

† Implied (Can be calculated)

Table 4 – Advantages of the SANTA dataset

Another new dataset based on real-world traffic is the Indian River State College (IRSC) dataset [8]. The dataset includes flow and packet data from a real-world network. The attacks were both controlled (generated by the authors) and uncontrolled (real attacks on the network). The inclusion of controlled attacks is intended to provide more accurate labeling. The authors indicate that the dataset could be further improved by including honeypots and additional attacks that are targeted at mobile users and routers.

While the SANTA and IRSC datasets seem to be meaningful improvements over the three most popular datasets, they have not yet been made publicly available. Once high-quality real-world datasets become public and widely adopted, they will be valuable as benchmark datasets for researchers that are evaluating the use of machine learning in network intrusion detection.

6. Remaining Issues

The introduction and acceptance of benchmark datasets based on real-world traffic and sound methodologies – while major and important steps – would by no means ensure that NIDS's using machine learning will be adopted en masse. These datasets will also need to be adaptive enough to incorporate modern attacks and differing types of attacks, like those focused on mobile networks. NIDS's using machine learning must also be able to deal with "the non-stationary behavior of network traffic along with the permanent growth of the network throughput" [4]. Furthermore, the systems must demonstrate that they can operate with reasonable and predictable computational efficiency.

In the evolving battle between network administrators and increasingly sophisticated attackers, there can be no guarantee of victory for network administrators. Widespread industry adoption of NIDS's that successfully incorporate machine learning would certainly be a big win for network administrators. While success with machine learning for anomaly detection has been shown with non-representative datasets, the true promise of NIDS's using machine learning will not be realized until they are demonstrated on representative datasets and deployed in real-world, operational environments. On the road to widespread adoption, developing and releasing high-quality, representative datasets is paramount. Although intrusion detection is a particularly difficult domain, recent research gives many reasons to be hopeful that machine learning can be applied to it successfully.

References

[1] M.M. Najafabadi, T.M. Khoshgoftaar, and C. Kemp. "The Importance of Representative Network Data on Classification Models for the Detection of Specific Network Attacks." *Proceedings of the 21st ISSAT International Conference on Reliability and Quality in* Design, 2015.

[2] M.M. Najafabadi, T.M. Khoshgoftaar, C. Kemp, N. Seliya, and R. Zuech. "Machine Learning for Detecting Brute Force Attacks at the Network Level." *2014 IEEE 14th International Conference on Bioinformatics and Bioengineering*, 379-385, 2014.

[3] M.M. Najafabadi, T.M. Khoshgoftaar, C. Calvert, and C. Kemp. "Detection of SSH Brute Force Attacks Using Aggregated Netflow Data." *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*, 283-288, 2015.

[4] C.A. Catania and C. García Garino. "Automatic network intrusion detection: Current techniques and open issues." *Computers & Electrical Engineering*, 38(5): 1062-1072, 2012.

[5] R. Sommer and V. Paxson. "Outside the closed world: On using machine learning for network intrusion detection." *IEEE Symposium on Security and Privacy*. IEEE Computer Society, 2010.

[6] G. Linden. "Make Data Useful." Data Mining Seminar, Stanford University, 2006. http://glinden.blogspot.com/2006/12/slides-from-my-talk-at-stanford.html.

[7] "Darpa intrusion detection data sets - mit lincoln laboratory." http://www.ll.mit.edu/ideval/data/.

[8] R. Zuech, T.M. Khoshgoftaar, N. Seliya, M.M. Najafabadi, and C. Kemp. "A New Intrusion Detection Benchmarking System." *Proceedings of the Twenty-Eighth International Florida Artificial Intelligence Research Society Conference*, 2015.

[9] C. Wheelus, T.M. Khoshgoftaar, R. Zuech, and M.M. Najafabadi. "A Session Based Approach for Aggregating Network Traffic Data - The SANTA dataset." *2014 IEEE 14th International Conference on Bioinformatics and Bioengineering*, 369-378, 2014.