A Predictive User Behaviour Analytic Model for Insider Threats in Cyberspace

Olarotimi Kabir Amuda[1], Bodunde Odunola Akinyemi[2], Mistura Laide Sanni[3] and Ganiyu Adesola Aderounmu[4]

[1234] Department of Computer Science and Engineering, Obafemi Awolowo University, Ile-Ife, Nigeria.

***Abstract*:** Insider threat in cyberspace is a recurring problem since the user activities in a cyber network are often unpredictable. Most existing solutions are not flexible and adaptable to detect sudden change in user's behaviour in streaming data, which led to a high false alarm rates and low detection rates. In this study, a model that is capable of adapting to the changing pattern in structured cyberspace data streams in order to detect malicious insider activities in cyberspace was proposed. The Computer Emergency Response Team (CERT) dataset was used as the data source in this study. Extracted features from the dataset were normalized using Min-Max normalization. Standard scaler techniques and mutual information gain technique were used to determine the best features for classification. A hybrid detection model was formulated using the synergism of Convolutional Neural Network (CNN) and Gated Recurrent Unit (GRU) models. Model simulation was performed using python programming language. Performance evaluation was carried out by assessing and comparing the performance of the proposed model with a selected existing model using accuracy, precision and sensitivity as performance metrics. The result of the simulation showed that the developed model has an increase of 1.48% of detection accuracy, 4.21% of precision and 1.25% sensitivity over the existing model. This indicated that the developed hybrid approach was able to learn from sequences of user actions in a time and frequency domain and improves the detection rate of insider threats in cyberspace.

***Keywords*:** Insider threats, Deep learning, Convolutional Neural Network, Gated Recurrent Unit, User Behaviour

1. Introduction

The advent of Information Systems and web technologies have created an entity called cyberspace which hinges on the operational integration of communication infrastructure and interaction of people, their values and personal interest. Cyberspace technology is widely applied in the area of automated processes such as on-line transactions, data management and storage through the internet [1]. Legitimate users within a specified cyberspace infrastructure should be able to have access to the right information based on the users' roles. A system in cyberspace should also be self-protected against rogue users among legitimate users. This is to ensure that that Services were provided on time, at the right location, in the right condition, meet quality standards, and are achieved at the lowest cost possible [2]. However, the flexibility cyberspace presents has raised a lot of security concerns. Increasing business dependency on information systems has made many more organizations vulnerable to cyber-attack and its consequences. The cyberspace technology is now repeatedly challenged by vulnerabilities, risks and threats which have constantly causing the exploitation of information system resources and the users. Therefore, automated tools are often required to dynamically detect user threats in organization cyberspace.

The well-known cyber-attacks are broadly classified into external attacks and internal attacks. In the past, threats coming from outside, such as Denial of Service (DoS), phishing, hacking, etc. use to be prevalent and organizations have designed various techniques or mechanisms to thwart the threat from outsiders. The attack from insider is unlike outsiders attack because insider has direct and legal right to access any information in the organisation. An insider attacker has rights and privileges to access information and intentionally misuses the rights and avail it to competitors [3]. The misuse of information systems by insiders is mostly centres on modifying and destruction of organisation data [4] and these illegal activities are perpetrated by technical persons. Major cyber-attacks of insiders are implemented by any form of filtration, data corruption and denial of service which result in huge losses to organizations including damage to reputation. A study of cyber-crime activities in the government sectors shows that 24% of incidents are carried out by unauthorized privileges users of which 11% involve illegal installation [5]. Although the majority of cyber-attacks are external attacks but insider attacks are often more damaging and costly due to the knowledge and access privilege to information systems within the organisation [6].

Insider attacks are very difficult to detect because they are not breaking access control but have enough privilege within the domain of access and take advantage of the opportunity to use it in a treacherous way. Insider threat is a growing threat to the world's businesses, governments and corporate entities. This considerable growth of the internal threat has made the traditional tools like Firewalls and Endpoints Anti-virus insufficient on their own. The current practices by most organizations to contain insider threats tend to be reactive. They are useful after the exploit has happened, therefore, there are no inferences into or predictive perceptions of the potential insider threat indicators. As a result of this, the ease at which rogue users get away with their nefarious act of information security breach and difficulty of tracking and preventing them is at an alarming rate. Therefore, timely detection of insider attacks on valuable data and sensitive information become very valuable in preventing organization from huge loss.

The complexity of the internal threat is extremely high due to abuse of trust, privacy and ethics. Employees that exhibits the traits of introversion characteristics like failure to take responsibility for their actions, prejudice of critiques, self-perceived value exceeding achievements, callousness, predisposition toward law enforcement, pattern of discontentment, and ineffective crisis management etc., usually have the tendencies to engage in malicious activities against the organization. Organization needs to understand current employee behaviour and make sure employees understand Information Technology (IT) ethics and

principles. Understanding of how people are accustomed to their past behaviour as being permanent or characteristic attributes will lead to the discovery of deviations in these principles. Therefore, a good understanding of these behavioural indicators is essential for the early detection of malicious tendencies.

Several techniques have been proffered to address the challenges of insider threat. Among the well-known insider detection techniques are the monitoring of employees' behaviour, their access to systems and mails, use of CCTV camera and enforcing stringent IT security policy, training employees to identify and report abnormal behavioural displayed by their peers or business partners [7]. Also, different algorithms have been developed to unravel anomalous behaviour of employees. An anomaly detection method tries to detect anomalous behaviour by comparing the pattern of user behaviour with known available malicious patterns and signatures. The change in normal behaviour indicates the event is either an unintentional or intentional attacks [8]. Some of the mechanisms used to address anomalous behaviour challenges are, the signature-based detection mechanism, the principle of least privilege technique, data mining-based detection methods- supervised learning unsupervised learning and deep learning Algorithms [7].

