IJCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-7, Issue-10, Oct 2019 E-ISSN: 2347-2693

Machine learning in the prediction, determination and further study of different cyber-attacks

Sagar Bansal[1*], Anshika Singh[2]

[1,2]Dept. of Computer Science, Bhaskaracharya College of Applied Sciences, Dwarka, University of Delhi, Delhi, India

Corresponding Author: sagarbansal719@gmail.com

DOI: https://doi.org/10.26438/ijcse/v7i10.2736 | Available online at: www.ijcseonline.org

Accepted: 11/Oct/2019, Published: 31/Oct/2019

***Abstract*—** Cyber Security introduces a group of methods, used to shield networks, data and programs from intrusion, deterioration and illegal access. Cyber intrusion is the act of breaking the security of one's computer with the means of a network. To cut down the threat of various illegal accessing in order to enhance the cyber security, Machine Learning approach is used widely. Machine learning in itself is the study of various ways to train the machine with real datasets and make them act like humans in similar circumstances. In this paper, most of the Machine Learning and Deep Learning algorithms that are used for enhancing cyber security have been summed up.

***Keywords*—** Machine Learning, Deep Learning, Cyber security, Intrusion

I. INTRODUCTION

The motive of Internet is to connect every person on this planet with Internet, so that knowledge can be shared among people at minimal cost, everyone can get their own platform and similar opportunities regardless of gender, skin-color, social background and geographic location, and anyone can share their own opinions about anything. But with this initiative the likelihood of being socially hacked comes handy. Now-a-days, cyber-attacks are one of the major concerns growing among people due to trivial fact that their privacy is at stake. Cyber-attack is a malicious practice due to which the computer system and network can lose its credibility. These cyber-attacks are intended to harm the person in any possible way. The hacker may want to steal user id/ password, or any kind of data (Project details, high level information, research paper), or stop the system from working, just to get financial/other benefits by leveraging them. Different researches as well as our own observations tell us that we (humans) are highly emotional about our privacy, and at any cost we are not ready to let anyone infringe in our personal space. Privacy is our born right, even if not a part of fundamental rights, but nonetheless less than that. Growth in the count of people connecting to Internet of Things (via various personal and social platforms) has also resulted in high number of cyber-attacks.

There have been many researches but none is fully efficient to control these daily-basis changing cyber-attacks. A method alone can't detect or prevent such type of attacks because as techniques are developed to prevent them so as the attacks. Many of the techniques are also easily available with little searching on search engines. The reason that large number of people is affected by these cyber attacks is just because there isn't any single trivial pattern among all the various types of cyber-attacks. Most of the techniques involved in intrusion detection and prevention use Machine Learning or Deep Learning in its core. We can achieve human equivalent performance by using ML and DL. They are easy to implement and give expected results with high accuracy. Also, same algorithms can be used in different ways for various problems and can be modified if any changes occur in future environment. We have to precisely categorize these different trends and then develop solutions according to the specificity of each type of attack. Here in this paper, we provide a summarization of most of the methods, which are highly efficient for future uses including detection, and prevention of such attacks.

Machine Learning vs Deep Learning

All of the presented methods or techniques either use Machine Learning or Deep Learning in its core. We have distinguished these two so as to make the differences clear beforehand.

The concept of machine learning comes from the idea when we think of machine as a substitute of humans. We all are working in this sector to replicate humans into machines exactly. The way a human kid can kick a football, the same way a machine should be able to kick a football. The art of interpreting from the current environment and reply back immediately is what needed in machines to be exact humans.

Machine learning in itself is the study of various ways to train the machine with real datasets and make them act like humans in similar circumstances. There are various ways to train a machine or to make it interpret the actions to be done. All of that is part of Machine learning. In machine learning, there are various methods designed for different tasks such as Linear Regression, Support Vector Machines, Decision Trees, Random Forest and many more. Example can be deciding whether to play football tomorrow or not based on factors like temperature, outlook etc. The machine will be trained with previous datasets in order to predict actions depending on different factor values.

Deep learning is a subclass of Machine learning. All the deep learning methods and concepts comes under machine learning but vice-versa is not true. Deep learning aims to automate predictive analytics by learning to express each concept in equivalence to simpler concept. Deep learning is generally used to gain high accuracy with flexibility. Deep learning recognize images and speeches as well as for processing visual arts and natural language. Some of the methods used in Deep Learning are: RNN, Bayesian Triphone GMM-HMM, Hidden Trajectory (Generative) Model, etc. Some of the areas of application are: automated driving, aerospace and defense, medical research, industrial automation and electronics.

II. RELATED WORK

Mohammad Esmalifalak et al. (2013) had proposed supervised machine learning based methods for stealthy attack recognition. In first they initially trained a distributed Support Vector Machine by taking the supervised learning approach over labeled data. Providing provable optimality & convergence rate, the design of this Support Vector Machine is grounded on the alternating direction technique of multipliers. Whereas, the second technique doesn’t need data for training and notices the deviation in values. Principal component analysis was used in the first and second method which lessens the dimensionality of the data to be computed, leading to lesser processing complexities.

