Prediction-Based Cyber Analytic Threat Detection

Kedrick Hill
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA 56267
hill1588@morris.umn.edu

ABSTRACT

Cyber attacks have increased over the years which has plagued businesses by exploiting their data. A leading development in network security uses fields that apply cyber analytic-based methods and techniques to overcome computing network attacks while ensuring businesses have secure systems. By using fields such as data science, machine learning, data mining and predictive analytics, detecting anomalies can become the next trend in efficient security based programs that minimize these threats with accuracy and efficiency. In this paper, we dive into the methods that are used in machine learning and predictive analytics to create systems that are decision based and are able to predict attacks.

Keywords

Cyber Analytics, Cybersecurity, Machine Learning, Predictive Analytics, and Data Science.

1. INTRODUCTION

In 2015, it was recorded by *CBS MoneyWatch* that 80 percent of businesses reported that they had been successfully hacked [1]. Personal safety and security can either be real world or virtual, so providing the necessary security to protect that information is vital. Every day, information is illegally obtained by virtual attacks across systems that are meant to be secure. The data can lead to false transactions, false identity, and other breaches of a person's privacy, safety, or security. The main defense for computing systems on a network is *cybersecurity*. For example, mobile phones are secured through their own security software created by their developer and our networks or computers are secured via software such as Norton. With the world constantly growing and advancing, it is important to have a secured system that will protect personal data.

The systems that protect us should have a stronger security algorithms that should be able to detect a breach and prevent it before the attack gets too far. One of those systems is a threat detection system which predicts the outcome of an attack based on its choices or its type. One that can detect a threat and extinguish it before information is taken. Data science methods, including machine learning, are leading the way in developing effective security software. Using historic data, machine learning can interpret decision making processes through predictive analytics.

Interconnected fields that are used in threat detection systems provide systems the ability to detect the misuse of data. The main purpose of this paper is to show the use of methods and techniques centered around machine learning algorithms through one or more of the methods discussed. However, before we discuss the usefulness of machine learning methods and techniques, we will talk about some background (section 2) info, consisting of cyber security, machine learning, and predictive analytics. Next, we will talk about data sets that are often used for studies in the field (section 3). Then we discuss methods and algorithms of machine learning and predictive analytics (section 4). Following that section, we discuss the results of studies mentioned and finally conclude (section 5).

2. BACKGROUND

Predicting the choices of a cyber-attack requires an extremely large quantity of data from a wide variety of past attacks. Many machine learning and predictive methods allow for these data sets to be used efficiently while building or using a model. The three models that will be discussed in section 4; Clustering, Support Vectors Machines, and Decision Trees are some methods work well with large data sets. However, before getting into using cyber analytic-based methods, we must understand their methodology and components.

2.1 Cyber Security

Cyber security is the process of ensuring information system protection which includes software, hardware, and any information or data [1]. Cyber security has grown in importance with the widespread development and usage of computing systems world wide. This is due to digitized data which has impacted the expansion of cyber crimes and attacks.

"Cyber attack[s] are commonly known as a computer network attack (CNA) which is the deliberate exploitation of computer systems, companies and networks dependent on technology" [1]. An effective CNA, through the use of malicious code, can disrupt data and jeopardize personal information. Preventing such occurrences is encompassed in the goals that Cyber security intends to accomplish and also limiting the current issues.

No business wants to have to deal with issues because of the toll it can have on them. Those issues that cyber security want to solve are:

UMM CSci 4901, October 2020 Morris, MN.

- Unprecedented Attacks
- Cyber Espionage
- Data Theft

Unprecedented attacks are new CNAs have not been recorded before and are difficult to prevent which leaves businesses worried about how they store and secure their data. Most of which have moved most of their data to the cloud since it has a much stronger firewall and higher security. Unfortunately, small and medium businesses are considered to be the most susceptible to attacks due to lack of assets to pay for a stronger infrastructure. They are often considered 'gateway' companies since they do not have strong security and hackers start with smaller businesses and work their way up to large businesses [1].

