NIPES

Journal of Science and Technology Research

Journal homepage: www.nipesjournals.org.ng

A Hybrid Supervised Machine Learning Model for the Prediction of Insider Threats

[a]Adun I. J. & [b]Amadin F. I.

[a,b] Department of Computer Science, Faculty of Physical Sciences, University of Benin, Nigeria.
[a] ivie.okungbowa@uniben.edu, [b]frankamadin@uniben.edu

Article Info

***Keywords*:** *Threat, Machine Learning, Training, Support Vector Machine*

Received 11 May 2023
Revised 29 May 2023
Accepted 21 June 2023
Available online 3 Sept. 2023

https://doi.org/10.5281/zenodo.8313125

Abstract

The quest and sensitivity of organizational resources has permeated need for information confidentiality while ensuring availability and integrity are met, if organizations are to thrive and survive. To fend off malicious insider, organizations have implemented strategies, policies and techniques to manage malicious insider attacks. Machine Learning (ML) algorithms are implemented as learning paradigms, having the ability to learn from prior instances. ML present intelligent implicitly designed models having the capability of predicting possible outcomes from machine learning dataset based on perceived features which might be computationally explored. Hybrid Supervised Machine Learning Model for the Prediction of Insider Threats (HSMLM-IT) has been designed, simulated and validated utilizing Support Vector Machine (SVM) for label classification and Adaptive Neuro Fuzzy Inference System (ANFSI) for predictive learning. The SVM blocks provides a classification accuracy of 92% and precision of 93% while the ANFIS training blocks provides an ANFIS accuracy of 91% and ANFIS error of 9%.

1.0. Introduction

The advert of ICT in collaboration with smart devices and network facilities has enhanced the pervasiveness of data and information leading to the facilitating and dissemination of information resources across geographical spheres. Furthermore, these technologies, supported by the ever-growing need for globalization have enhanced the productivity and availability of information and data [1]. With the passage of time, on-premises information access has been extended remotely, thanks to the synergy provided by these technologies, collectively providing an array of organizational resources. The quest and sensitivity of organizational resources has permeated need for information confidentiality while ensuring availability and integrity for organizations to thrive and survive. Organizational resources must be confided to authorized users in a complete and timely manner [1]. Organization resources such as sensitive information, proprietary files, network infrastructures and intellectual properties are valuable assets which are threatened by insiders-employees or contractors bestowed with access right, credentials and privileges to access organizational resources [2].

Insiders are usually classified into non-malicious and malicious with the later known for deliberate misuse of organization's credentials to cause mayhem to the organization while the former is an unintentional threat to organizational resources due to negligence or carelessness in performing

organizational task [3]. These cherished organizational resources have inspired the growth of malicious insiders. Insider attacks can be classified into pawns, goofs, collaborators and lone wolves with any of these threats resulting in significant damage such as compromise of sensitive information, integrity and availability violations, ICT sabotage, malicious data exfiltration and accidental data breach [4]. The fastidious activities of insiders globally have indeed become a massive threat to organizational resources [5].

The statical reports for insider attacks globally is indeed alarming, for instance, between 2018 and 2020, 47% increase in the frequency of insider attacks were recorded [6,7]. Another report predicts that the occurrence of insider data breaches will increase by 8% through 2021 [8,9]. The latest report of data breaches as reported by the Verizon 2021 data breach investigations report shows a 22% security incidence for insider attacks [7]. Furthermore, it has also been established that 60% of data breaches globally are caused by insider threats [10]. Moreover, in 2020 alone, 68% of organizations observed that insider attacks have become all too common [11, 12]. Presently, the global pandemic has not been helpful; in fact, malicious insiders are on a sprint developing and enhancing attacks mechanism, exploiting fears, uncertainties and the current unstable economic climate posed by the pandemic to cause mayhem [13]. These activities have persisted, evolving both in scope and size, causing havoc for businesses and organizations alike. Comprehending the intent of malicious insiders towards organizational resources and preventing such attacks across enterprise's systems are paramount, if organizations are to outlive these persistent onslaughts from insiders [14].