Generally, in cybersecurity management, a single algorithm may not adequately produce accurate prediction of internal threats due to its complexity nature. It is a known fact that insider attackers' activities are dynamic in nature, these existing models have the challenges of not being able to detect unknown or temporal user behaviour pattern, and therefore several abnormal user behaviour goes undetected. Furthermore, most existing models could not learn from the sequence of user behaviour thus leading to a low detection rate and high false alarm rate.

Thus, an attempt is made in this study to employ a hybrid technique that will dynamically detect the insider threats in cyberspace. This will assist the organisation in taking proactive action to forestall insider threat occurrences.

2. Related Works

In the last three decades, there are lots of research works on insider threats detection. Several algorithms and models of data mining and deep learning techniques have been proposed. In this section, the recorded successes and the deficiencies of the existing insider detection models in delivering the core mandate of protecting our assets in a cyberspace environment were critically analysed.

In [9], a comprehensive taxonomy to illustrate insider threats was derived. The taxonomy listed attributes such as access, privileges, motivation, tactics, knowledge, process, risks, and skill etc. The attributes are the well-known features being used to simulate malicious insider behaviour. Some general behavioural characteristics were also highlighted in [10].

The Intrusion Detective System (IDS) such as firewall logs, securely information and event management and data leak prevention system logs is a conventional method used to detect insider threats in an organization [11]. Analysis and investigation of users log using this method is usually time consuming and costly. Different variations of algorithms have been proposed to identify malicious users via cyber profiling, for example, Poisson-based algorithm [12], K-means and Kernel density estimation algorithm [13]. These algorithms were used to learn and analyse user behaviour and establish normal user profile based on behavioural data. Most of these algorithms are not persuasive enough and of limited extensibility.

Bayesian Networks-based human behaviour model was developed to detect insider threats [4], [14] - [16]. Bayesian networks models usually employed probabilities for the jugdements assessments for a known attack only and are not reflective of actual measured of user behavior. Also the Bayesian networks model only perform well on small data and may not perform optimally on a streaming data.

Graph-based framework is another technique that is used in network security management [17]-[18]. Graph-based model performed well in a static data, it cannot be extended to dynamic or evolving stream data. The Graph-based framework was used to analyse individual behaviour in [19], Social media data were analysed to detect insider threats premised on the sentiment level and negative emotion ratios. Sentiment analysis was conducted and users were classified according to a specified criteria to detect potential malicious insiders. The graphical analysis was able to detect malicious insider with an accuracy of 99.7%, but the system behaviour analysis was not taken into consideration.

Machine learning techniques has also been explored in various capacities look at patterns of users' behavior, and then analyzes them to detect anomalies that indicate potential insider threats in Cyberspace. Different algorithms like Decision Tree, K-means, OC-SVM [20], Support Vector Machine (SVM) [21]- [22], and Isolation Forest algorithm [23], However, machine learning approaches is only applicable to bounded-length, static data stream. It does not work well while handling high dimensional data.

In furtherance, ensemble machine learning approach was employed in insider threat detection. A user behaviour analytic model was developed using an ensemble approach platform to collect logs and extract features relating to potential insider threats [24]–[25]. Ensemble machine learning approach showed great performance in detecting user anomalous access and operation within an organization. However, it was noted that it does not take to account the dynamic nature of user behavior, and thus has limitation of not being able to detect unstable user behaviour.

The application of supervised machine learning yielded a good result in detecting insider threats but it is required that the system is constantly updated with new rules of attack. Supervised machine learning can be updated easily with new data and can learn non-linear relationships entities but lacks the natural flexibility to obtain quite complex patterns in streaming data. The unsupervised machine learning models are mostly static in nature and not able to handle dynamic data adequately by learning from large evolving data. The existing machine learning-based models have the challenges of not being able to learn from the sequence of user behaviour thus unable to detect unknown or temporal user behaviour pattern, therefore several abnormal user behaviour goes undetected, leading to a low detection rate and high

false alarm rate.

In order to accommodate the big, structured, and dynamic streaming data that usually emanate from system logs representing the user's behavior in cyberspace, Deep learning techniques were also explored to detect user's anomalous behaviour. Recurrent Neural Network (RNN) model [26], the Long Short Term Memory (LSTM) [27]-[28] and Convolution Neural Network (CNN) [27]. Deep learning algorithm gives more optimal, accurate and robust detection system [29], especially on large dimensional data with a considerable high degree of detection rate in the behavior defect, but the chances of occurrence of the defection was not considered.

However, in Cyberspace security management, a single algorithm is not sufficiently reliable as threats evolve. A single classifier may not adequately produce accurate prediction on a high dimensional data. Thus, an attempt is made in this study to employ a hybrid technique that will execute several approaches simultaneously and in real-time.to detect the insider threat in cyberspace.

3. Methodology

In this study, a hybrid approach comprising deep learning methods of Convolutional Neural Network (CNN) and Gated Recurrent Unit (GRU) was employed to determine if a user behaviour is likely to be malicious and then consequently determine a comprehensive threat prediction in the users log file dataset. An improvement was made on LSTM approach to threat detection proposed in [27]. The description of the proposed model concepts and the associated algorithms are presented as follows:

3.1 Description of the proposed model

The proposed model includes the roles of every user to the features in the dataset in order to achieve the role-based user behaviour threat detection techniques. As opposed to the feature engineering method applied in [27] for data processing which requires expertise and time consuming, the proposed model applied feature normalization and feature optimization for data processing to ensures that features on the dataset are on the same scale and also to reduce computation time as well as increasing the accuracy of the proposed model. The conceptual block diagram of the hybrid technique for insider threat detection is shown in Figure 1.

3.2 Data Pre-processing

Computer Emergency Response Team (CERT) insider dataset consisting of user logs, file access logs, decoy copy log was used as a data source. The dataset covers a period of three months which involves activity logs of 1000 users with insider threats in an organization. The logs of the user are logon/logoff, device usage and file access. The logs have details such as timestamps, system identification numbers, user identification numbers and user actions. The set of actions such as file access between 8.00 am – 5.00 pm was used to build the user profile based on user action sequence where logs outside the specified period are after-hour logs. Log lines of 3,015,990 were generated over the course of the period.