A solution to preventing such attack could be solved using threat detection systems with machine learning methods and algorithms. Reducing the time of an attack can ensure safe and secure networks and data. Later in the paper we will look more in depth on how machine learning and predictive analytic methods and algorithms can aid in the protection of data and the enhancement of cyber security as a whole.

2.2 Machine Learning (ML)

Machine Learning or Artificial Intelligence (AI) is the process of a system to learn through experience, or in this case, through data sets [5].

Data mining is a key resource that works cohesively with machine learning. Though it does not necessarily have a transparent definition, most express the general meaning as sifting through big data sets and locating trends [2]. It also serves as a junction between statistics and machine learning fields which are often used collectively.

Data often is distributed randomly into two sets, training and test datasets. Each has its own specific uses on how its data is applied. Training data is a random selection of data from the set that is used to predict outcomes generated from the training model. The model is created using the training data that is meant to be how we predict the test data [8]. The test data is what is used for final models and classification while also assessing the models performance using data not in the training set. It is used to test the predicted outcomes to ensure that the model is accurate. If the model fails or does not predict accurately then the training process is repeated with a whole new selection of training and test data until the model is classifying and predicting with high accuracy.

There are two types of learning associated with machine learning: Supervised and Unsupervised learning. Supervised Learning uses labeled data points to map a series of input and output pairings [5]. An example of labeled data in a threat detection system would be if a data point was considered as an "attack" or "not an attack". When applied to types of learning algorithms the system learns to recognize patterns associated with a particular label. Unsupervised Learning learn through the test data that is unlabeled or unclassified. Each learning type has a set of categories that certain machine learning models fit under. Clustering identifies clusters of data items that are similar to each other and is a form of unsupervised learning. Clustering utilizes both K-Means and K-Nearestneighbor equations to calculate what item belongs to which cluster (section 4.1). Classification assigns data items to predefined classes and is a form of supervised learning which incorporates Support Vector Machines (section 4.3) and Decision Trees (section 4.2).

2.3 Predictive Analytics (PA)

Predictive Analytics, which is precisely what one would expect from the name, predicts an outcome through data. Furthermore, "[p]redictive analytics is an area of statistics that deals with extracting information from data and using it to predict trends and behavior patterns" [6]. By utilizing the various predictive models that encompass data mining and machine learning, PA can calculate patterns or trends from past, current, or future data.

When applied to Cybersecurity, PA can analyze historical and current data and produce a more effective threat detection system [4]. Most use another form of analytics called prescriptive analytics which is more efficient at providing all possible outcomes of classification. Prescriptive analytics "analyzes all the possible outcomes to suggest the action plans during each particular case" [4].

3. THREAT DETECTION DATASETS

There are many data sets that were created with intention to help further study threat detection on systems. In this section we will look as some datasets that are commonly used for the methods discussed later in this paper (section 4). We will talk about the following sets: Knowledge Discovery in Databases (KDD), Netflow Data, and Defence Advanced Research Projects Agency (DARPA) 1999 data sets.

3.1 KDD Datasets

The Knowledge Discovery in Databases (KDD) 1999 data set is used for threat detection which contains Transmission Control Protocol (TCP) and Internet Protocol (IP) data collected by "MIT Lincoln Laboratory under the Defence Advanced Research Projects Agency (DARPA) and Air Force Research Laboratory (AFRL) sponsorship to evaluate computer network intrusion systems" [10]. KDD is one of the most widely used data sets which was also created for KDD cup challenge in 1999 [10]. The KDD cup challenge is an annual Data Mining and Knowledge Discovery competition that perform tasks on the KDD set which are chosen by the organizing party.

KDD includes a total of 41 variables that are split between 3 components: basic, content, and traffic features. We see what features are in each component of the set to help understand what can be seen and utilized. Some components share similar or the same variables as the others such as the two separate traffic features which are separated by connection types. Same-Host refers to both connections, or session, being the same for destination and current while Same-Service identifies if both users are using the same service as the current connection.