Michal Chora et al. (2015) proposed a model focusing on machine learning based technique which models the regular behavior of an application and senses cyberattacks for the purpose of abating evolving application layer cyberattacks which are the major threats and key challenge for network and cybersecurity. The model contains patterns that are determined by the use of graph-based segmentation method and also dynamic programming. The proposed model is based on data gathered from HTTP requests created by client to a web server. The outcomes support the efficiency of the suggested algorithm that can efficiently work for application layer attack recognition. In the experiments done on CSIC’10, the results prove that the suggested method can attain 94.46% of detection ratio with less than 4.5% of false positives.

Chaitrali Amrutkar et al. (2016) has proposed kAYO method to differentiate malicious mobile webpages from benevolent mobile webpages. This method makes the prediction based on constant attributes of a webpage starting from the numeral iframes to the presence of known fake mobile phone numbers. Binomial classification technique is used to develop a model for kAYO to deliver 90% correctness and 89% true positive rate. kAYO’s results indicate that its efficiency is either equivalent or exceeds the present static methods used in the desktop space. kAYO also senses numerous malicious mobile webpages not exactly sensed by present methods such as VirusTotal and Google Safe Browsing. kAYO’s constraint is that it is incapable to gather webpages that use JavaScript to sense and then forward to the mobile webpage.

Joseph Siryani et al. (2017) proposed a framework offering useable decision suggestions about whether an operator is needed for solving ESM issue of customer and also make better cost-predictions for operations of smart meter field by leveraging advanced analytics of ESM network communication-quality data. A framework for a decision-support system was supported that operated inside the IoT ecosystem. The model was pragmatically assessed with data sets of a commercial network. The efficiency of this framework with a Bayesian Network prediction model was compared with these three ML prediction model classifiers: Naïve Bayes, Random Forest and Decision Tree. Experiments were done considering the effectiveness and efficiency of the suggested technique. With the correctness of 96.69% (the maximum operational savings), the prediction model – Random Forest proved to be highly efficient for the estimation of smart meter data for big sets. After that, Naïve Bayes, Decision Tree and Bayesian Network attained the second, third and fourth position, respectively.

Nir Nissim et al. (2017) had proposed a framework named ALDOCX which aims to correctly recognize the novel spiteful docx files and in addition it efficiently improves the framework’s recognition abilities with time. Recognition depends on a Structural Feature Extraction Methodology, which is executed statically with meta-features gathered from docx files. Using ML algorithms with Structural Feature Extraction Methodology, a recognition model is thus formed that effectively senses novel unknown spiteful docx files. Since it is important to preserve the recognition model’s updatability and integrate novel spiteful files generated daily, ALDOCX incorporates our active learning approaches, which are built to efficiently guide anti-virus merchants by better directing their specialists’ analytical efforts and improve recognition capability. ALDOCX recognizes and then attains new docx files that are most possibly malicious, as well as informative benevolent files. These files are used for improving the knowledge stores of recognition model as well as anti-virus software.

Kensuke Fukuda et al. (2017) had proposed domain name system (DNS) backscatter as a novel resource about networkwide activity. Automatically, when the targets/middleboxes look for the domain name of the creator, a reverse DNS queries is generated, named as Backscatter. The commanding DNS servers that manage reverse DNS can access the queries. Despite the portion of backscatter they see significantly depends on the server's location in the DNS hierarchy, the action that touches several targets can be seen even in sampled observations. This data regarding the queriers is used to categorize originator activity via the use of machine-learning. This technique was used to study nine months of action from one authority to recognize trends in scanning, recognizing bursts complementary to Heartbleed, and wide & uninterrupted scanning of secure shell.

Song Deng et al. (2017) had proposed a hybrid gene expression programming (ID-HGEP) based intrusion detection algorithm. It is combined with gene expression programming & ARND-RS. The ID-HGEP algorithm can find a function model between network data flow and attack types. The function model can distinguish net flow from normal or abnormal data streams. If the net flow is an abnormal data stream, ID-HGEP can judge the attack type. The authors have also proposed a distributed intrusion DID-HGEPCloud computing platform to better handle massive and high-dimensional net flow. In the DID-HGEPCloud algorithm, firstly, network log is divided into a set of key/value pairs. Where value includes all parameters of ID-HGEP algorithm. Then, these key/value pairs are implemented in parallel by multiple Map tasks. Then, by sorting and inputting the results into reduce for parallel computing, a variety of local intrusion detection models is generated and finally merged into a global intrusion detection model.

George Loukas et al. (2017) strongly suggested divesting the continuous task of invasion recognition using deep learning. This approach was not limited to a single type of attack. The developed testbed for the experimental evaluation was a 40 cm stretched remote-controlled 4x4 robotic vehicle and a dual-core based on-board computer. DoS attack, Command injection attack and Malware attack were used for training the model. To reduce the processing time for reaching a detection decision, the concept of offloading was formulated, where a higher-end computing infrastructure carries out the processing. This results in network delays and potential for network failures, which place extra volatility on the overall detection latency.