To fend off malicious insider, organizations have implemented several strategies, policies and techniques to manage malicious insider attacks. Unfortunately, the extent of insider attacks is convoluted and discombobulated [3, 9, 1], causing rapid obsolesce while creating vulnerabilities and threats alike, possibly explored to cause serious damage. These demerits have fostered the need to evolve stable techniques, capable of protecting organizations resources. Organization and business have sort to addressed insider threat using machine and non-machine learning techniques [15]. Some of the non-machine techniques include block chain, audit data source, designing frameworks such as Coburg Utility Framework (CUF), Tableau and Limkurious platforms, role-based access control, scenario based, decoys and honeypots and risk analysis using psychological factors [15]. The demerits associated with non-machine learning techniques such as restricted application space, inability to handle sparse data with high dimensionality, resource mismanagement and inability to automate repetitive task has open the way for the adoption of machine learning algorithms [16, 15].

2. Related Work

Machine Learning (ML) algorithms are implemented as learning paradigms, having the ability to learn from prior instances. ML present intelligent implicitly designed models having the capability of predicting possible outcomes from machine learning dataset based on perceived features which might be computationally explored [17]. ML could be implemented either as Conventional Machine Learning (CML) and Hybrid Machine Learning (HML) paradigms, exploring supervised, unsupervised or reinforced learning algorithms [17, 15]. The selection of machine learning algorithm for implementation is dependent on size and formation of dataset, target value, computational time and learning paradigm. Despite the fundamental benefits associated with CML algorithms employed for the prediction of insider threat by previous researches, its prediction at best, perhaps is still questionable due to the frequent escalation of inside attacks. Could these escalations of insider attacks be caused by the negativities associated with CML algorithms utilized for predicting insider attacks? Could these escalations of insider attacks be addressed utilizing HML algorithms for predicting insider attacks? Would a comparative analysis between CML and HML algorithms provide a better understanding pertaining to insider prediction? These aforementioned

questions have created necessary issues needing prompt resolution. Non-machine techniques have been implemented, but the issues of restricted spaces, inability to handle sparse data with high dimensionality, resource mismanagement and inability of addressing repetitive task thrust researchers toward implementing Machine Learning (ML) algorithms either conventional or hybrid. Lately, CML has been implemented extensively for insider threats [16], yet the occurrence has not abated. The escalation of insider threat has been predicted to reach an unprecedented height [12, 10, 11]. With the prediction of ML algorithms been worrisome due to increase of malicious insiders, loss of proprietary properties and the cost implications incurred by organizations resulting from these unwanted losses. It is the intent of this research to comparatively analyze the prediction of two ML algorithms - Adaptive Neuro Fuzzy Inference System (ANFIS) and Support Vector Machine (SVM) for the prediction of insider attacks.

3. Materials and Method

The design of the Hybrid Supervised Machine Learning Model for the Prediction of Insider Threats (HSMLM-IT) was built on the premise of Artificial Intelligence (AI) specifically Support Vector Machine (SVM) and Adaptive Neuro-Fuzzy Inference System (ANFIS); which was utilized as the pillar of the model. This model improves on single methods like SVM and ANFIS by combining the strengths of the two methods whereby SVM performs dataset classification enabling labelled dataset while ANFIS provides flexible machine learning. The designed HSMLM-IT model is depicted on Figure 1.

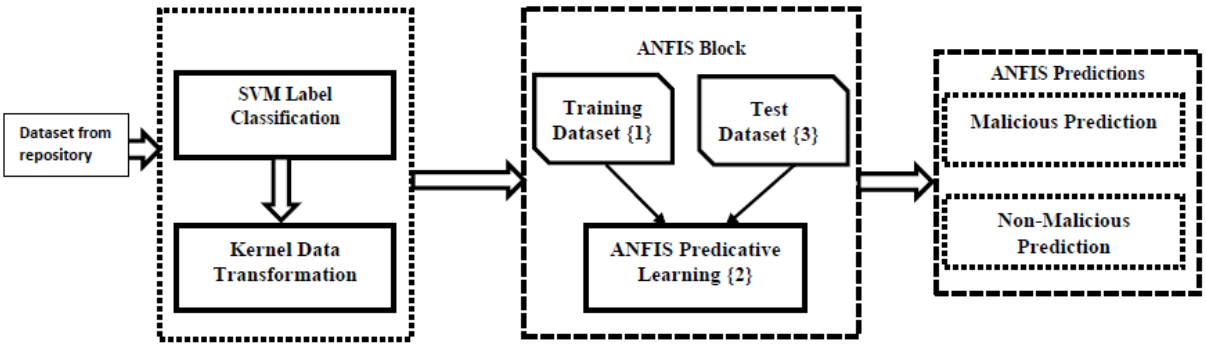

Figure 1: Hybrid Supervised Machine Learning Model for the Prediction of Insider Threats (HSMLM-IT)

3.1 Functional Design for the HSMLM-IT

The design concept spans both the integral tools and techniques covering Support Vector Machine (SVM) block and Adaptive Neuro-Fuzzy Inference System (ANFIS) block.