Each variable in each component is represented by the name of the variable, the data type, and a short description of the variable. There are some important terms from the feature names and descriptions that can better help understand the data. They are the variables: hot, su, root, port, and protocol, and the term checksum. A unique variable is the hot variable which is the number of misuses during a session. The description from figure 1 states that an indicator is true if it enters a system directory, creates a program or executes one. The prefix su means substitute user which allows another user to execute commands on the users operating system and can be seen used in binary representation under the su attempt variable. The root prefix attached to variables signifies the top-level or initial directory. On windows a top-level directory would be documents or downloads in the file system. The term port appears in some definitions and is known as the communication endpoint when directing a process on a system. A port that any internet user uses on the daily is port 80 which allows connection to network process World Wide Web. The variable protocol type is used in the components basic and traffic. A protocol is referred to as a set of rules or procedures when transmitting data over connections. Checksum is a term found under the wrong fragment variable which is also known as hashes, or a series of letters and numbers, and is solely used to check data for errors during transmission and storage of data.

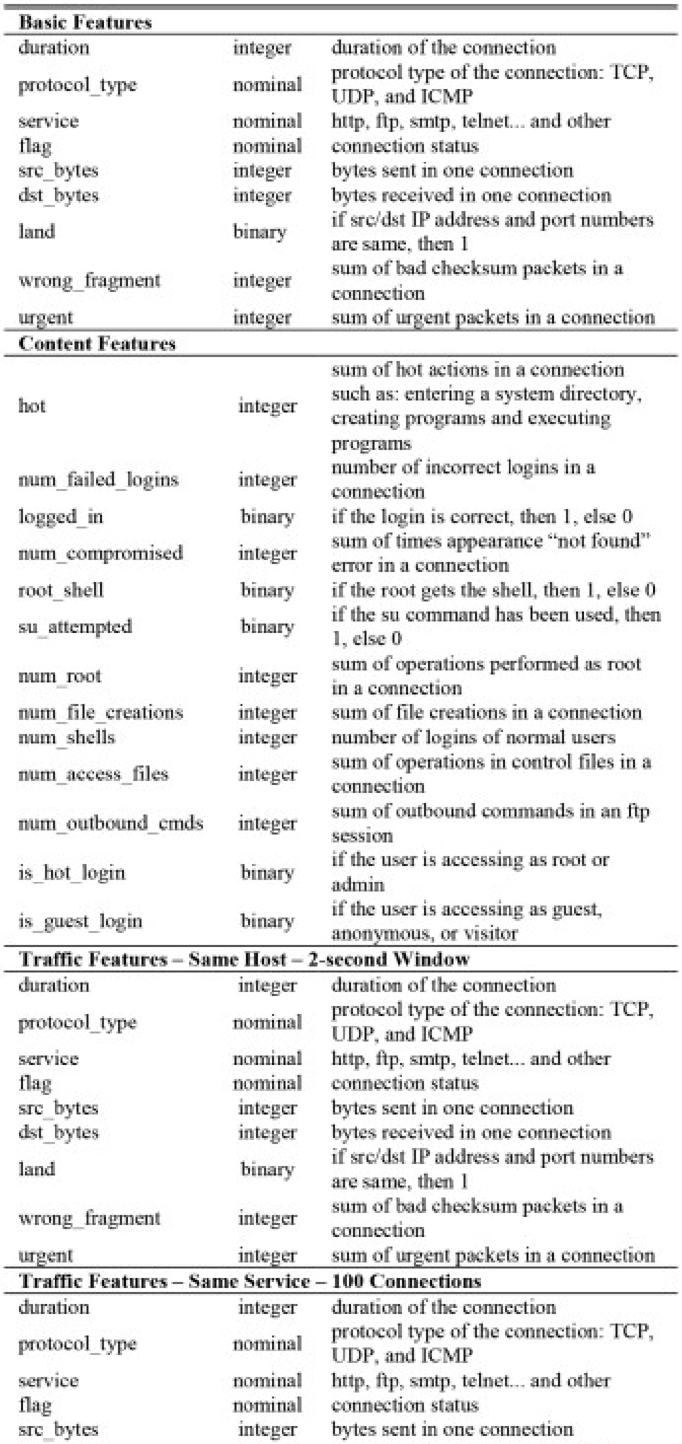
KDD has other versions of the KDD set such as the NSL-KDD and KDD Cup '99 sets. The NSL-KDD set contains the same records from KDD but they are revised and cleaned to solve some revolving issues with KDD '99 while the Cup '99 is the same as the base version.