Cho Cho San et al. (2017) categorized 11 different malware families by finding their important API features using the results of enhanced and scalable version of cuckoo sandbox. The suggested scheme provided feature extraction algorithm, reduction and representation process to recognize and denote the extracted feature characteristics. Decision Table, Random Forest and K-Nearest Neighbor ML multi-class classifiers were used and tested to classify various types of malicious software. Performance was evaluated on the basis of Accuracy, True Positive, False Positive, ROC Area.

James B. Fraley et al. (2017) explained how machine learning algorithms can be used to sense and highlight advanced malware and also it resulted in saving time for cyber defense analysts. The approach followed six parts: develop business understanding, analyze data and data dependencies, engage subject matter experts, prepare dataset, develop model and evaluate model. There were multiple runs in order to consume the dataset into a single model. TensorFlow was used to create and experiment with the model. TensorFlow can efficiently utilize hundreds of servers to quickly train and develop advanced NN and DNN models. TensorFlow had a unique ability to deliver and manage both computational and state management through parallelization. Classifier performance was measured based on common stratified k-fold cross-validation. Stratified cross-validation examined class distribution and how those classes are distributed consistently across each fold. Model validation also evaluated instances of the dataset that were correctly labeled.

Fanyu Bu (2018) proposed a high-order clustering procedure based on dropout DL for diverse statistics in cyber-physical-social schemes. For a heterogeneous data, like a video (consist of set of images, audio and some text), the task of this high-order k-means algorithm is to cluster the data into k subsets X = X1 U X2 U X3 …U Xk under the condition X1 ∩ X2 ∩ X3 … ∩ Xk =∅ depending on the similarity between each two objects. The architecture of the algorithm is separated into two parts: in first, three dropouts stacked autoencoders (each with three hidden layers) learn the features for the modalities of each object. Then we can get feature vectors from them. In next part, the tensor k-means algorithm measures the similarity between each object and every clustering center to cluster the heterogeneous objects represented by the feature tensors.

Saeed Ahmed et al. (2018) advocated a plan to sense a covert cyber deception assault in the state estimation – measurement feature data that were gather using a smart – grid communication network based on supervised machine learning. Feature selection was used to improve the categorization accuracy and abate the computational complexity and associated time-delay at PCC. Feature selection is a special technique for dimensionality reduction in which a part of the initial set of features is selected without any transformation to a low-dimensional space, i.e., the features in the original set that represent measurements of physical quantities retain their units. Only selected are those features of the SE-MF dataset that are discriminative and that can be used to accurately differentiate between compromised and uncompromised data. To detect CCD assaults, they employed a GA-based FS technique to select the most discriminative features, and then employed an SVM-based ML algorithm on the selected features.

Sheraz Naseer (2018) presented some anomaly detection models based on distinct deep neural network structures, comprising repeated neural networks, autoencoders and complex neural networks. NSLKDD training dataset was used to train these models and both NSLKDDTest+ and NSLKDDTest21 dataset were used to evaluate them. The model implemented different Autoencoders (Sparse, Denoising, Contractive, and Convolutional), LSTM and Convolutional Neural Networks (CNN). To compare they used Scikit - learn implementations of Binary classification algorithms to train conventional models. These models were trained on unraveled version of Training Datasets. Classification algorithms used to train conventional models include Extreme Learning Machine with Generalized hidden layer proposed by, RBF SVM, Decision Tree (J48) with 10 node depth, Naive Bayes, Random-Forest with Quadratic Discriminant Analysis, 10 J48 estimators, and Multilevel perceptron (MLP).

Naseer R. Sabar et al. (2018) took two antithetical objectives – accuracy and model complexity to formulat a SVM configuration process as a bi-objective optimization problem. They proposed a unique hyper-heuristic framework that is unconstrained to the problem domain for bi-objective optimization. Proposed framework had a high-level strategy and low-level heuristics. High-level strategy had generated a new SVM configuration using a low-level heuristic selected on the basis of search performance. Low-level heuristics each used unlike rules to effectively discover the SVM configuration search space. In each iteration, the high-level strategy selects a heuristic from the existing pool of low-level heuristics, applies it to the existing key to produce a new key and decides whether to take the new solution. The low-level heuristics constitute a set of problem-specific heuristics that operate directly on the solution space of a given problem.

Thiago Alves et al. (2018) talked about a protocol-independent design approach applying an open source PLC. The PLC was modified to protect all information it sends over the network. With it, to shield against network flood attacks, an IPS based on ML was included in the PLC network stack resulting in a secure mechanism. The intrusion prevention system interfaces all packets received from the external network. To detect DoS attacks and network anomaly, it boasts off an enactment of an entrenched unsupervised clustering algorithm distinguishing the incoming streaming data real-time. If an attack is detected the IPS generates its own custom rules to limit the attacker's IP from the network. The unsupervised clustering algorithm (Kmeans) makes the IPS generic and adaptable to any varying network state that a regular network may experience.
Lorenzo Fernández Maimó et al. (2018) suggested a different 5G-oriented cyber defense architecture which identified cyberthreats in 5G mobile networks with good efficiency. The architecture used deep learning methods to examine network traffic by taking features from network flows. This model allows adjusting the configuration in order to control traffic fluctuation, targeting both to improve the computing resources needed in each particular moment and to fine tune the behavior and the performance of analysis and detection processes. The anomaly detection is arranged in two levels: at the low level, the flow collector gathers all the different flows during a specified time interval and computes a vector of attributes that the ASD module will categorize as normal or abnormal. A symptom packet containing the time stamp, type of anomaly spotted and feature vector involved, is directed to the next level i.e., NAD module., if an anomaly is suspected. The NAD module will refine the final detection results.

