A cost-sensitive approach for managing intrusion alerts in OT environments

Alex Howe[1], Andrew Morin[2], Mauricio Papa[1], Tyler Moore[2]

[1] Tandy School of Computer Science, The University of Tulsa, Tulsa OK, USA
[2] School of Cyber Studies, The University of Tulsa, Tulsa OK, USA

Abstract. Network Intrusion Detection Systems (NIDS) are traditionally built to minimize the total number of misclassifications without considering financial implications. However, false positives and false negatives both impose monetary costs on an organization through wasted analyst time and damage from missed attacks. This work presents an approach which uses economically informed decision making to develop a cost-sensitive intrusion detection architecture that incorporates the cost of handling such misclassifications. Specifically, we propose a cost-sensitive supervised machine learning model alongside an economically informed thresholding technique to minimize the overall cost when dealing with cyber attacks. The models are evaluated across four unique scenarios in two environments, highlighting the broad suitability of the architecture. The various scenarios allow our architecture to be evaluated across a range of notoriously difficult to determine costs. Experimental results for the two domains demonstrate an average cost reduction of 59% over traditional accuracy-based intrusion detection systems. The trade-off, measured in reduced accuracy, is minor, with an average accuracy reduction of 1.25%. Our architecture allows organizations to make detailed and informed decisions about resource allocation when implementing security tools.

Keywords: Critical infrastructure · Intrusion detection · Security economics.

1 Introduction

Operational technology (OT) networks operate and manage critical infrastructures such as refineries, power plants, and water treatment facilities, where the integrity of such systems is paramount. These networks have traditionally relied on physical isolation and security through obscurity to mitigate exposure to cyber threats. However, in recent years they have become increasingly interwoven with public networks in pursuit of increased efficiency [9]. The accelerated rate at which these systems are being connected to public networks has outpaced the cyber security response. The landmark ransomware attack on several Colonial Pipeline systems in May 2021 highlighted the potential consequences of continuing to integrate OT networks with public networks in the absence of sufficient cyber security polices.

To combat the expanding threat landscapes, intrusion detection systems (IDSs) are used to identify and stop attacks. The goal of the IDS is to analyze network traffic and attempt to classify incoming traffic as benign or malicious. These IDSs are predominantly evaluated on their ability to minimize false positive and false negative classifications. While both of these classification errors equally affect the accuracy of the IDS, the financial impact of false positives and false negatives can vary widely. For cyber security teams operating on a finite budget, the economic efficiency of their reaction to potential threats is of utmost importance. This paper presents a framework to incorporate economic information into the intrusion detection and response process. Specifically, we apply cost-informed weights to the machine learning process, as well as a cost-based thresholding technique, to influence the detection model's decision making.

Cost-informed weights are used to perform cost-sensitive learning, a subset of machine learning which is dedicated to scenarios with varying misclassification penalties [4]. Consider a buffer overflow attack which costs a company $10,000 if it is successfully exploited, yet only $20 for an analyst to review the traffic. The cost of a false positive, or a nonmalicious packet flagged as malicious, is inexpensive compared to the false negative, a malicious packet incorrectly labeled as nonmalicious. Thus, if the IDS generates 200 false positives in order to correctly identify 1 buffer overflow attack, the organization still saves a total of $6,000. Integrating a cost-sensitive approach into the machine learning-based IDS allows for an organization to influence the model's learning process to force it to focus on high-priority attacks.

Thresholding techniques identify optimal decision boundaries for machine learning classifiers. This work introduces a cost-based thresholding technique referred to as a *scoring manager*. This method analyzes the probability values from each of the classes as well as their associated false positive and false negative costs in order to determine thresholds for each class. The scoring manager then creates an economic filter which can be used in combination with the cost-informed weights to optimize the expected cost.

Cyber security costs are notoriously difficult to study, as they can vary widely between, and even within, industries. To combat this, we apply our framework to four scenarios across two OT environments. The scenarios represent unique salaries, expected financial impacts, and cyber security maturity levels. We find that our framework consistently reduces costs incurred by the organizations in all scenarios. The overall costs are reduced in all of the eight scenarios, with an average reduction of 59%. The loss in accuracy to achieve these reductions is minor, with an average accuracy loss of 1.25%.

In Section 2, we review related literature. Section 3 details the architecture used. In Section 4, we provide the data used, as well as detail our methodology. In Section 5, we detail our findings and in Section 6 we provide conluding remarks.

2 Related Work

The approach presented in this paper combines ideas from two fields: cost-sensitive machine learning and security economics. On the one hand there is a need to develop effective intrusion detection systems to minimize the likelihood that an attack succeeds. On the other, there is a need to develop solutions that allow stakeholders to make economically informed security decisions. This section presents work related to these two fields.

Cost-sensitive intrusion detection has emerged as one solution to handling the inherent class imbalance of network intrusion detection. One popular approach is to utilize cost-sensitive machine learning and weight the minority classes according to class distribution. For example, in [14] the authors propose an ensemble approach based on Deep Neural Networks (DNNs) which use class minority distributions to influence the training process. Specifically, they apply a bootstrap aggregation approach in which ten DNNs are trained on unique training subsets, which are balanced by weighting the minority class based on the ratio when compared to the majority class. In [6], the authors propose a cost-sensitive IDS based on the XGBoost algorithm in which attack classes are weighted based on their probabilities, or ratio compared to the majority. They compare their cost-sensitive weighting approach to the SMOTE oversampling algorithm.

In [5], the authors propose a cost-sensitive ensemble approach in which both a weighted training scheme and an oversampling method is used to identify network intrusions. Three layers, based on the cost-sensitive DNN, Random Forest, and XGBoost algorithms, are proposed which have varying levels of detection granularity. The first layer incorporates a cost-sensitive DNN in which class ratios are used to weight the minority and majority classes. The second and third layers rely on oversampling techniques to rebalance the distributions.

In all three related works, the authors rely on class distributions to derive the weights. This work proposes an economically informed weighting scheme, allowing the user to tune the IDS operations to fit the organization's requirements.