3.2 NetFlow Dataset

NetFlow contains an assortment of network packet data that was obtained through a router or switch which collected input and output IP network traffic data. The data was collected from the real-world and simulated attack data using internet service providers who provided the attack data [2]. This makes a data set that can provide a real world aspect of how a threat could interact with a detection system. Just like the KDD set, the NetFlow set also has three components: "NetFlow Exporter, a NetFlow Collector, and an Analysis Console" [2]. Unlike KDD, NetFlow has ten versions of the data where version 1 to 8 are similar with minor changes, and the remaining version have more significant changes [2]. The data is also preprocessed so it has went through cleaning and has been has been reformatted to be understandable for users since real-world data is often incomplete. Figure 2 shows versions 1 through 8's two separate feature sets labelled as: Simple Network Management Protocol (SNMP) and Flow Statistics [2]; and contain mostly the same features with some more personalized features.

3.3 DARPA Dataset

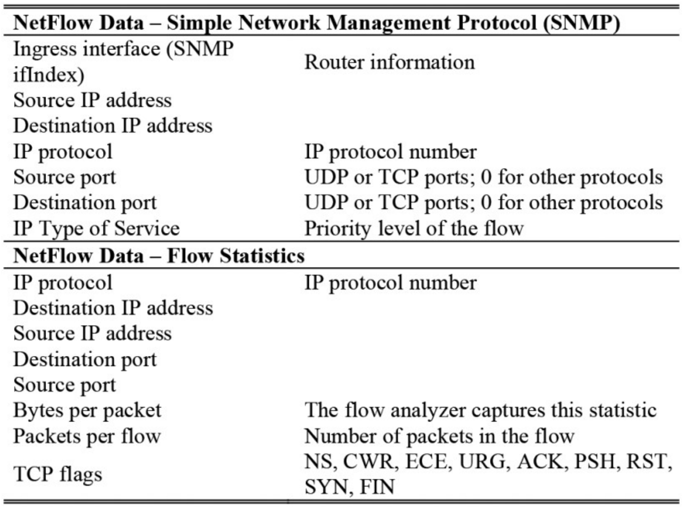
Two common DARPA sets are the one of '98 and '99 and are continuing to be some of the most broadly used data set in many publications. DARPA 98' is exactly the same data as KDD set while the DARPA '99 set was collected for 5 weeks and has a larger number of attack types. DARPA '99 is split into training and test sets with a 3 to 2 ratio respectively for the use of experimentation [2]. Though Sharing many attributes of KDD data, the DARPA sets also contain the TCP dump files and logs which is a program that is ran in the command line to view transmitted and received files over a network. The data set contains four types of attacks that are common. Denial of Service (DoS) which is an attempt to deny users from network or computing resources [2]. User to Root (U2R) where an attacker is granted root, initial or top-level directory of a file system, access to the user. Remote to Local (R2L) grants access for the attacker but over the local network. Finally, Probe to Scan is aimed



