Evaluating Machine Learning Based Intrusion Detection Techniques In A Real World Setting

Jonas Patrick Debes

October 26, 2017

Abstract

Intrusion detection systems tend to be assessed primarily on their ability to produce a low false positive classification rate, whilst keeping the detection rate at a reasonable level. However, these two criteria alone cannot sufficiently evaluate the performance of such a system in a real world operational context, where a technique is only feasible if it can produce an expected effectiveness, that outweighs its cost in a specific deployment setting. This paper aims to build upon existing guidelines that allow for such an assessment. In addition we introduce a metric that quantitatively defines the operational risk of a machine learning based classifier in a specific deployed settings. The application of these guidelines are then demonstrated by reviewing an existing publication that introduces two frameworks based on two distinct types of artificial neural networks. Based on the lack of scope and specific environmental context contained within the reviewed paper, we conclude that the results cannot be reliably interpreted to develop a expected effectiveness in any specific operational environment.

1 Introduction

Intrusion detection systems define a passive strategy for monitoring networks or systems to identify malicious activity based on a constructed profile of malicious or benign traffic. The increasing popularity of terms such as big data, illustrate that the amount of traffic data generated will continue to grow at a level that is unmanageable by security analysts via a manual approach. This enshrines machine learning based classification techniques as a pragmatic solution.

Our review has found that machine learning classification techniques are frequently published, often boasting a high detection rate and a low false positive rate. It is misleading to think that this detection rate is the primary indicator of an IDS's performance. In fact, due to the incredibly low percentage of true intrusion attempts amongst a swarm of traffic data, the limiting factor is more often the rate of false positives, due to the

base rate fallacy [1, 2]. Therefore, it is no surprise that a significant amount of existing literature, is focussed on reducing false positive classification rates.

This paper is primarily focussed on developing a set of guidelines and metrics that will allow the reader to evaluate any machine learning based intrusion detection technique's applicability in a specific real world operational setting. It is our hope that these guidelines will also outline the foundation of a successful machine learning based classifier in terms of providing real world value, filling a necessary void in a community that is primarily concerned with publishing techniques that minimise false positives. It is our belief that this real world value is determined not only just by a classification technique but also the operational environment in which it may find itself deployed to.

Keeping this in mind, we endeavour to provide two main contributions. The first contribution is to elaborate and further develop the machine learning application guidelines introduced by Sommer & Paxson [3] as a set of evaluation criteria to meet our goal. Furthermore, this will include a novel application of quantitative risk management to introduce a metric to theoretically define and practically compare the individual or cumulative risk of both false positive and false negative classifications in different operational environments. The second contribution involves the application of our criteria to review Zhang et al. [4]. These authors introduce two neural network frameworks based on the two distinct classes ANN classes.

The rest of this paper will be structured as follows; section 2 provides an introduction to the base rate fallacy in the context of machine learning. In section 3 we will outline the aforementioned assessment guidelines. Section 4, will first introduce the concepts surrounding artificial neural networks, before giving a brief overview of the techniques introduced by Zhang et al. Section 5 provides a discussion surrounding the application of each of the guidelines applied to the review of Zhang et al. Finally, concluding comments of the two contributions will be made in section 6.

2 Base Rate Fallacy

The base rate fallacy, stemming from the Bayes theorem illustrates that a given probability can differ greatly when taking the probability of related outcomes into account. The simple equation (1) can be used to calculate a conditional probability using the probability of two random variables.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \tag{1}$$

From equation (1), we can see that in order to calculate the probability of A given that event B has occurred, we must know the probability of A & B occurring independently, as well as the probability of B occurring given A. This formula can be derived from the calculation of the probability of both A and B occurring, illustrated in equation (2). The Bayes equation for either $P(A|B)$ or $P(B|A)$ can be constructed from this relationship as

shown in equation (3).