3.2 SVM Label Classification

The SVM label classification is the first block of the HSMLM-IT. This block is concerned with classifying the dataset into distinct labels prior machine learning. It is tasked with receiving the dataset and transmitting it to kernel data transformation, with the aim of transforming the dataset into label classes, categorized as malicious and non-malicious classifications.

3.3 ANFIS Block

The ANFIS is the second block of the Hybrid which is tasked with ensuring that predictive learning is achieved. It ensures that proper training is achieved utilizing hybrid training, catering both for forward and backward pass. The ANFIS phase employ six layers (0-5).

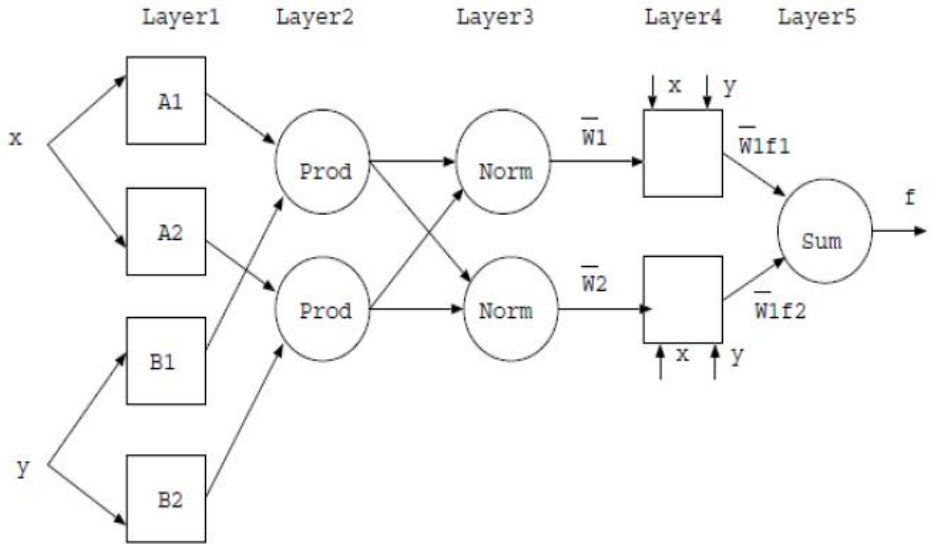

Figure 2: Adaptive Neuro Fuzzy Inference System (ANFIS)

3.3.1 Input Layers 0 {x, y}

The training input employs the labeled SVM dataset which has been merged into a universal ANFIS training data. This data is transmitted to the ANFIS input layer (Layer 0). The dataset comprises of six (06) attributes: vectors, date, user, source, action and insider threat. X and y at the far left identify the input into the model.

3.3.2 Membership Layers 1 {A1, A2, B1, B2}

The training membership function map variables input to varied membership using the appropriate membership function identifiable for the work. Several memberships exist with varied pros and cons. Suitability for training systems, smooth curve at switching point and acceptance of non-linear functions- gaussian membership function and generalized bell membership functions. Generally, generalized bell membership function is utilized as an assigner due to the availability of one more premise parameter more than gaussian membership function, enhancing it degree of freedom to adjust the steepness at the crossover (switching) points.

$$u\,(x) = \frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}} \qquad (1)$$

Where:

a = the mean of parameter values

b = bias, controls the slop of the curve of x at -b/2a, c-a, c+a, b/2a

c = centre of the curve

x = case fuzzy set decision variables value

The parameters a,b,c are referred to as premise or activation parameters.