Mahmood Yousefi-azar et al. (2018) have suggested a new system to identify malware which is called Malytics. This scheme is independent of any specific tool/operating system. It finds static characteristics of every given binary file to differentiate malicious and benevolent. This scheme comprises of three phases: feature extraction, similarity measurement & classification. These three stages are executed using neural network having two hidden layers & one output layer. Feature extraction, is completed by tf - simhashing, is corresponding to the first layer of a specific neural network. Using both Windows and Android platforms, the author has evaluated its performance. Malytics outperforms a varied range of learning-based methods and distinct state-of-the-art models on both of the platforms.

Pratik Chattopadhyay et al. (2018) proposed a technique for insider threat detection from time-series classification of user activities. At start, a group of single-day attributes is calculated using the user activity logs. A time-series attribute vector is next built from the statistics of every individual single-day attribute over a time interval. The label of every individual time-series attribute vector (be it malicious/non-malicious) is gathered from the ground truth. To categorize the imbalanced ground-truth insider threat information comprising of only a paucity of malicious occurrences, the cost-sensitive information adjustment method was used that under trials the non-malicious class occurrences arbitrarily. A two-layered deep auto encoder neural network was employed as a classifier to compare its performance with other popularly used classifiers: random forest and multilayer perceptron. Encouraging results were found by assessing our technique with the CMU Insider Threat Data, which is the only openly accessible insider threat data set consisting of about 14-GB web-browsing logs, along with logon, device connection, file transfer, and e-mail log files.

Mohamad Nazrin Napiah et al. (2018) proposed an Intrusion Detection System (IDS) also known as ‘Compression Header Analyzer Intrusion Detection System’ (CHA-IDS). This system analyzed 6LoWPAN compression header data to alleviate the discrete and combine routing attacks. For the purpose of data analysis, collection, and system actions; capturing and managing unprocessed data was done by this multi-agent system framework - CHA-IDS. To find only noteworthy features required for the invasion recognition, the projected CHA-IDS incorporated best-first & greedy stepwise with correlation-based feature selection. These characteristics were then verified by six ML algorithms to identify the best categorization technique that able to differentiate between an attack & non-attack and then from that categorization technique, rule is implemented in Tmote Sky. The CHA-IDS was evaluated with three kinds of combination attacks known as sinkhole, hello flood, and wormhole. The system was tested for correctness of recognition, energy overhead, and memory usage with the prior 6LoWPAN-IDS execution such as SVELTE and Pongle’s IDS.

Jinku Li et al. (2018) proposed a light weight approach “CodeTracker” to track and protect SMS authorization codes. Precisely, the taint tracking scheme was leveraged to mark the authorization code with taint tags at the origin of the incoming SMS messages (taint sources), and then propagate the tags in the system. The associated array operations, structure, string operations, inter-process communication mechanism, and file procedures for secondary storage of SMS authorization codes were modified to confirm that the taint tags cannot be removed. When the authorization code was sent out using either SMS messages or network connections, enforce pre-defined security policies and data’s taint tag were extracted to prevent the code from being leaked. Then a CodeTracker’s archetype was developed on Android’s ART virtual machine and 1,218 SMS-stealing samples were used to estimate the system.

Abebe Abeshu Diro et al. (2018) proposed a new distributed deep learning-based method for cyber-attack recognition in fog-to-things computing. The fog computing is the extension of cloud computing into the physical world of smart things designed to process events and data closer to the source. The given fog computing architecture provided IoT with the ability of embedded and distributed intelligence in data collection and resource utilization. This means fog nodes are more scalable and responsive for hosting and security services than the cloud. The distributed architecture of fog computing has a double role in that it reduces the storage space and computing power of security functions from IoT devices, and decreases latency issues related to the cloud. Fog nodes are the most efficient spot where attacks can be detected in IoT due to their distribution and resource limitations.

Jianqing Liu et al. (2018) discussed traffic analysis attack on smart homes, where adversaries intercepted the Internet traffic from/to the smart home gateway and profile residents’ behaviors through digital traces. The authors had proposed a privacy-preserving traffic obfuscation framework to achieve the goal. They leveraged smart community network of wirelessly linked smart homes and then purposely forwarded all smart home’s traffic to additional home gateway before going in the Internet. The design mutually took into account the network energy usage and the resource limitations in the IoT devices, while attaining solid differential privacy guarantee so that adversaries can’t relate any traffic flow to a particular smart home. Besides, to make protected multi-hop routing protocols guaranteeing the source/destination unlinkability and fulfilling user’s personalized privacy requirement, a smart community network was considered. Extensive simulations were conducted while evaluating the framework in network energy usage reduction and privacy protection.