Basic Features		
duration	integer	duration of the connection
protocol_type	nominal	protocol type of the connection: TCP, UDP, and ICMP
service	nominal	http, ftp, smtp, telnet... and other
flag	nominal	connection status
src_bytes	integer	bytes sent in one connection
dst_bytes	integer	bytes received in one connection
land	binary	if src/dst IP address and port numbers are same, then 1
wrong_fragment	integer	sum of bad checksum packets in a connection
urgent	integer	sum of urgent packets in a connection
Content Features		
hot	integer	sum of hot actions in a connection such as: entering a system directory, creating programs and executing programs
num_failed_logins	integer	number of incorrect logins in a connection
logged_in	binary	if the login is correct, then 1, else 0
num_compromised	integer	sum of times appearance "not found" error in a connection
root_shell	binary	if the root gets the shell, then 1, else 0
su_attempted	binary	if the su command has been used, then 1, else 0
num_root	integer	sum of operations performed as root in a connection
num_file_creations	integer	sum of file creations in a connection
num_shells	integer	number of logins of normal users
num_access_files	integer	sum of operations in control files in a connection
num_outbound_cmds	integer	sum of outbound commands in an ftp session
is_hot_login	binary	if the user is accessing as root or admin
is_guest_login	binary	if the user is accessing as guest, anonymous, or visitor
Traffic Features – Same Host – 2-second Window		
duration	integer	duration of the connection
protocol_type	nominal	protocol type of the connection: TCP, UDP, and ICMP
service	nominal	http, ftp, smtp, telnet... and other
flag	nominal	connection status
src_bytes	integer	bytes sent in one connection
dst_bytes	integer	bytes received in one connection
land	binary	if src/dst IP address and port numbers are same, then 1
wrong_fragment	integer	sum of bad checksum packets in a connection
urgent	integer	sum of urgent packets in a connection
Traffic Features – Same Service – 100 Connections		
duration	integer	duration of the connection
protocol_type	nominal	protocol type of the connection: TCP, UDP, and ICMP
service	nominal	http, ftp, smtp, telnet... and other
flag	nominal	connection status
src_bytes	integer	bytes sent in one connection
dst_bytes	integer	bytes received in one connection
land	binary	if src/dst IP address and port numbers are same, then 1
wrong_fragment	integer	sum of bad checksum packets in a connection
urgent	integer	sum of urgent packets in a connection
urgent	integer	sum of urgent packets in a connection

Figure 1: Features of the KDD sets components: Basic, Context, and Traffic features for TCP connections [10].

to collect information on network resources [2].



NetFlow Data – Simple Network Management Protocol (SNMP)	
Ingress interface (SNMP ifIndex)	Router information
Source IP address	
Destination IP address	
IP protocol	IP protocol number
Source port	UDP or TCP ports; 0 for other protocols
Destination port	UDP or TCP ports; 0 for other protocols
IP Type of Service	Priority level of the flow
NetFlow Data – Flow Statistics	
IP protocol	IP protocol number
Destination IP address	
Source IP address	
Destination port	
Source port	
Bytes per packet	The flow analyzer captures this statistic
Packets per flow	Number of packets in the flow
TCP flags	NS, CWR, ECE, URG, ACK, PSH, RST, SYN, FIN

Figure 2: Shows the minimum set of NetFlow data variables for sequence packets[2].

4. ML AND PA METHODS

Machine learning and predictive analytics use various methods when predicting outcomes. Some that have been mentioned already are neural networks which use supervised and occasionally unsupervised learning. However, there are a few in particular that are more commonly used in decision-making and prediction based analysis. The methods that are utilized across the fields mentioned in section 2 are:

- Clustering
- Support Vector Machines (SVM)
- Decision Trees

4.1 Clustering

Clustering is a pattern recognition method that takes high-dimensional unlabeled data by grouping data in similar classification groups. Buczak and Guven state in their article that, “The main advantage of clustering for intrusion detection is that it can learn from audit data without requiring the system administrator to provide explicit descriptions of various attack classes” [2]. In threat detection systems it is important when taking input to detect abnormalities in the data. One particular process is through K-means which processes clustered data quickly [3].

K-means (kM) is a centroid-based equation where each cluster is characterized by its mean vector, μ_j, which is the mean of all vectors, or data points where the coordinates are its features, in set J [2]. Centroid refers to the center point of the cluster group by calculating μ_j. Before we can calculate kM we must first acquire a data set that contains a list of vectors. Next, separating the points either randomly or in order equally based on the k spatial, or total, clusters is performed. This is then looked at and altered so that there is not outliers in the clusters. If so, then they are reassigned to a closer cluster group. For example, if we are looking for three clusters and have eighteen vectors, then we would equally distribute the vectors where each cluster has six vectors. To calculate kM, we want to calculate the euclidean distance (magnitude between two vectors), denoted by $\|...\|^2$, of all values in a cluster where observations (x_1, x_2, ..., x_i) are subtracted by the mean of all vectors in cluster j. This will be done until all of the clusters k have been calculated resulting in k total groups or clusters. Once the results are calculated, they are redistributed to new clusters based on their distance to nearest centroid. We repeat the process and equation until vectors are no longer reassigned to new clusters. The equation for kM is shown below in equation 1 and is intended to keep the vectors near clusters that show the most sufficiently low distance from a cluster. By repeating the process, we ensure that each vectors distance in the cluster is checked across various clusters until we have the most efficient cluster groupings.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \|x_i - \mu_j\|^2 \tag{1}$$