An economic approach has proved useful in understanding and managing cybersecurity risks [2]. Typically, security controls are developed and evaluated on their technical merits alone, such as their effectiveness in detecting and preventing attacks. But cybersecurity investment comes at a cost, and the interventions they introduce do too. Restricting access to networks and systems may reduce the likelihood of an attack, but it can also make completing a mission more difficult and expensive. By quantifying the costs and benefits of cybersecurity controls, it becomes possible to rationally evaluate cybersecurity investments and configure their operations in a way that optimally manages risk.

For example, many organizations balk at making investments to strengthen the cybersecurity of industrial control systems, despite the presence of long-standing weaknesses in these systems. One way to encourage investment is to demonstrate that the benefits outweigh the added costs. Papa et al. conducted a cost-benefit analysis for retrofitting wastewater facilities with an ICS attack detection system [11]. They estimated the costs associated with a successful attack using public data on harms, and then showed the circumstances under which

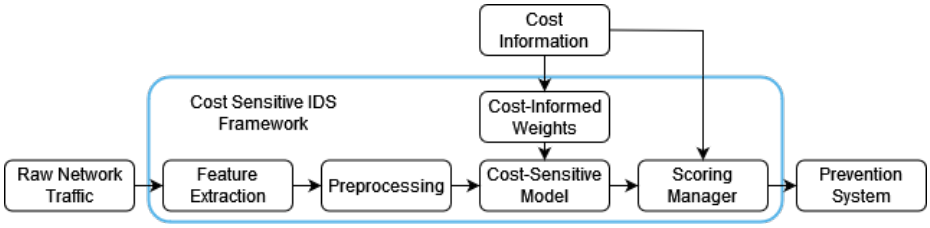

Fig. 1. Proposed cost-sensitive IDS architecture diagram.

an investment in a detection system to reduce the likelihood of accidental or intentional overflows could be justified.

The notion that IDSs must not only optimize security but also minimize cost has been proposed before. In 2002, [7] explored the problem of building cost-sensitive IDS. Their solution considers five different types of cost: development, operation, damage (when an intrusion is successful), manual response and automated response to attacks. They define a model to formulate the total expected cost of the IDS (using all five costs) and propose cost-sensitive ML-based techniques that are designed to reduce the overall cost of intrusion detection.

Another approach is to evaluate output from binary classifiers. Receiver operating characteristic (ROC) curves are commonly used to evaluate the diagnostic ability of a binary classifier system as its discrimination threshold is varied. ROC curves have also been used to evaluate cyber security costs. [10] investigated the use of ROC curves to optimize filters that ultimately define whether there is a need to respond to an intrusion and the associated cost. The advantage of the approach is that it helps identify a more optimal allocation of resources that minimizes overall expected cost.

The proposed IDS architecture relies on cost-sensitive machine learning, but takes an entirely different financial-based approach when it comes to class weighting. Additionally, we incorporate an economically informed thresholding technique which adds an additional layer of expected cost optimization.

3 Cost-sensitive OT IDS Architecture

The machine learning-based architecture was designed for real-time intrusion detection using the multi-stage approach shown in Figure 1. Four sequential stages analyze incoming network traffic and generate security alerts for malicious activity: feature extraction, preprocessing, machine learning and score managing.

The detection framework uses a cost-sensitive machine learning model, which is trained offline, to generate predictions/security alerts for each packet sent over the network. Security alerts are sent to an analyst for review, thus the architecture can work asynchronously apart from the network reducing any impact on bandwidth. Each stage of the framework contains tunable parameters which can be adjusted to meet the needs of any particular network it is deployed on.

3.1 Feature Extraction

Feature extraction and engineering refers to the process of transforming raw data and creating new features or variables suitable for use in machine learning models. In this particular domain, feature extraction refers to the process of extracting meaningful features from raw packets captured on the OT network. These features convey not only the state of that particular packet but may also provide information about the overall state of the OT system.

Network packets are messages formatted and sent over an established network. These messages are formatted using a protocol stack used to give context to the data being sent. Each protocol in the stack appends a header to the packet (and sometimes a trailer), providing incrementally more information. Feature extraction is a two-step process that identifies basic and statistical features. Basic packet features are extracted directly from from packet headers, such as addresses, port numbers, and control flags. These features are crucial to understanding the context of a particular packet within a network. In contrast, statistical-based metrics provide insight about the network at-large, and help characterize communication patterns. This includes information such as transmission rates for packets as well as average packet sizes. These are calculated using a running average for variables of interest in a sliding window with a configurable time width (e.g. one second).

3.2 Pre-processing

Feature pre-processing involves the transformation of features/variables to optimize them for use with machine learning algorithms. These transformations include handling any missing values, re-scaling data to remove biases, and restructuring features to match the assumptions made by some learning algorithms. The first step of pre-processing involves handling missing values which occur when the framework attempts to extract information related to a protocol that the captured packet does not use. Missing values are imputed with zeros, which signify the absence of a particular feature from the packet. For example, a TCP related feature would be set to zero when analyzing a UDP packet.

State variables, such as TCP control flags, are one-hot encoded to binary representations in order to retain their categorical information. For discrete structured variables, such as addresses, the encoding scheme segments the variables into multiple features by a protocol define separation character. For instance, IP address 192.168.33.56 is encoded into four decimal features with individual values 192, 168, 33 and 56 respectively. These values are treated as continuous and can offer insight into additional information such as identifying subnets and unique nodes. In Shao's work [12], this splitting technique was found to be much more effective than the standard one-hot encoding method.

Continuous variables, such as statistical-based features, are individually rescaled to have zero-mean and a unit variance. Re-scaling features to a common scale reduces bias while retaining relationships between the different variables. Finally, a basic feature selection method using a variance threshold of 0 is used during training to remove any unimportant features.

3.3 Machine Learning

Machine learning has become a promising candidate for developing robust and efficient intrusion detection systems. Three primary learning techniques are commonly used for intrusion detection: supervised, semi-supervised, and unsupervised. Supervised and semi-supervised algorithms use labeled data for training in order to predict the likelihood of a sample belonging to a class; unsupervised algorithms do not require labels and instead work to uncover latent patterns.