Dataset was extracted from the logon.csv, file.csv and device.csv files of the CERT Insider Threat Synthetic dataset v4.2. The logon.csv, file.csv and device.csv file from the CERT datasets has five columns consisting of id, date, user, pc and activity with a total of 335,111 instances. In this study there were five features in the dataset in which four features were in string character format; while only one feature was in date format. There are four main features in the dataset that were categorical variables and they were encoded into integers to ensure that all the features were in the same format using Algorithm 1. Features were scaled into a specific range using Min-Max normalization (Q) technique to scale the features from one range of values to a new range of values, this ensures that all features are in the same scale. This preserves exactly all relationship in the data after the normalization of the data. The dataset records used was a multi-dimensional dataset with features that are irrelevant to the proposed model's formulation. As a result, dimensionality reduction is required by choosing the best features from a large dataset. In Algorithm 2, the mutual information gain algorithm was used to choose the best features from the dataset based on the information gain.

Algorithm 1. Categorical Attributes Encoding Algorithm

```
Input: X(Dataset)
Output: X Identical Attributes of X
1: begin
2:  func textToNumericConverter(df):
3:  [row, column] = columnvalues(X)
4:      for each column in columns(X) do
5:      tex digit vals = {}
6:      func convertToInt(val):
7:          return tex digit vals[val]
8:      if col(type)6= (num) and col(type)6= (float): then
9:          df[col]. contents = df[col].values.tolist()
10:             uniqueelements = set(column contents)
11:         x=0
12:             for unique in unique elements do
13:             if unique not in tex digit vals then
14:                 tex digit vals[unique] = x
15:                     x +=1
16:             end if
17:             df[col] = list(map(convertTolnt, df[col]))
18:              end for
19:          end if
20:                 return df
21:             end for
22:         df = textToNumericConverter(df)
23: end
```

Algorithm 2. Mutual Information Gain Algorithm

Input: X(Sample dataset), T(Target variable)
Output: Selected Best Set (B) Features

I: **begin**

2: Input Sample dataset X which include Features F_i with Target class T

$$TotalIG = \sum_{i}^{N} IG(T, f_i)$$

4: **for** each feature f_i **do**

$$Information\ Gain\ (T, f_i) = \frac{IG(T, f_i)}{TotalIG}$$

6: **end for**
7: Sort Information Gain (T, f_i) in descending order
8: Put f_j, whose IG(T, f_j) > 0 into relevant feature Set R
9: Remove the remaining irrelevant features.
10: Input relevant feature set R
11: **for** each feature f_j **do**
12: Calculate Information Gain ($f_i f_j$)
13: **end for**
14: Select those features having IG($f_i f_j$) > T, with a well-defined threshold and put those features into set B
15: **end**

3.3 Model Formulation

The formulation of the model was based on the identification of the problem domain in cyberspace. The description of steps involved in developing the model presented in Algorithm 3 is as follows:

(i) A deep learning-based user behaviour analytic model that can detect attack based on anomalous user data in cyberspace was formulated using the synergy of Convolutional Neural Network (CNN) and Gated Recurrent Unit (GRU).

(a) **Establishment of CNN module**

The input parameters are the records of time sequence data (vector matrix) of combination PC access log, file access log and device copy of CERT dataset. The CNN modules use local connection and distribute weights to obtain local features directly from the user sequence data (vector matrix) and obtain accurate representation via convolution and pooling layers. After pooling operation, the vector matrix is condensed into 1-D data. The output is linked to the fully-connected layer.

$$X = \begin{bmatrix} X_1(1) & X_1(2) & \cdots & X_1(n) \\ X_2(1) & X_2(2) & \cdots & X_2(n) \\ \vdots & \vdots & \ddots & \vdots \\ X_k(1) & X_k(2) & \cdots & X_k(n) \end{bmatrix}, \quad (1)$$

where;

k denotes the k^{th} smart sensor,

n denotes the n^{th} time sequence, and

Xk(n) denotes the data captured by the k^{th} smart sensor at *n* time.

(b) Establishment of GRU module

The GRU captures the long term dependencies in order to learn from useful information in the vector matrix through memory cell and discard the unnecessary information using forget gate. The outputs are also linked to the fully-connected layer.

$$\begin{aligned} \Gamma_u &= \sigma\left(\omega_u\left[c^{\langle t-1\rangle}, x^{\langle t\rangle}\right] + b_u\right), \\ \Gamma_r &= \sigma\left(\omega_r\left[c^{\langle t-1\rangle}, x^{\langle t\rangle}\right] + b_r\right), \\ \tilde{c}^{\langle t\rangle} &= \tanh\left(\omega_c\left[\Gamma_r * c^{\langle t-1\rangle}, x^{\langle t\rangle}\right] + b_c\right), \\ c^{\langle t\rangle} &= \left(1 - \Gamma_u\right) * c^{\langle t-1\rangle} + \Gamma_u * \tilde{c}^{\langle t\rangle}, \end{aligned} \quad (2)$$

where;

bu, br, and bc are the vectors of the bias

ωc, ωu, ωr, and denote the training weight matrices of the candidate activation *c⟨t⟩,* update gate, and the reset gate, respectively.

The result is the average value neurons in the fully-connected layers which is the output prediction of the threat instances which is either '1' threat or '0' normal.

(ii) The new features were applied to the learning algorithm which then clustered the features and generated a model that classified the user logs instances as either normal or attacks as shown in Algorithm 4.

(iii) The new network instances were tested on the model and the resulting classification served as the outcome.