3.3.3 Training Rule - Layers 2 {prod, prod}

The training rules are built on operational data comprising of fuzzy rule, fuzzy AND operator, input variable and varied membership [low, moderate, high] which is utilized in producing the insider threat IF-THEN rules while minima value for training. Mathematically, the training rule is presented as

$$O_{3,i} = W_i = u_{A_i}(x_1)\, u_{B_i}(x_2) u_{C_i}(x_3) \ . \ . \ . u_{N_i}(x_n) \quad (2)$$

Where:

i = 1, 2, … n (total number of training cases)

A, B, C, … N = total number varied input variables

$X_1, X_2, X_3, \ldots, X_n$ = total number of varied input variable memberships

Wi = output of the ith training case

The HSMLM-IT model addresses rule redundancy at the point of training (learning) using a supervised hybrid training algorithm. The training weights are repeated adjusted and rule constituent outputs recomputed with the aim of presenting multiple varied training rules to HSMLM-IT. These dual processes (adjustment and re-computation) are achieved through the HSMLM-IT ANFIS epoch. The epoch comprises of forward and backward pass sponsored by Least Square Estimator (LSE) and Back Propagation Gradient Descent (BPGD). The HSMLM-IT ANFIS forward pass forms and adjusts the consequent parameters producing consequent outputs while the HSMLM-IT backward pass adjusts the premise parameters of the activation functions. BPGD adjust the parameters of activation captured on a membership input space (layer 1) by propagating the error signals backward recourse to the bias update of -1, 1 and 0, thereby shifting the input membership space either to the right, left or maintaining a constant input point respectively.

3.3.4 Training Normalization - Layers 3 {norm, norm}

The training normalization obtains the case rule value which is subsequently used in obtaining the normalized value of a given insider case. The i^{th} normalized value is obtained by dividing the sum of the entire case from the rule layer. This value represents the contribution of a given case to the defuzzification value. Mathematically, it is represented as:

$$O_{4,i} = \bar{w}_i = \frac{w_i}{W1+W2+W3 \ . \ . \ . \ Wn} \quad (3)$$

Where:
i = 1, 2, …, n (total number of training cases)
Wi = output of the ith case from the rule layer
$\bar{W}_i = output of the ith sample in the normalisation layer$

3.3.5 Training Defuzzification - Layers 4 {↓x ↓y}

The defuzzification value of each case is obtained using the normalized value membership value and pre-processed value respectively in producing the weighted consequent case values. The weighted consequent value of a given case rule is presented mathematically as:

$$O_{5,i} = \bar{w}_i f_i = \bar{w}_i (p_i x_1 + q_i x_2 + r_i) \quad (4)$$

Where:

$\{p_i, q_i, r_i\}$ = insider threat consequent(membership) parameters

$\{X_1X_2\}$ = Insider threat pre-processed values

$$\overline{W_i} = output of the insider threat ith case from the normalisation layer$$

$$\overline{W_i} f_i = defuzzification output of the insider threat ith case in the fifth layer.$$

3.3.6 Output - Layers 5

The output provides the final epoch summation value after a finite membership weight adjustment and weight computation by the membership and defuzzification layer respectively. Mathematically, it is represented as:

$$O_{5,i} = \sum_{i=1}^{n} \overline{W}_i f_i \qquad (5)$$

$Where$:

$$\overline{W_i} f_i = defuzzification output of the insider threat ith case in the fifth layer.$$

$i = 1, 2 \ldots n$ $(total\ number\ of\ insider\ threat\ ase\ in\ the\ fifth\ layer)$.

$O_{5,i} = final target output.$

3.4 Design of HSMLM-IT

The HSMLM-IT software designs cover all facet of the activities involved in conceptualizing and representing system modelling. The HSMLM-IT was designed with Unified Modelling Language (UML), focusing on two main views: user and behavior views. These views were subsequently represented using use case diagram and sequence diagram.

3.4.1 Use Case Diagram

The Hybrid Supervised Machine Learning Model for the Prediction of Insider Threats (HSMLM-IT) use case diagram shown in Figure 3 identifies the view of the system from the user perspectives. The use case diagram depicts the various interactions made between the system and the user. It portrays the various views accessible to the user. The processes include: *svmLabelData*, and *anfisPrediction.* The initiating and receiving actors are security administrators, granted system access right. The processes are sequentially executed, with sequential processes largely dependent on previous phases. The system boundaries and environment limit the system and defines the network administrator processes. Figure 3 depicts the HSMLM-IT user case diagram.