$$\begin{aligned} P(B \cap A) &= P(A \cap B) \\ P(A \cap B) &= P(A|B) \cdot P(B) \\ P(B \cap A) &= P(B|A) \cdot P(A) \end{aligned} \tag{2}$$

$$P(A|B) \cdot P(B) = P(B|A) \cdot P(A) \tag{3}$$

Axelsson [1] illustrates that regardless of how high the true positive classification rate in an intrusion detection system $P(A|I)$ is, the primary bottleneck of such a system is the large amount of false positives classification $P(A|\neg I)$ of benign network traffic. Due to the incredibly low probability of intrusion attempts $P(I)$ amongst a swarm of benign traffic data, the probability of any alarmed traffic log being truly malicious $P(I|A)$ is low.

$$\begin{gathered} P(I|A) = \frac{P(A|I) \cdot P(I)}{P(A|I) \cdot P(I) + P(A|\neg I) \cdot P(\neg I)} \\ P(\neg I) = 1 - P(I) \end{gathered} \tag{4}$$

Equation (4) shows that the probability of a particular piece of network traffic being intrusive given that it has set off an alarm, is mostly influenced by the very high probability of a piece of traffic setting off an alarm and being non-intrusive $P(A \cap \neg I)$. From the perspective of the designer of an IDS, $P(\neg I)$ cannot be influenced by the machine learning technique but $P(A|\neg I)$ can. Thus a focus on reducing $P(A|\neg I)$ has a greater impact on maximising $P(I|A)$ due to $P(\neg I)$ being very large, when $P(I)$ is very small.

However, achieving a reasonable P(I|A) by minimising $P(A|\neg I)$ will often require this value to be unrealistically low, depending on exactly how large $P(\neg I)$ is. Axelsson argues that in the intrusion detection domain $P(\neg I)$ can be as high as 0.99998, requiring an unachievable false positive rate. Such a high value is indicative of the amount of benign network traffic relative to real intrusion attempts.

3 IDS Performance Metrics

Sommer & Paxson define criteria to evaluate the effectiveness of machine learning algorithms in the domain of anomaly intrusion detection. The paper elaborates these criteria as a set of guidelines for the application machine learning to the intrusion detection domain.

3.1 Threat Model

In order to measure the performance of any IDS, it is important to establish the application environment and goals. The threat model, contains information in the form of a prioritised set of threats, discovered from a potential attacker's point of view. The threat modelling

process can be seen as the step preceding the development of security requirements [5], from which a prospective intrusion detection technique could be based.

Points to cover during threat modelling can range from the environment of the intended application to estimating the attacker's behaviour. The latter can be very important as many supervised machine learning algorithms classify data based on some form of statistical model constructed from the training data. Therefore a potential IDS may be compromised if the attacker is aware of the nature of the IDS and varies their attack from previous attacks. Furthermore, if the IDS is continually refining its model based on additional network traffic encountered, the attacker may inadvertently or advertently bias the classification model through further interaction with the system. Considering the environment i.e the operational context in which the system will be deployed, will give an indicator as to which general approach may detect future attacks best. For example an enterprise in the banking sector will likely have a different network environment with different security requirements than a social media website.

Although organisations in different environments may have the same category of general attack, the prioritisation of these threats, is the primary differentiator between environments. Prioritisation allows economic factors to weigh into the later security requirements, to ensure that an organisation is implementing a portfolio of security mechanisms that maximises security benefit at the lowest possible cost. Lampson [6] defines this as real world security, where a firm may increase its security to the point that the costs incurred by a potential attacker outweigh any benefit obtained from the attack. Lampson indicates that maximising security, is rarely done in practice, simply because it is too expensive. Therefore, in a more quantitative sense, a firm should optimise its security strategy to a point where the difference between the cost and benefit is maximised. Such analysis requires knowledge specific to a particular environment such as, the behavioural profile of potential attacker and the cost of both false alarms (false positive) and undetected intrusions (false negative).