Matthew David Smith et al. (2018) reformulated the “multiarmed bandits” (MAB) problem using Bayes-adaptive network security model. With the aim of helping decision-makers assess tradeoffs and fix resources with priorities, connections were added to the network and cyber defense teams were hired. The complete advantages and risks were analyzed during this experiment. Here at indeterminate Poisson-distributed rates, the network defender faced the likelihood of attacks against network nodes. This was in contrast to the general projects with indeterminate likelihoods of success as in the typical MAB problem. This method takes a different dynamic understanding of cyber security investment, discovering how network defenders can optimally assign cyber defense teams among various nodes. A case study about an electric utility was used to test the model taking into account the extent to which they should combine demand response into their smart grid network. The case study helped in finding both the optimal level of connectivity as well as the optimal approach for the successive assigning of cyber security resources.

III. RESULTS

S. No.	Application/Method	Result	Author(year)
1.	Detecting Stealthy False Data Injection Using Machine Learning in Smart Grid (Proposed own technique based on Machine learning)	1) Algorithm 1 used in the paper can detect the anomaly points by applying a threshold δ. 2) For larger values of δ, the algorithm is more sensitive and flags an anomaly for most of the operating points. For smaller values of δ, the algorithm is less sensitive and may lose detection of some attacked points.	Mohammad Esmalifalak et al. (2013)
2.	Machine learning techniques applied to detect cyber attacks on web applications (Using Graph based approach)	1) The result showed that it is possible to achieve significantly better results; almost 94.5% of attack detection and 4.3% of false positives.	Michal Chora et al. (2015)
3.	Detecting Mobile Malicious Webpages in Real Time (kAYO analyzing technique)	1) The proposed technique showed 90% accuracy in classification, and detected a number of malicious mobile webpages in the wild that were not spotted by prevailing methods like Google. 2) The ‘false positive rate’ of kAYO might be lower in practicality, given that mechanisms fail to categorize such pages as spiteful. 3) kAYO was able to detect eight out of the 10 webpages as malicious, whereas Lookout was only able to detect 2 malicious webpages.	Chaitrali Amrutkar et al. (2016)
4.	A Machine Learning Decision-Support System Improves the Internet of Things’ Smart Meter Operations (Using Bayesian Network, Naïve Bayes, Decision Tree and Random Forest)	1) Random forest showed maximum accuracy of 96.69% and an error rate of +/- 1.35% only. After that Naïve Bayes showed accuracy of 96.57% and an error rate of +/- 2.43%. Similarly, Decision Tree and Bayesian Network of (95.34%, +/- 2.61%) and (54.92%, +/-7.35%). 2) For Remote Support Cases and Expected Cost Savings in $, RF comes first, then NB and at last DT.	Joseph Siryani et al. (2017)
5.	ALDOCX: Detection of Unknown Malicious Microsoft Office Documents Using Designated Active Learning Methods Based on New Structural Feature Extraction Methodology (Using ALDOCX and SFEM model)	1) The TPR rate of 93.6% was achieved here through the AL process using only 1,311 docx files (811 initial set + 500 acquired after ten days) out of the total 16,811, which is 7.7% (in comparison to 93.6% of SVM)	Nir Nissim et al. (2017)
6.	Detecting Malicious Activity with DNS Backscatter Over Time (Using DNS Backscatter)	1) It was found that DNS Backscatter can be used as a novel basis for information regarding benign and malicious network-wide activity, including originators of mailings list traffic, CDN infrastructure, spammers and scanners. 2) Technique proved that it can categorize the activity into various classes (ad-track, cdn, cloud, crawler, dns, mail, ntp, p2p, push, scan, spam and update) with good precision.	Kensuke Fukuda et al. (2017)
7.	Distributed intrusion detection based on hybrid gene expression programming and cloud computing in a cyber physical power system (Proposed own ID-HGEP based intrusion detection algorithm)	1) Comparative experiments indicated that the DID-HGEPCloud algorithm has clear advantages in average convergence time, average time consumed, DAR, and false attack rate compared with the traditional GEP, GA, and GP algorithm. 2) DID-HGEPCloud algorithm also showed good	Song Deng et al. (2017)