These detection models often interface directly with humans, thus additional performance metrics such as false positive rates and the incurred monetary costs must be considered. Ideally, the organization would take the machine learning-based IDS and implement biases into it allowing them to shape the tool to their specific requirements or security objectives. Thus, this work considers the use of supervised machine learning algorithms to create cost-sensitive intrusion detection systems that adhere to economic requirements.

Cost-Sensitive Supervised Models Similar to traditional signature based IDSs, the supervised model is trained using labeled attack samples and attempts to classify these attacks in real-time. One assumption made in most multi-class classifications tasks is that all errors are equally undesirable. This assumption is not necessarily true for several environments including intrusion detection, where a false negative and false positive can have very different impacts.

Cost-sensitive machine learning leverages cost values to accommodate for imbalanced datasets or differences in misclassification types. Specifically, we can use economically informed weights to influence the training process of the cost-sensitive detection model forcing it to prioritize attack classes associated with higher costs/weights. Four cost-sensitive implementations of traditional supervised algorithms are explored in this work: random forest, support vector machines, XGBoost, and dense neural networks.

Cost-sensitive Random Forest involves factoring weights W into the Gini index/impurity equation:

$$1 - (w_0 * p_0^2 + w_1 * p_1^2 + ... + w_n * p_n^2) \tag{1}$$

Where w_i and p_i is the economically informed weight and probability for attack class i respectively. By weighting each attack class we can adjust the impact that a misclassification for that particular class has on the model's decision making.

Cost-sensitive Support Vector Machines (SVM) factor economically informed weights into the primal objective function:

$$Min(\frac{\omega^T \omega}{2} + C \sum_{i}^{n} (w_i * \xi_i)) \quad subject\ to\ \xi_i \geq 0 \tag{2}$$

where C is a regularization parameter and ξ_i is the misclassification allowance for class i. SVM can be applied to multi-class classification by taking a "one-vs-one" approach in which multiple binary classifiers are trained to separate pairs of classes.

Both extreme gradient boosting (XGBoost) and deep neural networks (DNNs) train using a defined objective function, or loss function, which tunes the parameters based on classification errors. For multi-class classification tasks one common objective function used is the multi-class Cross Entropy loss function which minimizes the negative log-likelihood over the classes. Cost-sensitive implementations of XGBoost and DNNs can be achieved by including a weighted version of the multi-class Cross Entropy loss function.

$$-\sum w_i * y_i * \log(\hat{y}_i) \tag{3}$$

Where w_i, y_i, and $\hat{y}_i$ represent the economically informed weight, true class, and predicted class for the ith sample.

Cost-informed weights Traditionally, supervised detection models prescribe equal weight to all classes and train the model to minimize the total number of false positive and false negatives. To emphasize a reduction in cost over accuracy, we consider an economically informed approach which factors the financial costs incurred by the generated alerts. Specifically, we employ a cost-sensitive supervised detection model in which the class weights w_i are developed based on the misclassification cost for attack i.

By weighting the importance of alerts based on the cost of misclassification, the optimization of the model is shifted from a purely accuracy based metric, to a minimization of costs. This also introduces flexibility in the model, allowing it to adapt to different environments where the same error may have varying costs. There are several methods for developing cost-informed weights; this work proposes an approach in which the misclassification costs are normalized and applied as weights. Normalization preserves the weights association with exploitation and investigation costs.

3.4 Cost-based Scoring Manager

When confronted with real-time network traffic, supervised and unsupervised models alike will provide a likelihood score corresponding to each attack, which represents the confidence of the model that the packet is associated with an attack. A difficulty with machine-learning based models is to choose a threshold above which alerts are considered to be valid. A threshold which is too low will result in benign traffic being caught in the filter, and a threshold too high will allow malicious traffic through.

Given a sample of network traffic alerts, there are four possible classification outcomes for each alert: false positive, false negative, true positive, and true negative. The circumstances for these outcomes are shown in Table 1, as well as the false negative rate, β, and the false positive rate, α. To identify the optimal filter threshold which minimizes both β and α, the sample traffic, with known labels, can be evaluated at a series of thresholds.

As discussed above, accuracy alone is not guaranteed to result in cost-effective configuration. In addition to the training and selection of machine-learning models, economic information can also be used to make cost-informed decisions about

Prediction	**Reality**	
	Malicious	Benign
Malicious	True Positive	False Positive
Benign	False Negative	True Negative
Rate	$\beta = \frac{FN}{(FN+TP)}$	$\alpha = \frac{FP}{(FP+TN)}$

Table 1. Confusion matrix of all possible outcomes.

how to respond to alerts. This is further motivated by the varying cost of false positive alerts. Certain alerts will require a more detailed and time-consuming investigation to be cleared, yet this variation in false positive cost is not accounted for in the model training phase. To incorporate economic information into alert response, we use an optimal filter configuration identification process outlined by [3]. Using this process, the cost and probability associated with classification errors can be factored into alert response for a cost-informed response. This can be done using the following equation:

$$\alpha^* = \arg\min_{\alpha} p \cdot \beta(\alpha) \cdot b + (1-p) \cdot \alpha \cdot a \tag{4}$$

where α^* represents the false positive rate associated with the minimum overall cost. The cost itself is determined by multiplying the prior probability of the malicious traffic, p, by the false negative rate, $\beta(\alpha)$, and the cost of a false negative, b. This is added to the product of the probability of the traffic being benign, $1-p$, the false positive rate, α, and the false positive cost, a. Taking the first order condition of Equation 4 gives us the slope of the indifference line where the costs of each error are equal. That is,

$$\beta'(\alpha^*) = -\frac{1-p}{p} \cdot \frac{a}{b} \tag{5}$$

where the optimal model configuration will be where this indifference line crosses the ROC curve. By determining the optimal threshold for each alert in the training sample, the models can be compared for any alert to select the lowest expected cost response.