In threat detection systems, this provides a way to classify attacks based on their specifications. For example, the amount of attempts can aid in classifying if a vector is an attack or not, or even what type of user they are (regular or intruder). A study by Blowers and Williams used clustering on network packets data collected in the KDD data set [2]. They used a form of clustering called Density-based spatial clustering applications with noise or DBSCAN. DBSCANs work similar to kM clustering with some differences, by grouping vectors that are close together creating several spatial clusters. In this study they separated the packets into either regular or anomalous network packets. As a result, the anomaly threat detectors performance was a high 98 percent, which is calculated by adding up all the correctly classified values and dividing it by the total amount of packets [2].

Another study from Sequiera and Zaki did a study on shell commands. Their data set contained 500 session from nine users at Purdue University that had long streams of commands [2]. The goal was to detect whether the user was a regular or an intruder in the system. Its stated that each “session was parsed into tokens and subsequently represented as a sequence of tokens” [2] which they then ran several approaches on. The approaches looked at the similarities of a sequence and checking if any matched other sequences. This particular study performed with a 80 percent accuracy rating which is lower than the previous; However, this can be because of the small amount of data that they were dealing with. They also had a false acceptance rate (FAR), likely-hood of accepting a unauthorized user, percentage of 15 which they compared to other studies that showed higher rates than they had. From this point of view, this is a improvement on previous studies with the same data set.

4.2 Decision Trees

Decision Trees are tree like structures for nodes that contain classifications of decision-based processes. Each node is an attribute that is tested while the branches are the potential outcomes derived from that attribute. In figure 3 you can see what a typical decision tree model looks like. Commonly used decision tree models are ID3, C4.5 and CART

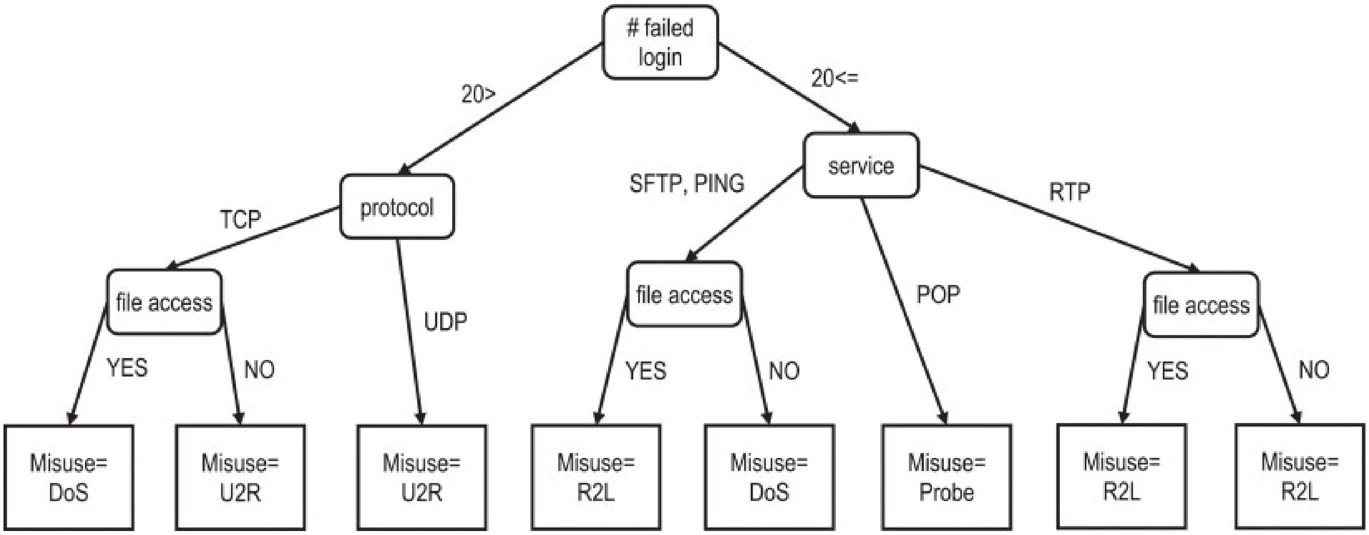


Figure 3: An example of a Decision Tree [2].

which are diverse in the way that they are represented [9].