		performance in terms of speedup and scaleup as the data set size and computing nodes increase.	
8.	Cloud-Based Cyber-Physical Intrusion Detection for Vehicles Using Deep Learning (Using Recurrent Neural Network (RNN) - based Deep learning enhanced by LSTM for robotic vehicle)	1) The model proved to be efficient for the detection in robotic vehicle. 2) It was found that using RNN with LSTM increased detection accuracy.	George Loukas et al. (2017)
9.	Malicious Software Family Classification using Machine Learning Multi-class Classifiers (Random Forest, k-NN and Decision Table)	1) Random forest showed highest accuracy among the three: accuracy of 0.958 (without cross validation) and 0.846(with cross validation) 2) Consecutively, k-NN showed accuracy of 0.958 (without cross validation) and 0.836(with cross validation) 3) And, Decision table showed least accuracy of 0.810 (without cross validation) and 0.751(with cross validation)	Cho Cho San et al. (2017)
10.	The Promise of Machine Learning in Cybersecurity (Using Tensor Flow Architecture)	1) Using previous analyst decisions about 9 million alerts were categorized for training the neural network. 2) As a result, the model showed ~ 99% accuracy for classifying alerts. 3) It was found that 78% of security analyst's time can be reduced using this model.	James B. Fraley et al. (2017)
11.	A High-Order Clustering Algorithm Based on Dropout Deep Learning for Heterogeneous Data in Cyber-Physical-Social Systems (Proposed a high-order k-means algorithm based on a dropout deep learning model)	1) HOK- means algorithm obtains bigger RI values than the HOPCM algorithm 2) HOPCM algorithm produces the highest RI value with 0.88 in the second experiment, but still smaller than the RI value of 0.89 of HOK-means algorithm 3) At the end HOK-means and HOPCM yield the average RI values with 91.2% and 86.4% respectively.	Fanyu Bu et al. (2017)
12.	Feature Selection–Based Detection of Covert Cyber Deception Assaults in Smart Grid Communications Networks Using Machine Learning (Proposed a FS-based CCD assault detection scheme)	1) The proposed FS-based scheme has above 90% performance for all the employed test systems. 2) The CCD assault detection accuracy of proposed scheme is nearly equal to 1.	Saeed Ahmed et al. (2018)
13.	Enhanced Network Anomaly Detection Based on Deep Neural Networks (Different methods such as LSTM, DCNN, ConvAE, Decision-Tree, SVM and k-NN)	1) LSTM (Long Short-term Memory) model showed 89% and 83% accuracy on NSLKDDTest+ and NSLKDDTest21 respectively. 2) DCNN (Deep Convolutional Neural Network) showed 85% accuracy on NSLKDDTest+. While in NSLKDDTest21 runner up was ConvAE. 3) With 82% Decision – Tree, SVM and k-NN had a tie on NSLKDDTest+ while in NSLKDDTest21 Decision tree showed 68%.	Sheraz Naseer et al. (2018)

14.	A Bi-objective Hyper-Heuristic Support Vector Machines for Big Data Cyber-Security (Using Hyper Heuristic Support Vector Machine framework)	1) When compared with individual low-level heuristics, it was found that based on logloss on BIG 2015 lower values were better. Whereas for NSL-KDD based on accuracy higher values were found better. 2) When compared with other algorithms based on logloss for BIG 2015 HH-SVM was found better than XGBoost, Random Forest, Optimised XGBoost. Similarly, based on accuracy for NSL-KDD HH-SVM was better than Gaussian Naïve Bayes Tree, Fuzzy Classifier, Decision Tree.	Naseer R. Sabar et al.(2018)
15.	Embedding Encryption and Machine Learning Intrusion Prevention Systems on Programmable Logic Controllers (An alternative design using open source PLC)	The attacks were performed on a water storage tank. 1)Using secure OpenPLC an Interception Attack was done. It was possible to intercept the packets on the networks but wasn't able to decipher its content due to the AES – 256 encryption. 2) When an Injection Attack was performed, the implanted IPS sensed an abnormal traffic and thus was able to speedily block the attacker node. The running system remained intact after the attack. 3) When a DoS attack was performed, the attacker node was banned within few milliseconds of attack.	Thiago Alves et al. (2018)
16.	A Self-Adaptive Deep Learning-Based System for Anomaly Detection in 5G Networks (Using cuBLAS library)	1) The model doesn't suffer from overhead imposed by the abstraction layers. 2) It was found that cuBLAS performed best among TensorFlow, Caffe2, Theano, PyTorch, MxNet and CNTK. 3) cuBLAS also performed better on the basis of CPU vs GPU performances.	Lorenzo Fernández Maimó et al. (2018)
17.	Malytics: A Malware Detection Scheme (Using Malytics)	1) The study was conducted in two different ways: family detection and chronological novelty detection. 2) Malytics had detected 95.5% of the Mal2017 as zero- day samples. (which is one percent less than ESETNOD32 detection rate) 3) Malytics is more precise and has better hit-rate	Mahmood Yousefi-azar et al. (2018)
18.	Scenario-Based Insider Threat Detection From Cyber Activities (Proposed own technique based on Scenario)	1) It is observed from the figures that the time-series-based classification of user activities outperforms single-day classification. 2) Majority of the existing approaches on insider threat detection aim at detecting only anomalousness, and users exhibiting high degree of anomaly score are suspected as probable insiders.	Pratik Chattopadhyay et al. (2018)
19.	Compression Header Analyzer Intrusion Detection System (CHA - IDS) for 6LoWPAN Communication Protocol (Using Compression Header Analyzer Intrusion Detection System (CHA-IDS))	1) The result conluded that 100% recognition of Hello Flood was shown by Random Forest and J48. They had highest TP rate. 2) In case of Sinkhole attack, 100% of TP rate was shown by Random Forest considering the complete simulation period whereas J48's performance increased at 20 and 30 minutes of simulation period to reach from 99% to 100%.	Mohamad Nazrin Napiah et al. (2018)