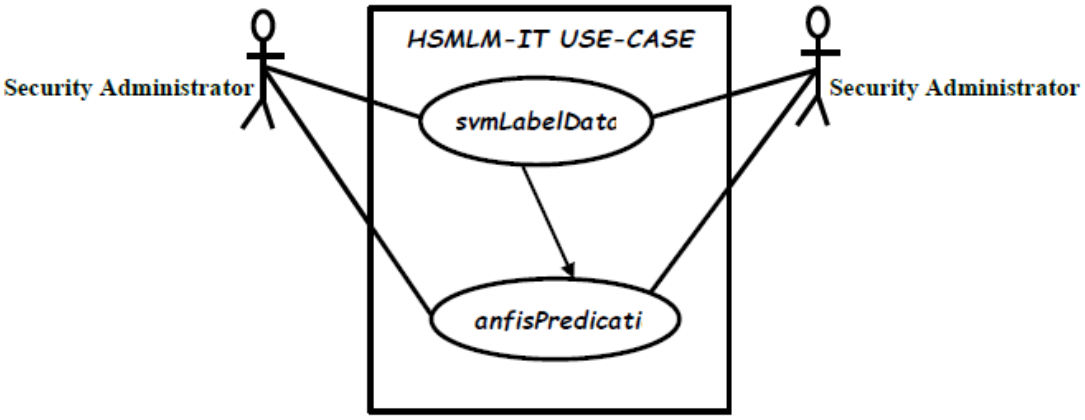

Figure 3: HSHMLM-IT Use-case Diagram

The use-case diagram portrays the HSHMLM-IT from the user perspective. It presents the modules the security administrator interacts with. Figure 3 presents the security administrators serving as both the initiating and receiving actors (user); system boundaries, ellipse as use cases, system

environment and several associations depicting the communication between security administrator and HSMLM-IT through the system boundaries and environment. The security administrator presents the data to the system. Prior to *anfisPrediction*, the dataset is labelled utilizing the *svmLabelData.* The *anfisPrediction* apply the fundamentals of ANFIS in training the dataset for predicting insider attacks. These processes are executed sequentially.

3.5 Implementation Dataset

The insider dataset used for implementing the HSMLM-IT model was obtained from an online repository- https://www.kaggle.com/nitishabharathi/cert-insider-threat/version/1. The dataset comprises of seven (07) variables which includes number, insider threat, vector, data, user, source, action. The dataset was preprocessed in other to expunge redundant and irrelevant features. The expunged features were number, insider threat and vector due to lack of correlation with the overall data. These remaining features led to effective prediction.

3.5.1 HSMLM-IT Model Interfaces

The HSMLM-IT interfaces portray the integral implementation of the HSMLM-IT. The interface was designed with Java programming language. The interface captures the implementation of the SVM block prior to the ANFIS training which was implemented using Java. The interfaces were built applying the fundamentals of java programing including packages classes, comments, return types access modifiers- public, private and protected, class members – methods, variable, objects as shown in Figure 4.

Figure 4: HSMLM-IT Buttons Interface

Figure 4, provide the Hybrid Supervised Machine Learning Model for the Prediction of Insider Threats (HSMLM-IT) frontier interfaces. The interfaces present the various buttons which initiate and execute the fundamental processes. The processes are executed sequentially- Load, Perform SVM, Generate ANFIS, Train ANFIS and Test ANFIS.

3.5.2 Support Vector Machine (SVM) Classification Simulation

The dataset retrieved were unlabeled thus the implementation of SVM. The SVM classification provides the dataset sample in respective classified labels. These classifications ensure ANFIS training is enacted with desirable results. This is significant owning to the fact that ANFIS trains with target (labeled) value. Figure 5 present the SVM label classification. The SVM label classification presented in Figure 5 identifies dataset which has been holistically partitioned into respective classes (labels). The graph plots user actions against varied scores. From figure 5, the

variation in graph markers between red and white sticks signify the present of insider attacks (present) and non-insider attacks (absent). The score range defines the range occurrence and absent. A score of 1900 and 2000 identify the highest score.