Threat modelling operates at a higher strategic level and is applicable to a the broader category of information security. Threat modelling should occur before evaluating any kind of specific approach such as intrusion detection. There is just as much benefit determining that a particular approach is not applicable to a specific environment, in terms of both return on investment and efficacy (elaborated in section 3.4). In other words, an intrusion detection technique may not be effective and/or may provide a benefit that is lower than its cost in a particular environment.

Establishing the threat model, helps place a prospective technique into context of a specific operational environment. A large proportion of existing literature measure the effectiveness of machine learning based intrusion detection techniques by analysing their ability to classify with a degree of accuracy on a general public network traffic training dataset [3]. Such an approach fails to take into account factors such as the operational environment, profile of an attacker and determining whether the prospective technique

provides any real world security i.e maximising benefit while minimising cost. Adding this additional performance criteria should provide this necessary context from which to evaluating the operational value of machine learning based intrusion detection in the context of a particular environment.

3.2 Cost of Errors

As outlined in section 2 discussing the base rate fallacy, the high rate of benign network traffic relative to the amount of malicious network traffic multiplied by the rate of false positives, shows that any IDS system based on machine learning will likely produce a large number of false positives, which presents itself as a limiting characteristic in machine learning [3, 1]. Therefore the decision to apply machine learning to any domain, rests on how expensive a false positive is to the stakeholder intending to apply such a technique. This concept introduced by the Sommer & Paxson can be further expanded upon by defining it as a quantitative measure of risk.

$$R_i = P(F) \cdot F_{cost} \tag{5}$$

Equation (5) shows the classical risk formula i.e a weighted probability applied to such a scenario, where the risk of a false positive can be expressed as the product of the probability $P(F)$ of a false positive (or false negative) occurring and the cost F_{cost} of this false positive. This definition, clarifies Sommer & Paxson's statement by inferring that the decision to apply a machine learning based technique to intrusion detection rests on the risk R_i of false positives.

The response portion of Wu & Zhang [2] provide an excellent example for such a risk analysis. In this paper the authors clearly outline that the high rate of false positives of a model based on supervised machine learning, that aims to infer criminality based on still faced images of humans makes it a poor choice to use in a practical real world setting. The authors and critics imply that the cost F_{cost} of false positives produced by a system that legally diagnoses criminality based on facial characteristics would be astronomically high. Therefore, no matter how far the false positive $P(F)$ rate is numerically reduced, the risk R_i remains high.

Although the calculated R_i value on its own does not provide much information of how relevant a specific technique is in its setting, in the same way that financial metrics such as the price to equity ratio when calculated on a single listed company equity do not necessarily indicate whether it is a well priced investment, R_i provides a good tool to compare different approaches in specific environments to one another, eventually allowing a rule of thumb value to be determined.

The false positive and false negative rates appear to have an inverse relationship [3] in sense that optimising any specific technique to reduce false positives will likely increase false negatives and vice versa. This highlights a potential strategy, namely calculating

the R_i of both false positives and false negatives, giving an insight as to how tightly the classification model should be fit to the training data, to minimise the sum R_T of both calculated R_i's, as shown in equation (6). A loose fit causes a decreased detection rate would reduce the rate of false positives, but increase the rate of false negatives. On the other hand a tighter fit would have the opposite effect. Fitting a model too tightly to the training data is also known as "overfitting".

$$R_T = P(FP) \cdot FP_{cost} + P(FN) \cdot FN_{cost} \tag{6}$$

In the case that the above optimisation does not reduce the total risk to an acceptable level, an additional post-processing step [3] can be added upstream of the the ML classifier to filter the false positives that are forwarded to the end user based on some additional criteria.

3.3 Scope

We have already discussed to some extent the issue regarding anomaly based intrusion detection's low tolerance towards false positives. Section 3.2, illustrates that undesired classification rates such as false positives and false negatives can be weighted with a cost to conform to the definition of risk. This risk assessment can be applied to determine how costly false positives are in their specific operational environment.