4 Datasets for Evaluation

One of the main challenges for intrusion detection research in the OT domain is the lack of available quality datasets. Several key works have been proposed to help mitigate the lack of available OT network data including using data captured from deployed systems, testbeds [1], and simulated environments [8]. Real time deployed environments offer the most accurate source of data when it comes to OT network behavior. However, the lack of validation when it comes to the ground truth of the data makes the data unreliable. On the other hand, simulated environments can be properly validated but can lack the characteristics

of deployed OT environments. Testbeds for physical OT networks are the ideal method for generating OT security data as they can exhibit the noise and characteristics of deployed OT environments while allowing proper validation when labeling attacks. Additionally, testbeds offer the ability to implement attacks and record the results in real time, which is not possible for deployed environments as they are often used in critical infrastructure.

This work utilizes two environmental datasets of raw network captures generated using a physical testbed in the additive manufacturing domain and a simulated OT environment of an electrical network. With each dataset several types of commonly encountered attacks were implemented in real-time and the packets related to those attacks were labeled after the capture. The model is trained using labeled examples of these attacks.

Our proposed cost-sensitive IDS architecture is evaluated on four unique scenarios within each environment. These scenarios illustrate varying costs and organizational cyber security maturity, allowing us to verify consistent performance across multiple applications.

4.1 OT Environment 1: Additive Manufacturing

The additive manufacturing testbed is a 3D printing system consisting of five devices connected over a closed Ethernet network. Four of the nodes are operational and work together to manufacture parts using various metals and alloys, the fifth node is a server acting as the target of the implemented attacks. Operating over the MQTT protocol, the network utilizes the publish-subscribe method for transmitting data. The workstation, one of the operational nodes, is used by an operator to send job files and commands to the other devices on the network.

Four datasets were generated from this OT environment consisting of 24 hours of normal traffic, as well as a scan attack, a MitM attack, and an anomaly attack. The scan attack was created by performing a network scan of the OT environment. The MitM attack was created by poisoning the ARP-cache allowing them to observe the network traffic. Finally, the anomaly attack is created by performing an ICMP ping sweep from a compromised server. Detailed description of the packet distribution is given in Table 2.

4.2 OT Environment 2: Electric Utility

This environment is characterized by a data set described by [8] and it corresponds to a small electrical network. The network has controllers (or RTUs) that are in charge of electrical circuits, each with a single supply branch operating at 12,000 V. Controllers provide voltage measurements on each branch to an MTU using the Modbus/TCP protocol.

Tests were conducted using datasets that involved one MTU and 6 RTUs. Separate Modbus traffic files containing both normal and malicious traffic were used in the training and evaluation of the IDS. The first file contains only polling commands from the MTU to the RTUs represents the normal operational behavior of the environment. The remote exploit attack involves an actor using

Additive Manufacturing					
Normal	Scan	MitM	Anomaly	-	Total
76,195	21,303	410	34	-	97,942
Electrical Utility					
Normal	File	Upload	Anomaly	Remote	Total
74,687	75	1,199	10	121	76,092

Table 2. Dataset packet distributions.

Metasploit's MS08-netapi exploit to compromise an active RTU. In file transfer, the actor uses the compromised RTU to transfer files to two other RTUs. The upload executable attack records the actor using the compromised RTU to upload an executable file to another RTU. In the anomalous attack, the actor uses the compromised RTU to forge and send fake Modbus commands to other RTUs. Detailed description of the packet distribution is given in Table 2.

4.3 Evaluation Scenarios

The IDS architecture proposed in this paper is evaluated on its ability to reduce overall costs associated with the misclassification of alerts compared to a strictly accuracy-based IDS. Therefore, we must first assign a cost to each misclassification of a packet. Such costs are challenging to identify due to the inherent differences between organizations, environments, and the changing threat landscapes over time. Therefore, instead of constricting the IDS evaluation to a single, limited scenario, we evaluate the IDS across a variety of scenarios representing different potential organizations. In total, we introduce four scenarios to be evaluated in each of the two environments: baseline, amplified attack, uneven harms, and high salary. The purpose of these scenarios is to illustrate how varying costs can affect decision making in the model. In practice, we would expect operators to provide cost estimates reflecting their deeper knowledge of deployment realities. When calculating the expected cost we take the sum of all false positives and false negatives multiplied by their respective costs.

Baseline Scenario First, we look to open-source data and prior work to identify the "baseline" costs for a single implementation. Morin and Moore [10] used open-source information to estimate the misclassification costs for similar OT environments. For false positives, they define the cost as the cyber security analyst time wasted investigating spurious alerts. Combining the Bureau of Labor Statistics median salary for a cyber security analyst with the industry reported rate of alert review by analysts, Morin and Moore estimated a single false positive would cost an organization $10.

The OT environments evaluated in [10] are specific implementations of our additive manufacturing and electric utility environments, therefore we adopt

Scenario	FP	Costs			
		Scan	MitM	Anomaly	-
		File	Upload	Anomaly	Remote
Baseline	$10	$67	$67	$67	-
		$2,500	$2,500	$2,500	$2,500
Amplified Impact	$10	$6,667	$6,667	$6,667	-
		$250,000	$250,000	$250,000	$250,000
Uneven Harms	$10	$195.60	$4	$0.40	-
		$480	$8,490	$920	$110
High Salary	$50	$195.60	$4	$0.40	-
		$480	$8,490	$920	$110

Table 3. The four scenarios are shown in the first column. Each row has the estimated false positive (FP) and false negative costs. The false negative costs are split into additive manufacturing above the dashed line, and the electric utilty costs below the dashed line.

their cost estimates. For the additive manufacturing environment, they identify four 3D printers with hourly titanium alloy printing costs. The impact in this environment is captured by an hour of lost printing material. We take the average cost of all four printers, resulting in an estimated attack cost of approximately $20,000. Although a false negative alert poses a risk, not all false negatives will result in a successful attack. Therefore, we choose a 1% probability that a false negative will result in a loss, resulting in an estimated false negative cost of $200. As there are three potential attack vectors for the additive manufacturing environment (scanning, Man-in-the-Middle, and anomaly), we evenly distribute the estimated impact across each. That is, the false negative cost for each malicious packet type is $200 divided by three, or $67.