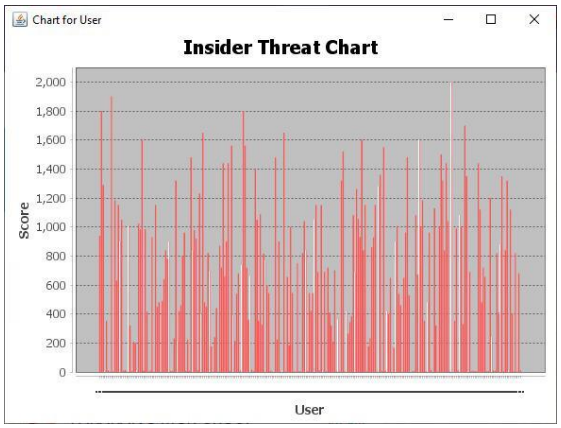

Figure 5: SVM Label Classification chart

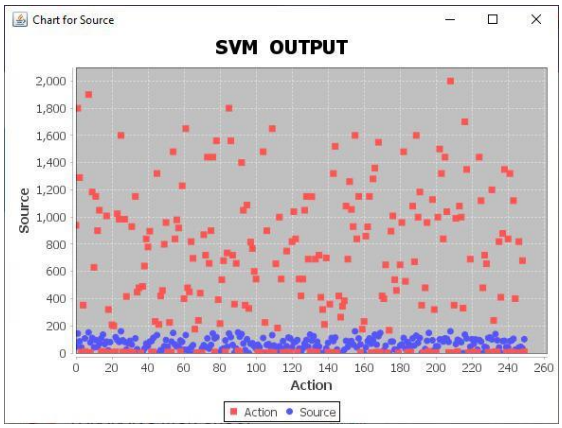

Figure 6: SVM Output Chart

The SVM output chart presented in Figure 6 presents the output from the SVM classification. The charts portray action plotted against source. The total distinction of chart markers (blue or red) presents the optimality of SVM label classification. From figure 6, the variation in graph markers between red and blue dots signify the present of insider attacks and non-insider attacks which have been properly differentiated. Figure 7 show some of the extension of SVM output charts portrayed on Figure 6 which holistically presents each sample data values. These values are comparable with the plot found on Figure 6. These values are cumulatively utilized in producing the accuracy and precision value.

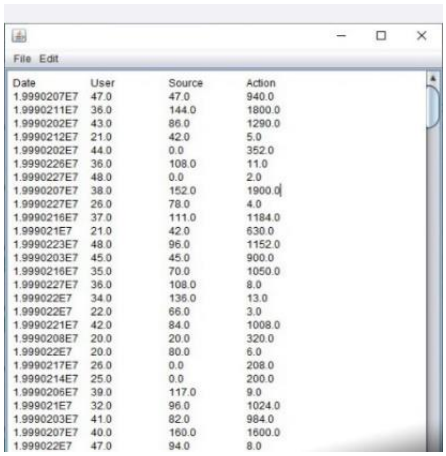

Figure 7: SVM Output Chart

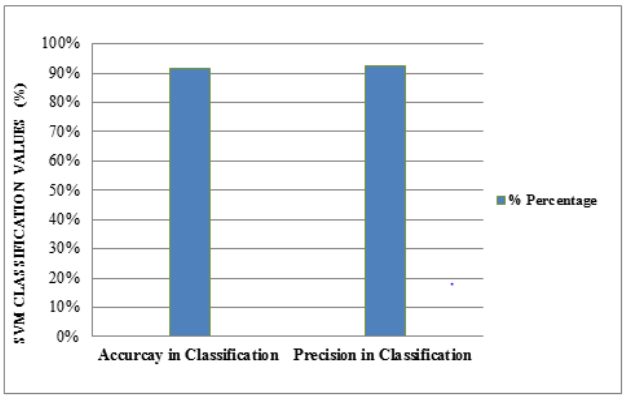

Figure 8: SVM Classification Graph

The graph of Figure 8 statistically validates the SVM training in terms of accuracy and precision. Accuracy identifies how close or far off measures are to the true value while precision identify the dispersity or closeness to values. It shows the accuracy values of 92% while precision value of 93%.

4. Results and Discussion

The Adaptive Neuro Fuzzy Inference System (ANFIS) Simulation presents the fundamentals of ANFIS training exhibited on the SVM classified labeled data. The dataset was subdivided into training and testing. A total of 350 sample dataset was employed for the ANFIS training. The dataset is subdivided into training -250 samples and testing -100 samples. The graph of Figure 9 graphically represents these portions in percentage – 100%, 71% and 29% while Figure 10 – Figure 13 presents the ANFIS simulation interface