It is also worthwhile to consider whether the false positive rates themselves can be reduced, when optimising risk. Sommer & Paxson introduce the idea of adjusting the scope of intrusion detection to minimise these rates. Unsupervised machine learning techniques rely on building a statistical model based on historical data, where the data points are highly similar/dissimilar to that being detected. By developing a specific and narrow scope for any intrusion detection model, one has a better chance of finding a specific training data set and machine learning technique that gives the best classification rate, with the lowest rate of false positives. In contrast, a very wide scope, will require a highly variable and dissimilar dataset. Such a dataset will likely result in a higher misclassification rate because it is difficult to establish a notion of normality or abnormality, from which to classify incoming data [3].

A an optimal model based on a less variable dataset can be achieved by narrowing down the detection to a specific type of intrusion attempt. For example, port scanning traffic which aims to determine to scan various IP addresses in a specific subnet and fingerprint network services which are open for each IP address. When the targeted machines are placed behind a firewall the port scanning tool attempts to send obscure message, which are outside of the usual flow of the TCP protocol in order to elicit a response. It is not hard to see why an anomaly detection strategy is appropriate in such a scenario.

A common pitfall when scoping discussed by Sommer & Paxson, is to take an inverse problem solving approach where a technique is deployed by deciding on a machine learning

approach first and then searching for a problem to which it could apply. Such an approach, can lead to a selection bias when attempting to assemble a training dataset. It may also lead an organisation away from finding an approach that maximises value based on a cost benefit analysis and towards a problem search in order to justify the use of the chosen technique.

3.4 Performance Evaluation

Once a technique has been decided upon, it becomes crucial that a relevant dataset is selected and that the technique's performance is correctly evaluated to gauge its performance in an operational real world setting. Therefore, a crucial part of introducing a new ML based IDS is the development of a clear set of evaluation criteria.

Similarly, a technique's behaviour in its intended environment is also dependent on the set of training and testing data used to calibrate its classification model. Sommer & Paxson stress the importance of finding a relevant dataset for this purpose. The results of many existing applications depend entirely on publicly available dataset, of which the most common supersets are DARPA [7] and KDD [8]. These datasets however, are outdated and often do not provide an accurate picture of what a real world traffic dataset may look like [3]. Existing alternatives as described by the authors, circulate around generating synthetic data, which is a common approach used by researchers when no ground truth data is readily available.

Generally, evaluating the performance of any software product should be done from the end users perspective. This is a well known fact in software development, where user interaction (UI) and user experience (UX) are often evaluated independently from a software functionality. In the case of applying ML to intrusion detection this would mean that an evaluation can still be negative, despite the technique possessing a high classification rate and low false positive rate simply because it does not provide any significant value to the end user of the system. In this context an end user can be any professional who is attempting to identify and investigate intrusion attempts, potentially in real time. Such an analyst would rely on the intrusion detection system to deliver a reliable subset of all existing traffic, containing intrusion attempts. Classification rates do play a significant role in providing value to this analyst. A high rate of false positives would present the main issue that the analyst is trying to solve with the application of ML based classifier, which is to find a relevant subset of data out of an unmanageably large collection of traffic. Similarly, a classifier that contains a large amount of false negatives may make the analysts situation worse as valid intrusion attempts are not being delivered to the analyst by the ML classifier in which the analyst places their trust. With this in mind Sommer & Paxson, advise researchers to take their experimental studies further than training a model and producing the various classification and false positive rates and to manually investigate both false positives and false negatives to determine why they are classified such. This has two benefits, the first being that it forces the authors to develop

a deep understanding of the constructed model, allowing one to evaluate whether the classification criteria make any intuitive sense to a human. The second benefit is that it forces the researcher to use the system from the end users perspective, whose objective is to achieve more than just classifying traffic data, but to investigate, confirm and respond to intrusion attempts.

4 Neural Network Based IDS

The following sections aim to review a detection intrusion detection classifier based on a neural network architecture introduced by Zhang et al. [4]. The paper will be reviewed based on the evaluation criteria described in section 3.