Algorithm 3: Model Formulation

1. Identify the relevant data files in the Computer Emergency Response Team (CERT) dataset
2. Use the log.csv, file.csv and device.csv file of CERT dataset as data source
3. Combine the three datasets together to form a single dataset in Excel sheet and save with .csv
4. Extracting necessary features from the dataset that follow the CERT Corporation Guidelines and International Journal standard format, using MS-Excel (Microsoft Excel) Sheet to determine targeting attack instance
5. Loading of the combined dataset into the data frame in Python using Pandas
6. Analysing the dataset in the data frame using Pandas and Numpy to get all the columns available in the dataset
7. Encoding of categorical attributes into numerical attributes using the categorical attribute encoding algorithm.
8. Normalization of each vector using Min-Max normalization

$$Q = \left(\frac{P - Min(p)}{Max(P) - Min(P)}\right) * (N - M) + M$$

Where the value of P feature needs to be normalized into Q. *Min(P)* and *Max(P)* is the minimum and maximum values of feature P respectively. M and N indicates Lower and Upper values respectively in the new range. (0, 1) is used to normalize the features of P, this makes Q to be in the range 0 and 1

9. Calculate the relevance of each feature to the label feature using Entropy Based (Mutual Information Gain) feature selection as indicated below:

$$\mathbf{H(X) = -\Sigma\,(p_i * \log_2 p_i)}$$

$$Entropy\ (p) = -\sum_{i=1}^{N} p_i \log_2 p_i$$

10. Select Best feature with highest information gain based on result from (9)
11. Divide the dataset into training and testing in ratio 60%:40%, 60% for training and 40% for testing the model.
12. Building the model by applying Hybrid (CNN/GRU) algorithms on the dataset.
13. Application of smaller part of the dataset divided for testing the insider threat model, labelling the normal as 0 and threat as 1.

Algorithm 4. Classifying the Dataset

Input: Test-points (M_i, i = 1… n), hybrid model

Output: Classified Test dataset

1. **begin**
2. Input Test-points into Hybrid(CNN/GRU) approximate Predict
3. **For each** Test-point M_i **do**
4. Classify Test dataset class as threat or normal
5. **end for**
6. Return Classified Test dataset from the hybrid (CNN/GRU model
7. **end**

4. Results and Discussions

The developed model was simulated and evaluated in Anaconda with Python version 3.6.3 environment. The Google Collaboration Laboratory Jupyter notebook served as the integrated Development Environment (IDE). A hybrid library tool that included Gated Recurrent Units (GRU) and Convolution Neural Networks (CNN) was utilised to simulate the developed model. The dataset was analyzed and the model's performance was evaluated using the *Scikit-learn* library and Python machine learning tools. The detailed results are presented as follows.

4.1 Analysis of the Dataset

The Computer Emergency and Respond Team (CERT) Insider Threat V4.2 dataset has a total of 335,110 datapoints which contains the sparse distribution of insider threats. Among these threats are unauthorized file access, indiscriminate decoy copy of documents and after-hour logon to workstations. The dataset statistics is presented in Table 1. The PC access logs has 167,598 datapoints which

represent 51.01%, file access logs has 87,759 which represent 26.18% and device copy logs has 79,758 which represent 23.80%. The *sklearn.preprocessing.Label* encoder library was used to encode the categorical features such as log id, user id, pc id and role in the dataset. The encoded format on the enumeration of user action is presented in Table 2. The dataset was normalized using standard min-max scalar to place the features on the same scale as shown in Figure 2. The features in the dataset are which represent 0 – 9 respectively in the mutual gain selection technique. Table 3 shows the result of mutual gain selection technique with respect to the performance of each feature in the model. There are only nine features in the dataset; id, user, role, pc, *date_hour*, *date_minute*, *date_day*, *date_month*, *activity* and *status*. It was shown that the id column has '0' value and thus has no contribution to the study. The new selected dataset is presented in Figure 3. The normalized dataset was divided into 60:40 ratios as training and test samples, using the *train_test_split* function of Python Programming Language. The training dataset contains 201,066 datapoints which represent 60% while the testing dataset has 134,044 datapoints which represent 40%. The dataset has 295, 163 normal instances and 39,947 threat instances. An output prediction of '0' normal or '1' threat was trained using a deep learning-based user behavior analytic hybrid (CNN/GRU) model.

Table 1. Statistical distribution of datasets

Datasets	No. of datapoints	Representation (%)
PC Access Logs	167,598	51.01
File Access Logs	87,759	26.18
Device Copy Logs	79,758	23.80

Table 3. Mutual Information Gain Selection Result

S/N	Features	Scores	Ranks
0	Id	0	9th
1	User	0.11450904	2nd
2	Role	0.06101811	3rd
3	Pc	0.05735364	4th
4	Date_hour	0.22341095	1st
5	Date_minute	0.00269378	7th
6	Date_day	0.00169675	8th
7	Date_month	0.01638053	6th
8	activity	0.02942816	5th

	0	1	2	3	4	5	6	7
0	0	22	894	8	9	4	1	1
1	0	22	894	18	20	4	1	2
2	0	22	894	8	19	5	1	1
3	0	22	894	18	11	5	1	2
4	0	22	894	8	9	6	1	1
....	....	....	....	....	....	....	.....	....
335105	678	17	591	22	11	31	3	4
335106	678	17	591	22	11	31	3	4
335107	300	17	738	22	52	31	3	4
335108	300	17	738	22	52	31	3	5
335109	300	17	738	22	52	31	3	8

Figure 3. Selected Dataset

4.2 Model Simulation Results

The developed model was simulated using the training and testing dataset on the CNN-GRU hybrid algorithm which clustered and classified the dataset into normal and attack instances, label as 1 and 0. The result of the classified clusters was stored in the cluster. Tables 4 and 5 showed the results of the developed and the selected existing model, LSTM model. The table shows the True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values where;

- True Positive (TP) implies when an anomaly or attacks is correctly predicted as anomalous or attacks
- True Negative (TN) implies when a normal instance is correctly predicted normal activity.
- False Positive (FP) implies when a normal instance is wrongly predicted as a threat.
- False Negative (FN) implies when anomaly is wrongly predicted as normal activity.