		3) For Wormhole attack, considering the complete simulation period, 100% recognition was shown by Random Forest and J48 while 99.95% was shown by remaining algorithms and obtained 99.98% by enhancing 30 minutes of simulation period in their TP rate.	
20.	CodeTracker: A Lightweight Approach to Track and Protect Authorization Codes in SMS Messages (Proposed a CodeTracker)	1) The CodeTracker stored 1,311 target IP addresses and 294 target phone numbers for the stolen authorization codes (from 1,218 sample's information). 2) It was found that locations such as China, USA, and Hong Kong were among the major target addresses and China contributed to it by having 87.66% (1,407/1,605) of the target addresses. 3) On average, CodeTracker introduces an approximate 0.07% overhead with respect to the size of oat files and an approximate1.79% overhead with respect to the compilation time. 4) Among the outcomes, the greatest overhead presented by CodeTracker is 6.92% (String score), whereas the lowest loss is 0.01% (Sieve score). 5) The overhead of the IPC execution time is 0.90%, and the memory usage overheads for the client and server are0.66%and1.17%, respectively	Jinku Li et al. (2018)
21.	Deep Learning: The Frontier for Distributed Attack Detection in Fog-to-Things Computing (Using Fog-to-things Method)	1) In the paper, attacks have been detected with DR of 99.27 percent with the deep learning model, while DR of 97.50 percent has been recorded in the same cases of shallow learning. 2) Based on FAR, deep model showed 0.85 percent while shallow model showed 6.57 percent. 3) For accuracy, deep model showed 99.20 percent while shallow model showed 95.22 percent.	Abebe Abeshu Diro et al. (2018)
22.	EPIC: A Differential Privacy Framework to Defend Smart Homes Against Internet Traffic Analysis (Proposed a privacy-preserving traffic obfuscation framework)	1) In the environment of extensive simulations, the framework showed benefits over the benchmark mechanism in protecting smart home's privacy. 2) It also abated the network energy consumption.	Jianqing Liu et al. (2018)
23.	Cyber Risk Analysis for a Smart Grid: How Smart is Smart Enough? A Multiarmed Bandit Approach to Cyber Security Investment (Using a probabilistic risk analysis framework)	1) It was observed that the optimal strategy is to only connect six of ten possible customer nodes. 2) Optimal connectivity can be assessed through a risk analysis based on existing attack data, engineering models, economic analysis, and expert opinion.	Matthew David Smith et al. (2018)

IV. CONCLUSION AND FUTURE SCOPE

This paper presents an analysis of most of the Machine Learning and Deep Learning algorithms which are widely used in cyber security. Under cyber security, many papers worked on specific cyber applications such as smart homes, authorizing SMS messages, robotic vehicles, optimizing cyber security, intrusion classifications, Big Data cyber security, malicious software, web pages, web applications, and 5G network to prevent these attacks. In addition to different frameworks have been explained along with their taxonomy which introduces the applications of various models/methods such as kAYO, Bayesian Network, Naïve Bayes, Decision Trees, Random Forest, Recurrent Neural Network, k-NN, Dropout, Tensor Flow architecture, Compression Header Analyzer Intrusion Detection System (CHA-IDS), SVM, Malytics, DNS Backscatter, ALDOCX, and SFEM model.

The most promising techniques among all were Random forest, kAYO, Malytics, and ALDOCX. Albeit other methods were upright, they had their own limitations. Random forest gave higher accuracy compared to other ML methods tested in similar environment in all the papers which used it. It was also able to efficiently use large datasets. kAYO was compared with Lookout to track harmful mobile webpages. It gave better results than Lookout. Malytics had proved good in both family and chronological novelty detection having improved hit-rate and precision. ALDOCX worked for signaling harmful MS office documents produced appreciable outcomes as well.

REFERENCES

[1] M. Esmalifalak, Nam Tuan Nguyen, Rong Zheng, Han. Zhu, *"Detecting stealthy false data injection using machine learning in smart grid"*, 2013 IEEE Global Communications Conference (GLOBECOM), pp.**1-9, 2013**.

[2] M. Chora, R. Kozik, "*Machine learning techniques applied to detect cyber attacks on web applications",* Logic Journal of IGPL, Vol. **23**, Issue.**1**, pp. **45–56, 2015**.

[3] C. Amrutkar, Y. S. Kim, P. Traynor. "*Detecting Mobile Malicious Webpages in Real Time", IEEE Transactions on Mobile Computing*, Vol. **16**, Issue.**8**, **2017**.

[4] J. Siryani, B. Tanju, T. J. Eveleigh. "*A Machine Learning Decision-Support System Improves the Internet of Things' Smart Meter Operations. IEEE Internet of Things Journal*, Vol. **4**, Issue.**4, 2017.**

[5] Nir Nissim, Aviad Cohen, and Yuval Elovici, "*ALDOCX: Detection of Unknown Malicious Microsoft Office Documents Using Designated Active Learning Methods Based on New Structural Feature Extraction Methodology*". IEEE Transactions on Information Forensics and Security, Vol. **12**, Issue.**3, 2017**.

[6] K. Fukuda, J. Heidemann, A. Qadeer, "*Detecting Malicious Activity With DNS Backscatter Over Time",* IEEE/ACM Transactions on Networking, Vol. **25**, Issue.**5, 2017**.

[7] S. Deng, A. H. Zhou, D. Yue, B. Hu, L. P. Zhu, "*Distributed intrusion detection based on hybrid gene expression programming and cloud computing in a cyber physical power system"*, IET Control Theory & Applications, Vol. **11**, Issue.**11, 2017**.