4.1 Artificial Neural Networks Overview

In the context of this paper, an Artificial Neural Network (ANN) is a highly malleable supervised machine learning algorithm that consists of a set of nodes that are connected by weighted edges to form a graph with a predetermined number of input nodes and output nodes.

An individual node can be simply defined by equation (7). Each node has n weighted inputs and the weighted sum of these inputs form the nodes output value.

$$o = \sum_{i=0}^{n} x_i w_i \tag{7}$$

Rules set around the output function determine whether the node activates, given set of inputs and weights. The set of these nodes can be connected and arranged to each other in a specific manner, which determines the ANN's type. Any type of ANN consists of a set of input nodes, output nodes and at least one layer of hidden nodes. The input and output nodes determine how many inputs are passed to the ANN and how many outputs the forward propagation produces. The hidden layer consists of connected nodes, whose inputs are determined from the output of a connected node. The weights are often initially randomised and then adjusted during a training phase which attempts to map the input values to the given output values contain within the training dataset with a minimised error. This is achieved by adjusting each edge weight in some of the nodes. The technique used to perform this training phase depends on the type of ANN used.

4.2 The Reviewed Technique

Zhang et al. test two types of ANN: Back Propagation Learning (BPL) and Radial Basis Function (RBF). These techniques differ based on their criteria to determine whether a node is activated (activation function), and the method used to perform the training step. In a BPL based ANN, the training is achieved via back propagation after the outputs are

calculated given a set of random starting weights, where the nodes of each hidden layer are progressively optimised using a non-linear optimisation technique such as gradient descent [4]. BPL's activation is calculated by passing the output value from equation 7 through a non-linear function such as a sigmoid curve. The RBF approach differs from BPL in both activation and training. In this case activation is performed with a weighted radial basis function in each node. The RBF training stage has two phases where unsupervised learning is used to optimise parameters of the radial basis function and supervised learning is used to adjust the weights of the output nodes only [4].

The authors introduce a system that combines anomaly detection with misuse detection, arranged to form two separate frameworks. In the context of intrusion detection, misuse detection, can be viewed as the inverse of anomaly detection. While anomaly detection establishes a model based on normal traffic and from this classifies outliers of this profile as malicious, misuse detection constructs a profile of malicious or abnormal behaviour and then classifies outliers from this abnormal profile as normal. This paper's primary focus is on the application of these two types of neural networks. Specifically, the review will be based on experiment construction and conclusions drawn from the results by the authors, with regards to the concepts introduced in section 3.

4.2.1 Dataset & Results

The two ANN techniques were applied to the KDD Cup 1999 Data public dataset. The authors asserted that all data points could be grouped into four categories. The separate training and test datasets had 22k malicious packets and 10k benign. At this point it is important to note that the frequency malicious data within the training set is around 69%.

On the testing data the BPL & RBF achieved an anomaly detection rate of 93.7% & 99.2%, the false positive rates were 7.2% & 1.2% respectively. Both techniques achieved misuse detection rates of 99.2% & 98% and false positive rates of 1.2% & 1.6% respectively. On account of the experimental data the authors concluded that RBF can achieve better misuse classification rates than BPL, while anomaly detection gave similar results.

5 Discussion

In this section the approach outlined in Zhang et al. is holistically evaluated based on each of the guidelines outlined in section 3. On a higher level, the goal of this section is to determine whether the information presented in Zhang et al. can demonstrate a clear applicability of the enclosed ANN techniques and frameworks to any specific real world operational setting.