At the top of the tree is a basic attribute with nodes to the left and the right which are traversed until they reach a node, or leaf, that classifies that there is an attack. In this specific tree, it is checking on how many attempts were failed. Based on the total fails it moves down the corresponding branch to the next attribute node. Eventually the tree will reach a attribute node called files accessed. From there, the tree is only one test away from disclosing the attack type (outcome).

Using data mining techniques of gathering training and testing data is very useful for a Decision Tree when identifying the classification nodes and determining what type of outputs that are possible.

Applying Decision Trees in threat detection systems can benefit the systems by quickly traversing a tree that can classify if a user is an attacker or not or what type of attack the user is attempting. We can see this applied in a study from Relan and Patil that used two types of KDD data sets: Cup 99' KDD and NSL-KDD sets, both mentioned in section 3.1. They used two different decision trees for this study. One decision tree was used with pruning while the other did not use pruning. Pruning reduces the size of decision trees by removing are redundant or unneeded classifications. An example of pruning would be the removal of a input node that asked if a su command was used then proceeding to the next input asking if it was not used. Since this could be done in one question, pruning will remove that second input node. Their results showed that, with pruning, the decision tree produced a accuracy of 98.45 percent and a FAR of 1.55 percent[9]. This is due to the removal of extraneous parameters that would cause the process to take longer from extra inputs.

Another study that shows the usage of decision trees in threat detectors was done by Kruegal and Toth which used an open source tool called Snort to aid in their detection on TCP network conversation through which application programs communicate, dump files [2]. They used the DARPA 99' data set that contains TCP dump files which we discussed in section 3.3. Snort as mentioned is an open sourced tool uses a signature based approach for threat detection systems [2]. Kruegal and Toth also incorporated the use of clustering to aid in developing their decision tree which is a useful process referred to as hybrid detection systems. They use multiple methods and algorithms to assist one another in detecting threats. In this study they derived the decision tree using the clustering methods classifications ability. This minimized the total outputs from an input node decreasing the total comparisons that were needed. While testing the speed of this hybrid system, they produced results of 5 to 105 percent increase of speed with a 40.3 percent average increase [2]. This shows that the use of multiple methods can benefit threat detection systems by reducing the processing time of the decision tree up to double the speed that decision tree by itself would.

4.3 Support Vector Machines (SVM)

Support Vector Machines are accurate, robust, and reliable machine learning algorithms that are well known for their generalization ability and are particularly useful when the number of features are high and the number of data points is low [2]. The quickness of an SVM is crucially important for cyber security. This builds on the integrity of the system reducing a CNA's ability to exploit data in the network. A SVM, in a hybrid algorithm, may benefit from a clustering addition since it does already handles much of the operations that were discussed in section 4.2.

When it comes to threat detection the quickness of the SVM makes it a common pick when classifying vectors in a similar way that clustering does. One study that used an SVM done by Wagner et al. looked into NetFlow data which we went over in section 3.2. The SVM that was used is called a "one-class SVM classifier, which is considered a natural approach for anomaly detection" [2]. Multiple tests were ran in the study with results ranging from 89 to 94 percent accuracy and false positive rates of zero to three percent, which is calculated using the quantity of false positives classified divided by the total quantity of values.

An additional study using SVMs was done by Perez and Farid which used the NSL-KDD data set from section 3.1.