For the electric utility environment, [10] measure the impact as the financial loss resulting in a momentary electrical outage in four different U.S. cities. We again take the average cost from the four samples, which provides an estimated cyber attack cost of approximately $1,000,000. Similar to the additive manufacturing attacks, it is reasonable to assume not all false negatives will result in a successful attack, and we again take 1% of this cost. As a result, the cost of a false negative in the electric utility environment is $10,000 divided evenly across the four packet types, resulting in a false negative cost of $2,500. Note that the electric utility environment has four malicious packet types versus the three in the additive manufacturing environment.

Amplified Impact Scenario The second scenario we evaluate is the "amplified impact" scenario. In this scenario we leave the analyst salary unchanged, while raising all attack impacts. An electric utility example of such a scenario could be the introduction of industrial and commercial losses to the outage impact.

Existing literature points out that the majority of losses from an electrical outage are from industrial and commercial customers. [13] As a utility provider becomes more informed of the true cost, the impact may become amplified while the analyst salary remains unaffected. For this scenario, we take the baseline scenario false negative costs and multiply them by ten.

Uneven Harms Scenario In the first two scenarios we have evenly distributed the estimated false negative costs across malicious packet types. The underlying assumption is that the organization is uninformed about which packet types are most likely to appear. The reality is that this is likely untrue, and certain malicious packets will appear more often. In the third scenario, "uneven harms", we consider an organization which is well informed about the appearance rate of malicious packet types. In this case, we multiply the baseline costs by the relative appearance rates of each packet type in the training data. For additive manufacturing, Man-in-the-Middle packets make up 2% of all malicious packets, and therefore we multiply the $200 original cost by 0.02 for a cost of $4.

High Salary Scenario Finally, we consider the "high salary" scenario when analysts are paid a higher salary, resulting in costlier false positives. For this scenario we maintain the costs from the uneven harms scenario, and multiply the false positive cost by five, resulting in a false positive cost of $50. This could be a scenario in which more experienced analysts are required, or that more analysts are involved in alert remediation.

5 Evaluation

We begin our evaluation by comparing the cost-informed results of four different machine-learning classification models to identify the model best suited for cost-informed weight information. Next, we measure the relationship between accuracy and cost as the machine-learning weights are gradually adjusted to align with the relative cost of each type of misclassification. While a cost reduction in one scenario is interesting, we proceed to evaluate the consistency of these results by applying these cost-informed weights to four unique scenarios across both environments. Finally, we combine the cost-informed weights of the best model with the scoring manager to measure the overall performance of our proposed IDS architecture.

5.1 Cost-Sensitive Model Selection

The four classifier models described in Section 3 each incorporate the economic information differently. Because of this, we start our analysis by evaluating the cost performance differences between them. The expected cost of a model is determined by training each model on the cost-informed weights and summing

	SVM	Random Forest	XGBoost	DNN
Additive Manufacturing				
Baseline	$1,232.00	$1,290.10	**$1,097.55**	$1,105.42
Amplified Impact	$741,109.06	$108,376.47	**$42,631.35**	$78,980.45
Uneven Harms	$26,088.60	$26,851.32	**$9,564.45**	$13,323.30
High Salary	$27,269.10	$21,729.38	$14,130.59	**$14,059.36**
Electrical Utility				
Baseline	$181,910.00	$59,330.00	**$840.00**	$92,043.00
Amplified Impact	$10,146,493.33	$4,166,760.00	**$28,896.66**	$38,774,440.00
Uneven Harms	$61,116.44	$26,743.62	**$9,364.88**	$48,819.28
High Salary	$68,186.44	$25,176.52	**$14,011.62**	$49,712.96

Table 4. Cost comparison of the four proposed models

up the cost of each classification error during the testing phase. We perform this test for each scenario in both environments. The results can be seen in Table 4.

The data was split using a 70/30 train-test split and the recorded values are averaged over three different controlled seeds. It is important to note that the incurred cost values are entirely dependent on the proposed scenarios, meaning that scenarios with overall low attack costs will have lower expected costs than scenarios with higher attack costs (e.g. scenarios 1 and 2).

We find the cost-informed XGBoost model achieved the best performance in terms of minimizing the overall expected cost in seven of the eight scenarios. The single outlier was the fourth scenario in the additive manufacturing environment. In this scenario the deep neural network achieved a marginally lower cost than XGBoost. Although all models had positive results, we find the XGBoost model was best suited for cost-informed weights. Therefore, any evaluations henceforth reported on will be done using the XGBoost classifier model.

5.2 Cost-Informed Weights

Before testing each scenario, we first seek to measure the interaction between the costs and accuracy as the model transitions from a purely accuracy-based model to a cost-informed model. Prior to training, the costs are normalized to a scale of $[0, 1]$ weights. The accuracy-based model, which prioritizes a minimization of misclassifications, evenly weights each error type. Specifically, for the additive manufacturing environment with four possible packet types, the weights are all 0.25. For the electric utility environment with five packet types, these weights are all 0.2. The final cost-informed weights, which will differ from the even weights to varying degrees, are also normalized to $[0, 1]$. By breaking up the difference between the accuracy and cost-informed weights into a set of 100 equidistant intermediate weights, we can observe the relationship between accuracy and cost as the weights progressively become cost-informed. The result of this process can be seen in Figure 2. The left two plots show the cost and accuracy as the weights