5.1 Threat Model

This performance metric is the most difficult to evaluate the reviewed approach upon as it is very personal to the real world application of the technique. Almost all reviewed publications, tend to promote their technique based on its ability to classify with a low false positive rate on a public dataset, often citing Axelsson as a motivation for this approach. Zhang et al. also took this approach towards this paper. Sommer & Paxson identify that many published intrusion detection approaches are not deployed in a real world setting. We believe this is primarily caused by the lack of a clear threat model that describes the specific environment in which the technique could be applied and profiles a specific attacker and how this attacker will likely interact with the defined environment. This is crucial as an attacker may behave differently, when performing the same attacks against different targets, this in itself can lead to a classification model for each different target hoping to detect such attacks using a specific ML based technique. In other words, the results presented in Zhang et al. are only useful if the application environment has similar traffic patterns to this public dataset. Given that one of the intrinsic challenges of intrusion detection is the high diversity of network traffic at both high and low network protocol levels [3], we believe that the real world value of any published technique without a clearly defined threat model is highly limited.

5.2 Scope

Sommer & Paxson argue that the ideal approach to successfully applying a machine learning technique to IDS, is to focus on a specific niche of intrusion attempts, then determine which general approach would best detect these e.g anomaly or misuse based detection. Finally, this would lead to a small subset of potential techniques within the general approach.

The authors of the reviewed paper however, mention that the intrusion attempts within their chosen public dataset fit into four categories, which aim to cover a large range of different intrusion attempts. This attempt to produce a technique for general intrusion detection attempts appears to violate Sommer & Paxson's guideline. This is illustrated well when contrasted with Kruegel et al. [9], a paper praised by Sommer & Paxson for picking a highly narrow and specific scope: exploiting web servers via specific HTTP queries. On the other hand, the scope of Zhang et al. appears to be only limited by the amount of different attacks contained within their chosen public dataset. This can be viewed as a red flag, possibly indicating that the authors did not attempt to find a problem first, before working on a technique to address the problem, but rather chose to search for problem to solve, given the technique first.

5.3 Cost of Errors

Although defining a narrow scope can go a long way in reducing false positive rates, putting the cost of each false positive into perspective can also aid in determining the tolerability in this area. In general it is clear that the cost of a false positive is higher in the information security when contrasted with other applied machine learning domains [3]. However, no two information security environments are exactly the same, therefore determining the real cost in each environment may help determine the applicability of the chosen technique. The reviewed paper cannot realistically determine how well the proposed technique will apply in terms of risk, to all possible environments. However, looking back at equation (5) we can see that the reviewed paper does provide the probability of a false positive $P(F)$, which is the false positive classification rate. Any interested party can substitute a cost F_{cost} into the equation to calculate a weighted risk that can be applied to compare the risk with other environments.

Another approach to eliminating costs associated with false positives, involves re-evaluating classifications in a post processing stage to further filter them, reducing the amount of false positives that get relayed to the end user. In the reviewed paper the authors take an interesting approach to dealing with the false positives. Initially two types of ANN's are applied to the public dataset individually and their results are assessed. The BPL ANN applied an anomaly detection strategy while the RBF applied a misuse detection strategy. The authors then assert that these two ANN's could be assembled into a filter chain type architecture where packets are run through the anomaly classifier first then through the misuse classifier. Finally the classification model is refined as increasing attacks are added to the system. This multilayer approach appears to be a similar strategy on top of the existing 1.2% false positive rates of the anomaly detection.

5.4 Performance Evaluation

Zhang et al., like many publications in this area base their experiment on a public network traffic dataset (KDD Cup 1999 Data). We have already mentioned in the performance metrics section the shortcomings of such datasets as described by Sommer & Paxson. An IT professional hoping to apply this technique to their specific operational setting, is less likely to rely on the results presented in the paper. Sommer & Paxson assert that public datasets such as the one used in the reviewed study, should only be used for basic testing of a technique, but not as a dataset to perform extensive performance tests upon. While we are not suggesting that a publication should produce a series of datasets that can give an insight into the techniques performance in any organisational environment, more effort could be made to gather a more applicable dataset from an organisation. Sommer & Paxson proposed that researchers build closer relationships with existing organisations to order to negotiate some data. This could involve offering consulting in exchange for the data.