They expanded the SVM tests by fluctuating the total input features and also checking the performance of both the training and test sets in the data set. They performed a series of tests on the number of input features of values 3, 36, and 41; which resulted in a classification accuracy percentage of 91 99, and 99 respectively [9]. They used This showed that as the number of input features get higher, so does the accuracy of the algorithm. Lastly, the results from the performance of both sets were scored using what is called as the F1-score or F-score. F-Score is calculated using binary classification using the equation $tp/(tp + 1/2(fp + fn))$; where tp is true positives, fp is false positives, and fn is false negatives [7]. The score for the training data was a 0.99 and they found that the test data scored lower, which is uncommon, at a mere 0.77 [9]. Perez and Farid, summarized their findings by stating that the test data was low due to "poor generalization, [since] it cannot effectively detect unknown network intrusions" [9].

5. RESULTS

The results from various studies using the methods of machine learning and predictive analytics show their effectiveness when predicting attacks to a system and classifying them. The purpose of this paper was to show how the use of machine learning and predictive analytics in cyber security can show that it is an effective application for threat detection systems. The use of clustering grants a classification to the data or groups of data based on their specifications as shown through the studies. Bowers and Williams proved that using a form of clustering called DBSCAN was effective when clustering groups in small area.

Sequeira and Zaki showed that clustering can even be used effectively on shell command data. Decision Trees also see similar results in their studies. With the variety of areas and accuracy, clustering can be an effective tool for threat detection. Both studies used similar data sets, one just had extra attack simulations than the other, but still revealed high accuracy.

Kruegal and Toth identified that utilizing multiple methods in one detection system increases processing speeds between 5 to 105 percent. In Cyber security speed of detection plays a key role in preventing attacks. Lastly, Support Vector Machines, when studied, revealed that they can be high in accuracy with high presence of features and be able to predict anomalies in systems.

6. CONCLUSION

The incorporation of cyber-based analytic based methods have been a rising tool in cybersecurity due to its reliability in threat detection. In the paper, we discussed cybersecurity's goals for network security and issues that are still leaving businesses vulnerable. The use of machine learning provides a new method of threat detection based on the accuracy of the predictions. The three models provide pattern recognition, generalization of data, and decision processing when predicting what type of attack is occurring. Whether the methods are used solo or in conjunction, they provide effective ways to detect and prevent the further attacks and is still a new frontier in network security for systems.

Predicting anomalies in a system is vital to ensuring the integrity of a system. Increasing the speed of the detection through the use of machine learning and predictive analytic methods provides a system with stronger security, which is a underlying goal of cyber security. The use of multiple methods, hybrid methods, can increase accuracy of a threat detection system.

7. REFERENCES

[1] R. Adlakha, S. Sharma, A. Rawat, and K. Sharma. Cyber security goal's, issue's, categorization data breaches. In *2019 International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, pages 397–402, 2019.

[2] A. L. Buczak and E. Guven. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys Tutorials*, 18(2):1153–1176, 2016.

[3] H. M. Farooq and N. M. Otaibi. Optimal machine learning algorithms for cyber threat detection. In *2018 UKSim-AMSS 20th International Conference on Computer Modelling and Simulation (UKSim)*, pages 32–37, 2018.

[4] K. Li, B. Martino, L. Yang, and Q. Zhang. *SMART DATA: state-of-the-art perspectives in computing and applications.* CRC PRESS, 2020.

[5] Wikipedia. Machine learning — Wikipedia, the free encyclopedia. `https://tinyurl.com/yxwsf7dp`, 2020. [Online; accessed 21-October-2020].

[6] Wikipedia. Predictive analytics — Wikipedia, the free encyclopedia. `https://tinyurl.com/y6sa9ktz`, 2020. [Online; accessed 21-October-2020].

[7] Wikipedia contributors. F-score — Wikipedia, the free encyclopedia, 2020. [Online; accessed 30-November-2020].

[8] Wikipedia contributors. Training, validation, and test sets — Wikipedia, the free encyclopedia, 2020. [Online; accessed 23-October-2020].

[9] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang. Machine learning and deep learning methods for cybersecurity. *IEEE Access*, 6:35365–35381, 2018.

[10] O. Yavanoglu and M. Aydos. A review on cyber security datasets for machine learning algorithms. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 2186–2193, 2017.