Table 4 demonstrated that the proposed model correctly classified 2,188 as threat instances (TP) while 128361instances were correctly classified as normal instances (TN). The result shows that 3,456 instances was misclassified as threat instances (FP) while 39 instances were misclassified as normal instances (FN). Table 5 demonstrated that the existing model correctly classified 479 instances as threat instances (TP) while 128,090 instances were correctly classified as normal instances (TN). The result also shows that 5,165 were misclassified as threat instances (FP) while 310 instances were misclassified as normal instances (FN). The models of the two components of GRU-CNN hybrid network were also simulated separately to know the true picture of each model. The confusion matrix result for individual model is presented in Table 6.

Table 4. Confusion Matrix of the Proposed Model

	Predicted Negative	Predicted Positive
Actual Negative	128361	39
Actual Positive	3456	2188

Table 5. Confusion Matrix of the Existing Model

	Predicted Negative	Predicted Positive
Actual Negative	128090	310
Actual Positive	5165	479

Table 6. Confusion Matrix of the Models Component

Model	True Positive (TP)	True Negative (TN)	False Positive (FP)	False Negative (FN)
CNN	1500	1283880	4144	20
GRU	486	128084	5158	316
LSTM	479	128090	5165	310
GRU-CNN	2188	128361	3456	39

4.3 Model Evaluation Results

The performances of the existing and developed models were evaluated using accuracy, precision, sensitivity, Receiver Operative Curve (ROC), and Model loss defined as follows:

(a) Accuracy measures how precisely and effectively the model can detect normal or attack instances. This is calculated as follows.

$$Accuracy = (TP+TN/TP+TN+FN+FP) * 100 \quad (3)$$

(b) Precision shows how many instances were correctly identified in the positively identified set. This is calculated as follows

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

(c) Sensitivity shows the number of positive instances captured by the predicted positive instances. This is calculated as follows:

$$Sensitivity = (TP/TP+FN) * 100 \quad (5)$$

(d) Receiver Operative Curve (ROC) shows the performance of the classifiers without regard to the class distribution.

(e) Loss means prediction error.

The performance metric evaluation results in Table 7 and Figure 4 showed that the proposed model has an increased detection accuracy rate of 1.48%, sensitivity rate of 1.25% and a precision rate of 4.21% over the existing model. This showed that the performance of the proposed model outperformed the existing model.

Table 7. Evaluation Results of the Component Models

Model	Accuracy (%)	Sensitivity (%)	Precision (%)
CNN	96.89	96.87	99.98
GRU	95.92	96.12	99.75
LSTM	95.91	96.12	95.75
GRU-CNN	97.39	97.37	99.96

The Receiver Operating Characteristic Curve (ROC) is presented in Figure 5. The Area Under Curve (AUC) of the proposed model 0.90 which is a good performance in the insider threat model as compared to the LSTM model of 0.54. The extended ROC result for CNN and GRU showed that CNN has 0.89 and GRU has 0.537 as indicated in Table 8. The ROC Curve for the proposed model with a value 0.90 is well above 45 degrees and tends to 1 on the True Positive rate -axis and LSTM, GRU and CNN models with value 0.54, 0.537 and 0.90 respectively are below the Hybrid model getting close to 45 degree. This means Hybrid model has more capacity and higher proportion of actual prediction (True positive) of insider threats than the LSTM model.

As indicated in Figure 5, the proposed model tends to 1 on the accuracy axis where LSTM, GRU and CNN model are below the proposed model on the same axis. This means that the proposed model has more accurate prediction capacity than the other models.

The proposed model loss as indicated in Figure 6 has 0.12 which tends to 0 on the loss -axis where LSTM, GRU and CNN have 0.17, 0.17 and 0.15 respectively are above the proposed model on the loss-axis. As indicated in the Table 9, it showed that the proposed model has the lowest prediction error.

Table 8. ROC Result of the Component Models

Model	ROC Value (3d.p)
CNN	0.890
GRU	0.537
LSTM	0.540
GRU-CNN	0.900

Table 9. Model Loss for the Component Models

Model	Loss
CNN	0.17
GRU	0.17
LSTM	0.15
GRU-CNN	0.12

5. Conclusions

This work focused on the detection of insider threats in cyberspace using user behaviour analytics. The existing models have the challenges of not being able to detect unknown or temporal user behaviour pattern, therefore several abnormal user behaviour goes undetected. Furthermore, most existing models could not learn from the sequence of user behaviour thus leading to a low detection rate and high false alarm rate. In this research, a hybrid technique for the insider threat detection model using a deep learning approach to increase the detection of insider threats in cyberspace was developed. The simulation result indicates that the proposed model can detect more insider threats; and has higher detection accuracy, precision, sensitivity rate than the existing single LSTM, GRU and CNN models. It also has the capacity to learn a sequence of user data leading to a high detection rate and reducing false alarm rate.

6. Acknowledgement

This Research was funded by the TETFund Research Fund" and Africa Centre of Excellence OAK-Park.

References

[1] B.O., Akinyemi, A. O., Amoo, E. A. Olajubu, "An Adaptive Decision Support Model for Data Communication Network Security Risk Management," International Journal of Computer Applications, Vol. 106, No. 8, pp. 1-7, 2014.

[2] B.O., Akinyemi, A.O., Amoo, Aderounmu G.A., "Performance Prediction Model for Network Security Risk Management," Communications on Applied Electronics (CAE), Vol.2, No.8, pp.1-7, 2015. .Doi: 10.5120/cae2015651816.

[3] E. E. Schultz, "A framework for understanding and predicting insider attacks. Computers and security, Vol. 21. No.6, pp.526-531, 2002. Doi: 10.1016/S0167-4048(02)01009-X

[4] F.L Greitzer, R.E, Hohimer, "Modeling Human Behavior to Anticipate Insider Attacks," Journal of Strategic Security, Vol. 4, No.2, pp.25–48, 2011. Doi: 10.5038/1944-0472.4.2.2

[5] S. J., Stolfo, S. M., Bellovin, A. D., Keromytis, S., Hershkop, S. W, Smith, S.Sinclair, "Insider attack and cyber security: beyond the hacker," Advances in Information Security, Vol. 39, 2008. Doi: 10.1007/978-0-387-77322-3

[6] R. A., Caralli, J. H., Allen,. P. D., Curtis, D. W., White, L. R. Young, "Improving Operational Resilience Processes: The CERT Resilience Management Model", in proceedings of the 2010 IEEE Second International Conference on Social Computing (SocialCom), Minneapolis, MN, US, pp. 1165-1170, 2010. Doi: 10.1109/SocialCom.2010.173.