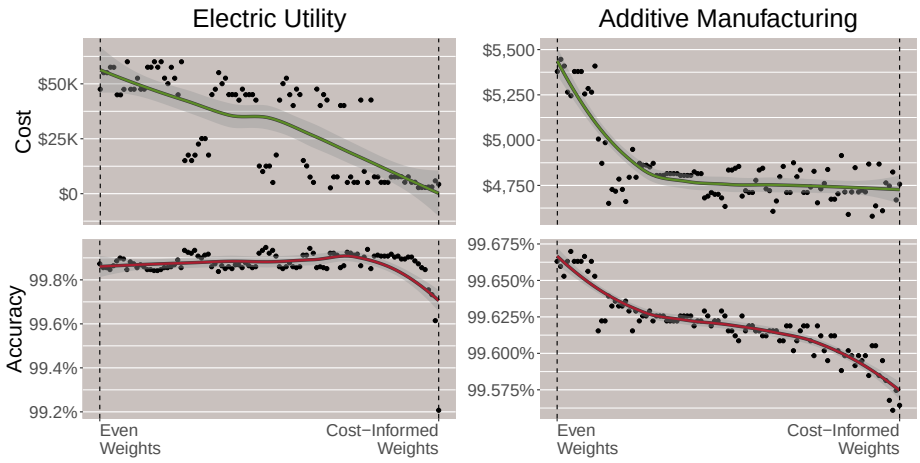

Fig. 2. The cost (top) and accuracy (bottom) for the electric utility (left) and additive manufacturing (right) environments using the baseline cost estimates. The left side of the plots are evenly weighted models, and the right side is the fully cost-informed weights.

progress from evenly weighted to cost-informed weights for the electric utility, and the right two plots show the same information for the additive manufacturing weights. For both environments we used the baseline scenario for cost estimates.

The plots in Figure 2 illustrate the trade-off between cost and accuracy for both environments. The electric utility costs fall steadily and significantly, from an initial cost of $62,560 at even weights, to $6,790 at the cost-informed weights for an 89% decrease. The change in accuracy is a noticeable, but relatively minor 0.66% decrease. For the additive manufacturing environment, the change in cost drops from $5,487 to $4,737, for a 14% decrease. Again, we see a relatively minor accuracy drop of 0.09%. Both sets of plots are experiencing a decrease in costs as the model permits an increase in the relatively cheaper false positives to catch more of the costly attacks. In the next section we will evaluate how these costs are handled across all scenarios.

5.3 Cost-Informed Scenarios

We now investigate how the performance of a cost-informed model changes across our four proposed scenarios. For robustness, we test our scenarios using twenty unique splits of the data for each scenario. In total, we train and test the model 80 times. While each scenario is unique, the direct comparisons between the baseline and amplified impact scenarios, as well as the uneven harms and high salary scenarios, are particularly interesting. The primary difference between the baseline and amplified impact scenarios are the general false negative costs. In both scenarios the analyst salary remains constant, and the organization remains equally uninformed about attack likelihoods, yet the relative cost of false

	Change(%)		Misclassification Change (%)			
Scenario	Accuracy	Cost	Normal	Scan	MitM	Anomaly
Baseline	-0.07	-15.35	1.91	-0.11	-0.25	-0.25
Amplified Impact	-9.5	-83.77	125.8	-0.56	-0.94	-0.5
Uneven Harms	-0.17	-16.96	0.33	-0.33	0.51	6.75
High Salary	-0.18	-23.44	-0.54	-0.20	0.79	6.75

Table 5. The four scenarios for the additive manufacturing environment are shown in the first column. The second and third columns show the accuracy and cost change as a percentage compared to even weights. The last four columns show the misclassification change as a percentage compared to even weights, where a positive number represents a decrease in accuracy.

negatives becomes larger. As such, missing a legitimate attack is amplified by a factor of ten and an organization will tolerate ten times more false positives than in the baseline scenario. This comparison highlights the false negative aversion of an organization as false negative costs increase relative to false positive costs. For the uneven harms and high salary scenarios, the analysts are well informed about packet likelihoods. However, the false positive cost increases in the high salary scenario as analyst time is costlier. Therefore, these scenarios highlight the false positive aversion introduced by higher investigation costs.

In the remainder of this section we will explore each of these relationships in both environments. The additive manufacturing results can be seen in Table 5, and the electric utility results can be seein in Table 6.

Additive Manufacturing False Negative Aversion Observing the baseline scenario from Table 5, we see a median decrease in cost of 15.35%, and a median decrease in accuracy of 0.07%. These values align with the single instance evaluated in the previous section and shown in Figure 2. The amplified impact scenario, where the cost of a successful attack is 10 times larger, shows a larger decrease in median accuracy of 9.5%, and an 84% decrease in median cost. The last four columns highlight where the change in accuracy is most prevalent. In the baseline scenario a 1.9% increase in false positives over even weights is tolerated to reduce a minor reduction in each false negative rate. Compare this to the amplified impact scenario where false positive rates increase 126% over even weights to push the malicious scan, MitM, and anomaly false negatives down by 0.56%, 0.94% and 0.5% respectively. Because a single false negative is an overwhelming cost driver, this relatively minor decrease at the expense of a significantly higher false positive rate remains cost-effective.

	Change(%)		Misclassification Change (%)				
Scenario	Accuracy	Cost	Normal	File	Upload	Anomaly	Remote
Baseline	-0.7	-79	65	-1	-0.89	-0.75	-1
Amplified Impact	-9.96	-89.6	920.7	-1	-1	-0.86	-1
Uneven Harms	-1.08	-85.44	85.67	0.33	-1	-0.56	0
High Salary	-0.12	-77.42	12.88	0.33	-0.88	-0.56	0

Table 6. The four scenarios for the electric utility environment are shown in the first column. The second and third columns show the accuracy and cost change as a percentage compared to even weights. The last five columns show the misclassification change as a percentage compared to even weights.

Additive Manufacturing False Positive Aversion The uneven harms and high salary scenarios are shown in the last two rows of Table 5. In both scenarios we see the median accuracy remain largely unchanged, dropping by 0.17% in the uneven harms scenario, and 0.18% in the high salary scenario. In the uneven harms scenario the well informed cyber security team is able to reduce the median costs by just under 17%, which is better than the median baseline cost from an uninformed cyber security team. This is highlighted in the misclassification columns where we see the relatively uncommon anomaly and MitM traffic is often ignored to improve the accurate classification of the much more common scanning traffic, even at the expense of increased false positives. For the high salary scenario, the analyst salary is five times higher, resulting in costlier false positives. As a result, the model improves the median cost by 23% over even weights by including the prioritization of minimizing the costly false positives in addition to the common scanning traffic.