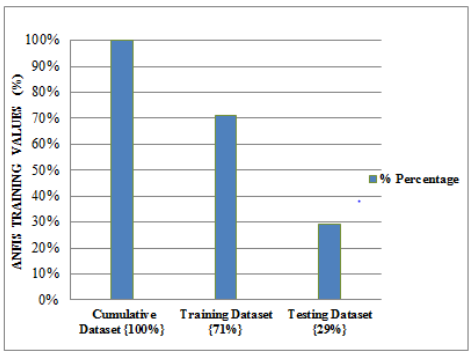

Figure 9: ANFIS Training Dataset Graph

Figure 10 presents the ANFIS training data which is the output of the SVM classification block. The dispersed nature of the dataset along the chart shows the usefulness of the dataset in terms of training patterns and signature having the propensity to produce desired results. Figure 11 presents the ANFIS training across several epoch with associated errors. The graph line which is uneven portrays the ANFIS training which is either overfitting or overfitting to noise. Epoch 1 to Epoch 19 shows a flat training line with 0.00. Epoch 19 presents training value of 0.64. Epoch 26 presents 0.90 while the highest training error of 0.93 is presented by epoch 29. Figure 12 presents the ANFIS testing interface. The training percentage is presented in Figure 13 presents ANFIS validation with accuracy of 91% and error of 9%.