[8] G. Loukas, T. Vuong, R. Heartfield, G. Sakellari, Y. Yoon, D. Gan, *"Cloud-Based Cyber-Physical Intrusion Detection for Vehicles Using Deep Learning"*,IEEE Access, Vol. **6, 2018.**

[9] C. C. San, M. M. S. Thwin, N. L. Htun, "*Malicious Software Family Classification using Machine Learning Multi-class Classifiers*", *Computational Science and Technology*, pp. **423–433, 2018.**

[10] J. B., Fraley, J. Cannady, "*The promise of machine learning in cybersecurity"*, SoutheastCon 2017, **2017**.

[11] F. Bu, "*A High-Order Clustering Algorithm Based on Dropout Deep Learning for Heterogeneous Data in Cyber-Physical-Social Systems*", IEEE Access, Vol. **6, 2018.**

[12] S. Ahmed, Y. Lee, S. H. Hyun, I. Koo, "*Feature Selection–Based Detection of Covert Cyber Deception Assaults in Smart Grid Communications Networks Using Machine Learning".* IEEE Access, Vol. **6, 2018.**

[13] S. Naseer, Y. Saleem, S. Khalid, M. K. Bashir, J. Han, M. M. Iqbal, K. Han, *"Enhanced Network Anomaly Detection Based on Deep Neural Networks",* IEEE Access, Vol. **14**, Issue.**8**, pp. **1–15, 2018.**

[14] N. R. Sabar, X. Yi, A. Song, "*A Bi-objective Hyper-Heuristic Support Vector Machines for Big Data Cyber-Security",*. IEEE Access, Vol. **6, 2018.**

[15] T. Alves, R. Das, T. Morris, *"Embedding Encryption and Machine Learning Intrusion Prevention Systems on Programmable Logic Controllers"* IEEE Embedded Systems Letters, Vol. **1, 2018.**

[16] L. Fernandez Maimo, A. L. Perales Gomez, F. J. Garcia Clemente, M. Gil Perez, G. Martinez Perez, *"A Self-Adaptive Deep Learning-Based System for Anomaly Detection in 5G Networks",* IEEE Access, Vol. **6, 2018.**

[17] M. Yousefi-Azar, L. Hamey, V. Varadharajan, S. Chen, *"Malytics: A Malware Detection Scheme*", IEEE Access, Vol. **4**, pp. **1–14, 2018.**

[18] P. Chattopadhyay, L. Wang, Y. P. Tan, *"Scenario-Based Insider Threat Detection From Cyber Activities*", IEEE Transactions on Computational Social Systems, pp. **1–16, 2018**.

[19] M. N. Napiah, M. Y. I. Bin Idris, R. Ramli, I. Ahmedy, *"Compression Header Analyzer Intrusion Detection System (CHA - IDS) for 6LoWPAN Communication Protoco".* IEEE Access, Vol. **6, 2018.**

[20] J. Li, Y. Ye, Y. Zhou, J. Ma, *"CodeTracker: A Lightweight Approach to Track and Protect Authorization Codes in SMS Messages",* IEEE Access, Vol. **6, 2018.**

[21] A. Abeshu, N. Chilamkurti, "*Deep Learning: The Frontier for Distributed Attack Detection in Fog-to-Things Computing",* IEEE Communications Magazine, Vol. **56**, Issue.**2**, pp. **169–175, 2018**.

[22] J. Liu, C. Zhang, Y. Fang, *"EPIC: A Differential Privacy Framework to Defend Smart Homes Against Internet Traffic Analysis*", IEEE Internet of Things Journal, Vol. **5**, Issue.**2**, **2018**.

[23] M. D. Smith, M. E.Pate-Cornell, *"Cyber Risk Analysis for a Smart Grid: How Smart is Smart Enough? A Multiarmed Bandit Approach to Cyber Security Investment",* IEEE Transactions on Engineering Management, Vol. **65**, Issue.**3**, **2018**.

[24] D. Mallampati, *"An Efficient Spam Filtering using Supervised Machine Learning Techniques",* International Journal of Scientific Research in Computer Science and Engineering, Vol. **6**, Issue.**2**, pp.**33-37**, **2018**.

[25] B. Wahyudi, K. Ramli, H. Murfi, *"Implementation and Analysis of Combined Machine Learning Method for Intrusion Detection System",* International Journal of Communication Networks and Information Security, Vol. **10**, No. **2**, **2018**.

Authors Profile

Sagar Bansal is presently an undergrad student at Department of Computer Science, Bhaskaracharya College of Applied Sciences, University of Delhi, Delhi, India. He is pursuing Bachelor of Science (Honors) in Computer Science. His research interests lie in Data Science, Machine Learning, Deep Learning, Mining and Big Data Analytics.

Anshika Singh is Assistant Professor, Department of Computer Science, Bhaskaracharya College of Applied Sciences, University of Delhi, Delhi, India. She had done her B.Tech from JSSATE Noida and M.Tech from USIT, Dwarka. She has 4.5 years of teaching experience. Her research interest lies in Text Mining, Machine Learning, Deep Learning and Data Analytics.