Electric Utility False Negative Aversion The false negative aversion within the electric utility environment can be seen in Table 6. In the first two rows we again see the median estimated costs decrease significantly for both scenarios. The costs decrease from 79% in the baseline scenario, to 89.6% in the amplified impact scenario as the attacks become costlier. The median accuracy also drops, with a 0.7% decrease in accuracy in between even weights and baseline weights, to a 10% for the amplified impact scenario. In the misclassification columns we see that this decrease in accuracy is the result of increased misclassification in the less costly false positives. In the baseline scenario false positives increase by 65% over even weights, while the amplified impact scenario results in a 920.7% increase in false positives. This increase in false positives results in a decrease in false negatives for the much more expensive upload and anomaly attacks.

Scenario	Metric	Cost-Informed Weights		Scoring Manager		Both	
Baseline	Accuracy	**0.9989**	**(-0.02%)**	0.9497	(-4.94%)	0.9987	(-0.04%)
	F1-Score	**0.9980**	**(-0.04%)**	0.8991	(-9.94%)	0.9974	(-0.09%)
	Cost	$1,246	(8.21%)	$15,095	(1211%)	**$1,098**	**(-4.65%)**
Amplified Impact	Accuracy	**0.9403**	**(1.06%)**	0.4881	(-51.15%)	0.9379	(-6.12%)
	F1-Score	**0.8810**	**(-11.75%)**	0.5386	(-46.04%)	0.8768	(-12.17%)
	Cost	$50,807	(-52.41%)	$172,599	(61.07%)	**$42,631**	**(-60.06%)**
Uneven Harms	Accuracy	0.9906	(-0.86%)	0.9414	(-5.78%)	**0.9939**	**(-0.52%)**
	F1-Score	0.9798	(-1.85%)	0.8829	(-11.56%)	**0.9872**	**(-1.11%)**
	Cost	$10,685	(-83.24%)	$44,709	(-29.85%)	**$9,564**	**(-84.99%)**
High Salary	Accuracy	**0.9955**	**(-0.36%)**	0.9494	(-4.97%)	0.9947	(-0.44%)
	F1-Score	**0.9908**	**(-0.76%)**	0.8986	(-9.98%)	0.9889	(-0.95%)
	Cost	$19,962	(-68.84%)	$101,610	(58.58%)	**$14,131**	**(-77.94%)**

Table 7. Accuracy, F1-Score and cost change (and percent change) from baseline configuration for each IDS architecture configuration across all four scenarios in the additive manufacturing environment. The best improvement in each metric is bolded.

Electric Utility False Positive Aversion The bottom two rows of Table 6 show the uneven harms and high salary scenarios for the electric utility environment. Both show improvement in median cost reduction, again showing a decrease in false positives in the high salary scenario compared to the uneven harms scenario. However, unlike the additive manufacturing scenario, we see the median accuracy increase in the high salary scenario relative to the uneven harms scenario. As a result, the median cost reduction is lower for the high salary scenario. This is a result of the increased false negative costs in the electric utility environment. As a result, false positives are still largely preferred over false negatives and the model slightly reduces sensitivity to upload attacks to increase accuracy. Still, the median cost reduction for high salary is a 77% improvement over even weights.

5.4 Cost-Sensitive Framework Evaluation

The cost-informed weights are the first step of introducing economic information into the proposed IDS architecture. After the models are trained on these weights, the classification scores are then passed to the second economic stage: the scoring manager. The scoring manager assesses the likelihood scores and filters alerts based on the optimal filter configuration described in Section 3.4. In this section we add the scoring manager to the cost-informed weights to evaluate the performance of our IDS architecture. The IDS architecture is evaluated in four configurations: no economic information, cost-informed weights only, scoring manager only, and finally the combination of cost-informed weights and the scoring manager.

The first configuration is no economic influence, and uses a traditional supervised model and even weights which minimize misclassifications. The second

Scenario	Metric	Cost-Informed Weights		Scoring Manager		Both	
Baseline	Accuracy	**0.9993**	**-(0.04%)**	0.9989	(-0.08%)	0.9988	(-0.09%)
	F1-Score	**0.9814**	**-(1.06%)**	0.9741	(-1.80%)	0.9689	(-2.32%)
	Cost	$967	(-44.87%)	$887	(-49.43%)	**$840**	**(-52.09%)**
Amplified Impact	Accuracy	0.9799	(-1.98%)	0.5592	(-44.07%)	**0.9829**	**(-1.68%)**
	F1-Score	0.6455	(-34.92%)	0.3490	(-64.81%)	**0.7187**	**(-27.55%)**
	Cost	$46,240	(-73.58%)	$150,613	(-13.94%)	**$28,897**	**(-83.49%)**
Uneven Harms	Accuracy	0.9989	(-0.54%)	**0.9989**	**(-0.08%)**	0.9904	(-0.93)%
	F1-Score	0.8762	(-11.67%)	**0.9741**	**(-1.80%)**	0.7993	(-19.43)%
	Cost	$10,741	(-58.69%)	$12,787	(-50.82%)	**$9,365**	**(-63.98%)**
High Salary	Accuracy	**0.9989**	**(-0.07%)**	0.9989	(-0.08%)	0.9982	(-0.15%)
	F1-Score	**0.9776**	**(-1.44%)**	0.9741	(-1.80%)	0.9698	(-2.23%)
	Cost	**$13,273**	**(-48.97%)**	$13,667	(-47.46%)	$14,012	(-46.13%)

Table 8. Accuracy, F1-Score and cost change (and percent change) from baseline configuration for each IDS architecture configuration across all four scenarios in the electric utility environment. The best improvement in each metric is bolded.

configuration is cost-informed weights only, which takes the costs defined by the scenario, normalizes them into weights, and uses these weights during the model training process. This is the configuration used in the previous evaluation of cost-informed scenarios. The third configuration is scoring manager only, which uses a traditional supervised model using even weights as input into the scoring manager. The scoring manager then uses the scenario costs to determine the optimal filter threshold for each class based on Eq. 4. The accuracy based model will output likelihood scores for each packet class, and the largest difference between likelihood score and optimal filter threshold is selected as the predicted class. Specifically, the class which is furthest from the break-even cost is chosen as the predicted class. If no attack score exceeds its individual threshold then the packet is labeled as normal. The fourth and final configuration considers both a cost-informed weights model and the scoring manager. The cost-informed weights model produces a score for each packet. These scores are then sent to the scoring manager which uses the defined costs to find individual class-specific thresholds that result in the minimum overall cost within the training data. The combination of cost-informed weights and cost-based optimal filters are used to evaluate the testing data.