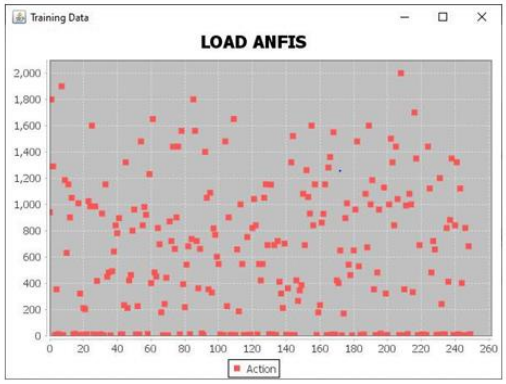

Figure 10: ANFIS training data

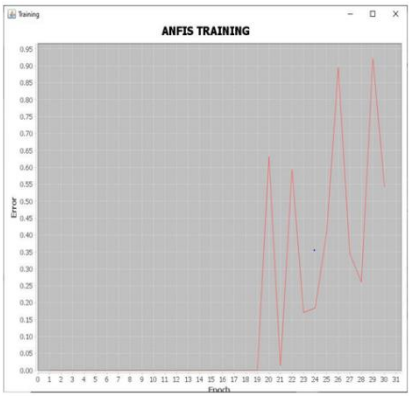

Figure 11: ANFIS Training

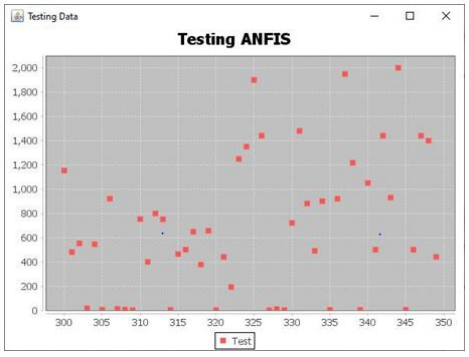

Figure 12: ANFIS Testing

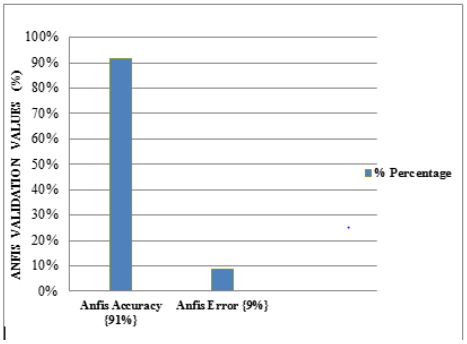

Figure 13: ANFIS Testing Graph

5. Conclusion

Insider attacks have persistently bedeviled organizations. This dilemma, has posed a detrimental situation with tremendous effect experienced globally, affecting performance and productivity. These issues are addressed implemented organizational policies and procedures on security and privacy. Overtime, organization have also designed and implemented software security approach, in addition to address network security to combat insider attacks. Although, these has addressed insider attacks to a considerable extent resulting in cost saving, resources management and performance. The unpredictability of insider attack patterns has thrust researchers into the adoption of intelligent machine learning approach in addressing insider attacks. This research implements a Hybrid Supervised Machine Learning Model for the Prediction of Insider Threats (HSMLM-IT). The HSMLM-IT was conceptualized utilizing Support Vector Machine (SVM) and Adaptive Neuro Fuzzy Inference System (ANFIS). The former method was implemented to classify the retrieved unlabeled dataset. Three Hundred and Fifty (350) dataset samples were labeled Therefore, SVM assign a label to each dataset components. The later method was applied in machine training with dataset subdivided into 250 (71%) training and 100 (29%) testing samples The HSMLM-IT model functionalities was captured using Unified Modelling Language (UML). The design present two main views – user and dynamic view. These views were presented using use case and sequence diagrams. The HSMLM-IT was simulated using Java programming language providing several buttons – Load Data, Perform SVM, Generate ANFIS, Train ANFIS and Test ANFIS. The various simulated interfaces present the fundamentals of SVM and ANFIS. HSMLM-IT SVM blocks provides a classification accuracy of 92% and precision of 93% while the ANFIS training blocks provides an ANFIS accuracy of 91% and ANFIS error of 9%.

References

[1] Wisley, C. (2020), "*Confidentiality, Integrity and Availability"*. Retrieved http://resource https://whatis.techtarget.com/definition/Confidentiality-integrity-and-availability-CIA

[2] Magklaras G. B. and Furnell S. M. (2005). "A Preliminary Model of End User Sophistication for insider threat prediction in IT systems". Computers & Security, Pp. 371-380.

[3] Al tabash, K.and Happa, J. (2018)."*Insider - threat Detection Using Gaussian Mixture Models and Sensitivity Profiles*". Computer & Security, pp. 1-22.

[4] Mohammed, N. A., Rabiah, A., Z. Zainal, A., Warusia, Y., Aslinda H., Karrar Hameed A., Nabeel, S. A., and Zahri, Y. (2020). "*A Review of Insider Threat Detection: Classification, Machine Learning Techniques, Datasets, Open Challenges and Recommendation".* Journal of Applied Sciences, Vol.10, Pp. 2-41.

[5] Stewart, D. (2019). *IBM Study Reveals cyber–Security Data Breach Cost*. Retrieved resource https://totalcrypto.io/ibm-study-reveals-cyber-security-data-breach-costs/ 19 July 30, 2021.

[6] ObserveIT (2020).The Real Cost of Insider Threats in 2020. Retrieved from ObserveIT

[7] Maddie, R. (2021). "*Insider Threat Statistics You Should know*: Update 2021". Retrieved https://www.tessian.com/blog/insider-threat-statistics.[Accessed December 1, 2021]

[8] Bitglass (2020). "*Bitglass' 2020 Insider Threat Report",* Retrieved at https://www.bitglass.com/press-releases/bitglass-2020-insider-threat-reportfrom Bitglass, August 10, 2021

[9] Shey (2020). "Predictions 2021: The Path to A New Normal Demands Increased Cybersecurity Resilience. Retrieved from Forrester.

[10] Goldstein (2020).*What Are Insider Threats and How Can You Mitigate Them*?Retrieved from Security Intelligence September 13, 2021.

[11] IBM (2020).*Cost of Insider Threats*: Global Report 2020. Retrieved from IBM.

[12] Cybersecurity Insiders (2020)."*Insider Threat Report".* Retrieved online from https://cybersecurity.att.com/resource-center/analyst-reports/insider-threat-report [Accessed December 26, 2021]

[13] Jurgen, S. (2020).*Interpol report shows alarming rate of cyberattacks during Covid-19*. Retrieved https://www.interpol.int/News-and-Events/News/2020/Interpol-report-shows-alarming-rate-of cyberattacks-during-COVID-19.[Accessed December 30, 2021].

[14] Ko, L. L., Divakaran, D.M.; Liau, Y. S., Thing, V.L.L. (2017) "*Insider Threat Detection and its Future Directions*". International Journal of Security and Network, Vol. 12, Pp.168–187.

[15] Oladimeji T. O., Ayo C.K. and Adewumi S.E. (2019).*Review on Insider Threat Detection Techniques. Journal of Physic: conference Series 1299*, Retrieved https://iopscience.iop.org/article/10.1088/1742-6596/1299/1/012046/pdf.

[16] Choras, M. and Kozik, R. (2018)."*Machine Learning Techniques for Threat Modelling and Detection*". Security and Resilience in Intelligent Data-Centric Systems and communication Networks, Pp. 179 – 192

[17] Jason, B. (2019), "*How to choose a Feature Selection Method for Machine Learning",* Retrieved online Url using the link- https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/ [Accessed December 21, 2021]