[7] A. G., Pramanik, V., Singh, R., Vig, A. K. Srivastava, D. N. Tiwary, "Estimation of effective porosity using geostatistics and multiattribute transforms: A case study," Geophysics, Vol.69, No. 2, pp.352-372, 2004. Doi: 10.1190/1.1707054

[8] A., Lazarevic, L., Ertoz, V., Kumar, A., Ozgur, J. Srivastava, "A comparative study of anomaly detection schemes in network intrusion detection," In Proceedings of the 2003 SIAM international conference on data mining, San Francisco, CA, USA, pp. 25-36, 2003. Doi: 10.1137/1.9781611972733.3

[9] B. Wood, "An insider threat model for adversary simulation. SRI International," Research on Mitigating the Insider Threat to Information Systems, pp. 1-3, 2000.

[10] A. I., Schoenholtz, P. G. Schrag, J. Ramji-Nogales, "Lives in the balance: Asylum adjudication by the department of homeland security." NYU Press, 2014.

[11] A., Shabtai, Y. Elovici, L. Rokach, "A survey of data leakage detection and prevention solutions, "Springer Science and Business Media. 2012, ISBN: 978-1-4614-2052-1

[12] A. Sapegin, A. Amirkhanyan, M. Gawron, F. Cheng, C. Meinel, "Poisson-Based Anomaly Detection for Identifying Malicious User Behaviour." In: Boumerdassi S., Bouzefrane S., Renault É. (eds) Mobile, Secure, and Programmable Networking. MSPN 2015. Lecture Notes in Computer Science, Vol. 9395, 2015, Springer, Cham. doi:10.1007/978-3-319-25744-0_12

[13] A. W., Udoeyop, "Cyber Profiling for Insider Threat Detection." Master's Thesis, University of Tennessee, 2010.http://trace.tennessee.edu/utk_gradthes/756

[14] G., Alghamdi, K., Laskey, X., Wang, D., Barbara, T., Shackelford, E.Wright, J. Fitzgerald, "Detecting threatening behavior using bayesian networks," In Proceedings of the Conference on Behavioral Representation in Modeling and Simulation, Arlington, Virginia, pp. 32-33, 2004.

[15] S. McKinney, D. S. Reeves, "User identification via process profiling," In Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Challenges and Strategies, Oak Ridge Tennessee USA, pp. 1-4, 2009. .Doi:10.1145/1558607.1558666

[16] H., Wang, S., Liu, X., Zhang, "A prediction model of insider threat based on multi-agent." In 2006 First International Symposium on Pervasive Computing and Applications, Urumqi, China, pp.273-278, 2006. Doi: 10.1109/SPCA.2006.297582

[17] S., Staniford-Chen, S., Cheung, R., Crawford, M., Dilger, J., Frank, J., Hoagland, D., Zerkle, "GrIDS-a graph based intrusion detection system for large networks". In Proceedings of the 19th national information systems security conference, Baltimore, Maryland, pp.361-370, 1996.

[18] B.O; Akinyemi, O.V; Jekoyemi, T.A; Aladesanmi G.A; Aderounmu, B.H Kamagate , "A Scalable Attack Graph Generation for Network Security Management." Journal of Computer Science and Information Technology (JCSIT), Vol.6, No.2, pp.30-44, 2018. Doi:10.15640/jcsit.v6n2a4.

[19] A., Gamachchi, L., Sun, L. Boztas, "Graph based framework for malicious insider threat detection." In Proceedings of the 50th Hawaii International Conference of System Science, Hilton Waikoloa Village, Hawaii, USA, pp. 2638–2647, 2017. Doi: 10.24251/HICSS.2017.319

[20] B. K., Szymanski, Y. Zhang, "Recursive data mining for masquerade detection and author identification". In Proceedings of the Fifth Annual IEEE SMC Information Assurance Workshop, West Point, New York, USA, pp. 424-431, 2004. Doi: 10.1109/IAW.2004.1437848

[21] P., Parveen, N., Mcdaniel, Z., Weger, J., Evans, B., Thuraisingham, Hamlen, K. L. Khan, "Evolving insider threat detection stream mining perspective." International Journal on Artificial Intelligence Tools, Vol.22, No.05, 1360013, 2013. Doi: 10.1142/S0218213013600130

[22] G., Gavai, K., Sricharan, D., Gunning, J., Hanley, M., Singhal, R., Rolleston, "Supervised and unsupervised methods to detect insider threat from enterprise social and online activity data." Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications Vol.6, No.4, pp.47–63, 2015.

[23] L., Sun, S., Versteeg, S., Boztas, A., Rao, "Detecting anomalous user behavior using an extended isolation forest algorithm: an enterprise case study." arXiv:1609.06676, 2016.