As mentioned Sommer & Paxson suggested that researchers should manually investigate both false positives and false negatives to develop a greater understanding to determine whether the ANN actually models the intuition of a sensible manual classification method. This is especially necessary if there is limited amount of unique testing data to run the trained model against. Which is the case of the reviewed paper where a single set of test data with the exact size of the training data was prepared to produce the results with. The paper did not manually analyse any false positive or false negatives and based their discussion of the methods efficacy mainly around the classification results produced from the testing data. From our point of view this puts the efficacy of the authors proposed technique into question. At this point we can only see how the technique performs on a single test data set, although the authors have correctly split their training and testing data, this does provide only a small testing sample size, the classification rates could change significantly on further test datasets. Sommer & Paxson's technique of manually analysing false positives and negatives in order to determine the intuition of the constructed model, provides a highly recommended alternative, when researchers are unable to acquire more unique testing data.

6 Conclusion

This paper has provided two major contributions. The first contribution is to elaborate and further develop the machine learning application guidelines introduced by Sommer & Paxson as a set of evaluation criteria to determine whether any specific machine learning based intrusion detection classifier can demonstrate the ability to provide real world value in an operational context. This includes the novel application of quantitative risk to introduce a metric to theoretically define and practically compare the individual or cumulative risk of both false positive and false negative classifications in different operational environments. The second contribution involves the review of a publication introducing two neural network based frameworks based on the two distinct classes ANN classes. The goal of the review is to determine whether the publication illustrates a machine learning based intrusion detection classifier that demonstrates applicability to a specific operational setting based on the guidelines described in this paper. This has lead to the conclusion that the reviewed paper authored by Zhang et al. did not adhere to the described guidelines. The author did not define a clear threat model, environment or narrow set of attacks which their introduced technique could address. In fact, the authors scope was limited only to what type of attacks were contained within their chosen public dataset. The results produced by the authors, were taken from just a single testing data set. Furthermore the authors did not attempt to rationalise the constructed model by manually analysing the produced false positives and/or false negatives. Therefore, it is difficult to evaluate the robustness, of the classification rates obtained. We believe that this presents overfitting to be a significant risk, due to the characteristically diverse nature of traffic data, causing

such a type of ML classifier to produce poor classification rates, on a more diverse range of datasets containing only variations of the 4 attacks mentioned. The authors did however, produce a detection and false positive rate (albeit only from one test dataset), allowing a reader to apply our introduced quantitative risk metric to analyse in perspective, the costs of both false positive and false negative classifications.

References

[1] S. Axelsson, "The Base-rate Fallacy and the Difficulty of Intrusion Detection," *ACM Trans. Inf. Syst. Secur.*, vol. 3, pp. 186–205, Aug. 2000.

[2] X. Wu and X. Zhang, "Automated Inference on Criminality using Face Images," *arXiv preprint arXiv:1611.04135*, 2016.

[3] R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," in *2010 IEEE Symposium on Security and Privacy*, pp. 305–316, May 2010.

[4] C. Zhang, J. Jiang, and M. Kamel, "Intrusion detection using hierarchical neural networks," *Pattern Recognition Letters*, vol. 26, pp. 779–791, May 2005.

[5] S. Myagmar, A. J. Lee, and W. Yurcik, "Threat modeling as a basis for security requirements," in *Symposium on requirements engineering for information security (SREIS)*, vol. 2005, pp. 1–8, 2005.

[6] B. W. Lampson, "Computer security in the real world," *Computer*, vol. 37, pp. 37–46, June 2004.

[7] "SIGKDD - KDD Cup." http://www.kdd.org/kdd-cup.

[8] "MIT Lincoln Laboratory: DARPA Intrusion Detection Evaluation." https://ll.mit.edu/ideval/data/.

[9] C. Kruegel and G. Vigna, "Anomaly detection of web-based attacks," in *Proceedings of the 10th ACM conference on Computer and communications security*, pp. 251–261, ACM, 2003.