For evaluation purposes, the evenly weighted accuracy-based IDS configuration is treated as a baseline. The accuracy, cost, and F1-Score of the remaining three configurations are shown as raw values and percent change from the baseline configuration. The configurations are evaluated on each of the four scenarios in both environments. Results for the additive manufacturing environment are displayed in Table 7 and the electrical utility results are shown in Table 8.

The tables show that across all but the first scenario in the additive manufacturing environment, the inclusion of cost-informed weights leads to both significant average cost decreases and minimal average accuracy and F1-score

degradation. The second configuration, where only the scoring manager is used, provided lower average costs in five of the eight scenarios. All three increased average cost scenarios were in the additive manufacturing environment. We believe this is due to the distributions for each attack class in the additive manufacturing environment. The equation used to calculate optimal threshold values relies heavily on the probability of the class appearing in the data. Thus, the scoring manager performs best in highly imbalanced environments, which the additive manufacturing environment was not.

It was found that in both environments, incorporating both the cost-informed weights model as well as the scoring manager resulted in the minimum expected cost in nearly every scenario. Scenario 4 in the electric utility environment was the single outlier where cost-informed weights outperformed the other two configurations. However, in this scenario the combination configuration did not perform poorly, rather all three configurations performed similarly well, with all three average expected cost reductions being in the range of 46% - 49%.

6 Conclusion

This work demonstrates the potential benefits of including cost-sensitive learning in the development of intrusion detection systems within OT networks. By considering the financial impact of different classification errors, the detection system can make economically-informed decisions, trading minor accuracy reductions for significant cost savings. This work proposed a configurable IDS framework which incorporates two cost-sensitive techniques, including a cost-based weighting scheme which prioritizes classification errors based on their estimated cost, and a cost-based scoring manager which screens alerts based on an optimized filter. By integrating these techniques, the IDS shifts its focus from maximizing accuracy to minimizing the expected costs.

The costs used in this paper were based on prior estimates by [10]. However, these costs can be modified to accurately reflect any individual scenario. The ratio of misclassification costs drives the performance of our model, not the costs themselves. For an organization that is able to estimate their implementation specific costs, this framework offers an economically-informed alternative to accuracy based IDS techniques.

Results emphasize the importance of cost-aware decision making for cyber defense. The inclusion of cost-sensitive techniques into OT network security provides organizations with the ability to be proactive and make more informed security decisions in order to mitigate the potential operational and financial consequences associated with cyber attacks.

References

1. Ahmed, C.M., Palleti, V.R., Mathur, A.P.: Wadi: a water distribution testbed for research in the design of secure cyber physical systems. In: Proceedings of the 3rd international workshop on cyber-physical systems for smart water networks. pp. 25–28 (2017)

2. Anderson, R., Moore, T.: The economics of information security. Science **314**(5799), 610–613 (2006). https://doi.org/10.1126/science.1130992, https://tylermoore.utulsa.edu/science-econ.pdf
3. Böhme, R., Moore, T.: Modeling optimal filter configuration (2012), https://tylermoore.utulsa.edu/courses/econsec/f12/reading/lnse-fpfn.pdf
4. Freund, Y., Schapire, R.E.: A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences **55**(1), 119–139 (1997)
5. Gupta, N., Jindal, V., Bedi, P.: Cse-ids: Using cost-sensitive deep learning and ensemble algorithms to handle class imbalance in network-based intrusion detection systems. Computers & Security **112**, 102499 (2022)
6. He, S., Li, B., Peng, H., Xin, J., Zhang, E.: An effective cost-sensitive xgboost method for malicious urls detection in imbalanced dataset. IEEE Access **9**, 93089–93096 (2021). https://doi.org/10.1109/ACCESS.2021.3093094
7. Lee, W., Fan, W., Miller, M., Stolfo, S.J., Zadok, E.: Toward cost-sensitive modeling for intrusion detection and response. Journal of computer security **10**(1-2), 5–22 (2002)
8. Lemay, A., Fernandez, J.M.: Providing scada network data sets for intrusion detection research. In: CSET@ USENIX Security Symposium (2016)
9. Leverett, E.P.: Quantitatively assessing and visualising industrial system attack surfaces (2011)
10. Morin, A., Moore, T.: Towards cost-balanced intrusion detection in ot environments. In: 2022 IEEE Conference on Communications and Network Security (CNS). pp. 1–6. IEEE (2022)
11. Papa, S., Casper, W., Moore, T.: Securing wastewater facilities from accidental and intentional harm: a cost-benefit analysis. International Journal of Critical Infrastructure Protection **6**(2), 96–106 (2013), https://tylermoore.utulsa.edu/ijcip13.pdf
12. Shao, E.: Encoding IP address as a feature for network intrusion detection. Ph.D. thesis, Purdue University Graduate School (2019)
13. Sullivan, M., Schellenberg, J., Blundell, M.: Updated Value of Service Reliability Estimates for Electric Utility Customers in the United States. Tech. Rep. LBNL–6941E, 1172643 (Jan 2015). https://doi.org/10.2172/1172643, http://www.osti.gov/servlets/purl/1172643/
14. Thakkar, A., Lohiya, R.: Attack classification of imbalanced intrusion data for iot network using ensemble learning-based deep neural network. IEEE Internet of Things Journal (2023)