[24] W., Ma, K., Sartipi, D., Bender, "Knowledge-driven user behavior pattern discovery for system security enhancement." International Journal of Software Engineering and Knowledge Engineering, Vol.26, No.03, pp.379-404, 2016. Doi: 10.1142/S0218194016500169

[25] X., Xi, T., Zhang, D., Du, G., Zhao, Q., Gao, W. Zhao, S. Zhang, "Method and System for Detecting Anomalous User Behaviors: An Ensemble Approach." In proceedings of the 30th International Conference on Software Engineering and Knowledge Engineering (SEKE), San Francisco, USA, pp. 263-262, 2018. Doi: 10.18293/SEKE2018-036

[26] Tuor, A., Kaplan, S., Hutchinson, B., Nichols, N. and Robinson, S. (2017). Deep learning for unsupervised insider threat detection in structured cybersecurity data streams. In Workshops at the Thirty-First AAAI Conference on Artificial Intelligence. arXiv:1710.00811

[27] Yuan F., Cao Y., Shang Y., Liu Y., Tan J., Fang B. (2018) Insider Threat Detection with Deep Neural Network. In: Shi Y. et al. (eds) Computational Science – ICCS 2018. ICCS 2018. Lecture Notes in Computer Science, Vol 10860. Doi: 10.1007/978-3-319-93698-7_4

[28] Matterer J. and Lejeune D. (2018). Peer group metadata-informed LSTM ensembles for insider threat detection. In proceedings of the International Florida Artificial Intelligence Research Society Conference, pp. 62–67.

[29] M. Al-Shabi, "Design of a Network Intrusion Detection System Using Complex Deep Neuronal Networks." International Journal of Communication Networks and Information Security (IJCNIS), Vol. 13, No. 3, 2021. doi:https://doi.org/10.54039/ijcnis.v13i3.5148

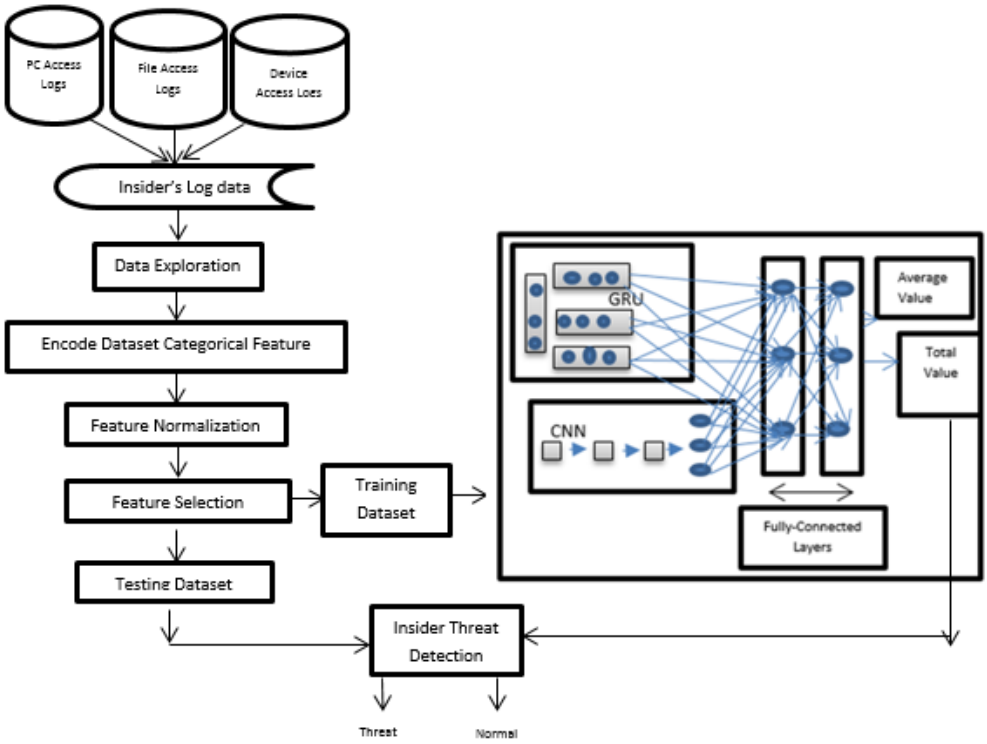

Figure 1. Conceptual Framework for a predictive model on insider threat in cyberspace environment.

Table 2: Encoded user actions

Time	Computer	Activities	Action	ID	Description
In hour action (8am-5pm) or after hour action (5pm-8am)	On assigned PC or unassigned PC	Logon/Logoff activity	Logon	0	User log on a computer
			Logoff	1	User logoff on a computer
		File activity	Copy *.exe* files	2	*.exe* file access or copy to a removable media
			Copy *.doc* files	3	*.doc* file access or copy to a removable media
			Copy *.pdf* files	4	*.pdf* file access or copy to a removable media
			Copy *.txt* files	5	*.txt* file access or copy to a removable media
			Copy *.jpg* files	6	*.jpg* file access or copy to a removable media
			Copy *.zip* files	7	*.zip* file access or copy to a removable media
		Device Activity	Connect	8	User inserted removable media device
			Disconnect	9	User removed removable media device

	0	1	2	3	4	5	6	7	8
0	- 0.846742	1.226732	-1.606201	-0.080583	-0.187193	0.899181	1.169832	1.742794	-0.362246
1	-0.877985	0.675491	-0.247002	-0.080583	-0.746559	-0.620299	-0.040836	-0.062700	-0.36224622
2	0.930674	-2.264464	1.132896	-1.238434	1.602777	0.197883	0.169832	-0.965447	-0.362246
3	1.229224	0.583617	1.281235	1.077268	-0.914368	1.600480	-1.251504	-0.062700	-0.362246
4	-0.610679	0.675491	-0.888654	-0.080583	1.323094	1.834246	1.169832	-0.062700	-0.362246
….	….	….	….	….	….	….	…..	….	….
201061	-1.159178	0.859238	0.284259	1.540409	-1.473734	0.782296	1.69832	-0.664532	2.66612
201062	-1.263323	-1.253855	1.709003	-0.775294	-1.585607	0.8991811	1.169832	-0.965447	-0.362246
201063	-1.291095	0.216123	-0.471235	-1.238434	1.490904	-1.321598	1.169832	-0.965447	-0.362246
201064	1.152851	0.859238	0.584387	0.845698	-0.355003	-0.854065	-1.251504	0.238216	-0.362246
201065	-0.051765	0.583617	-1.185332	-1.238434	1.267158	0.314766	-1.251504	-0.965447	-0.362246

Figure 2. Normalized data

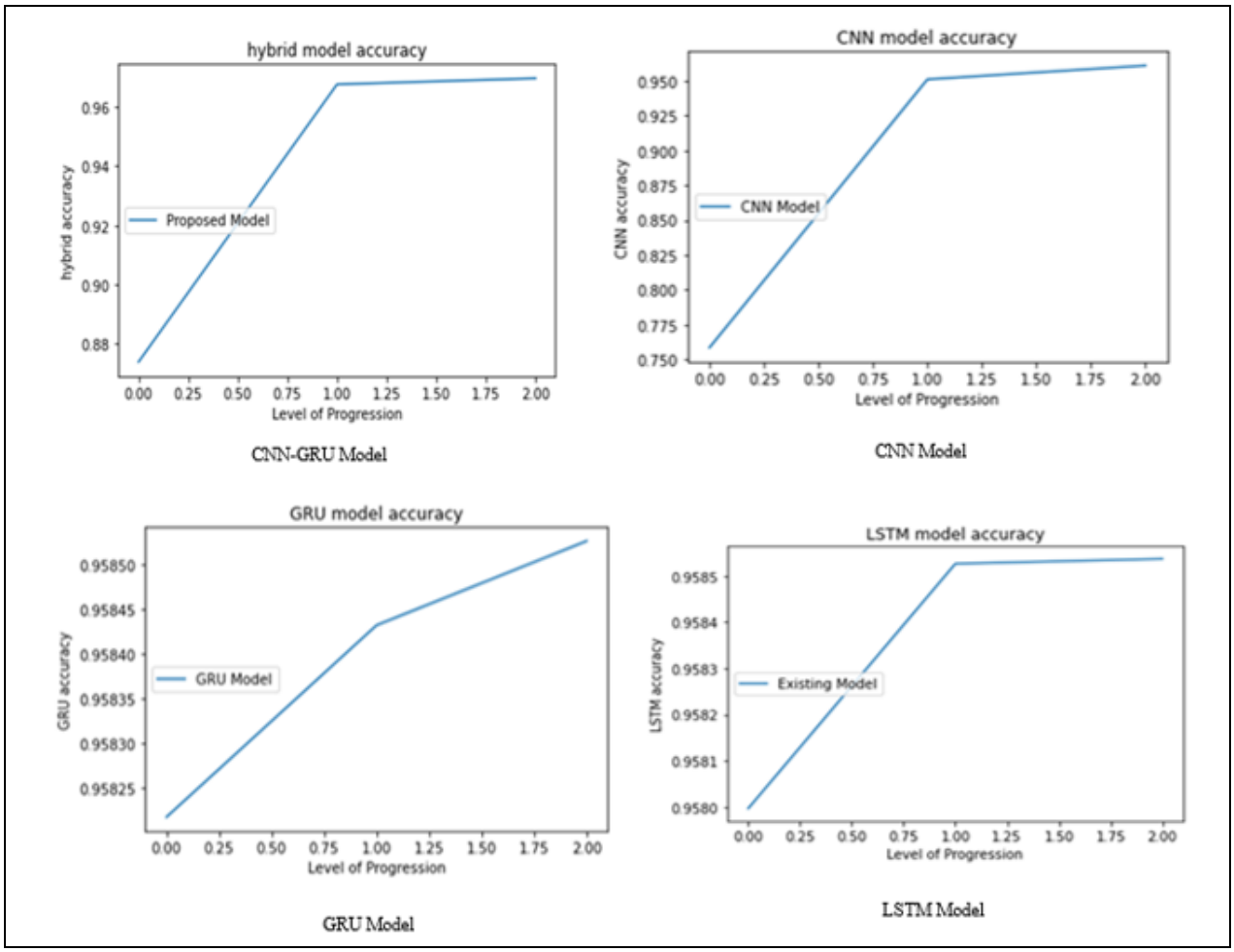

Figure 4. Accuracy graphs

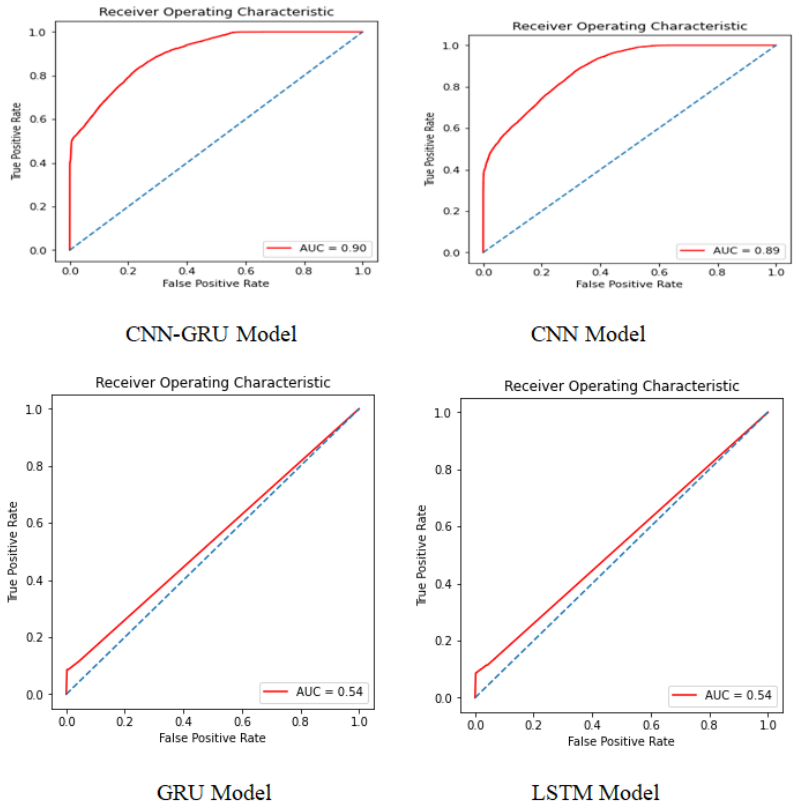

Figure 5. ROC Graphs for the Models

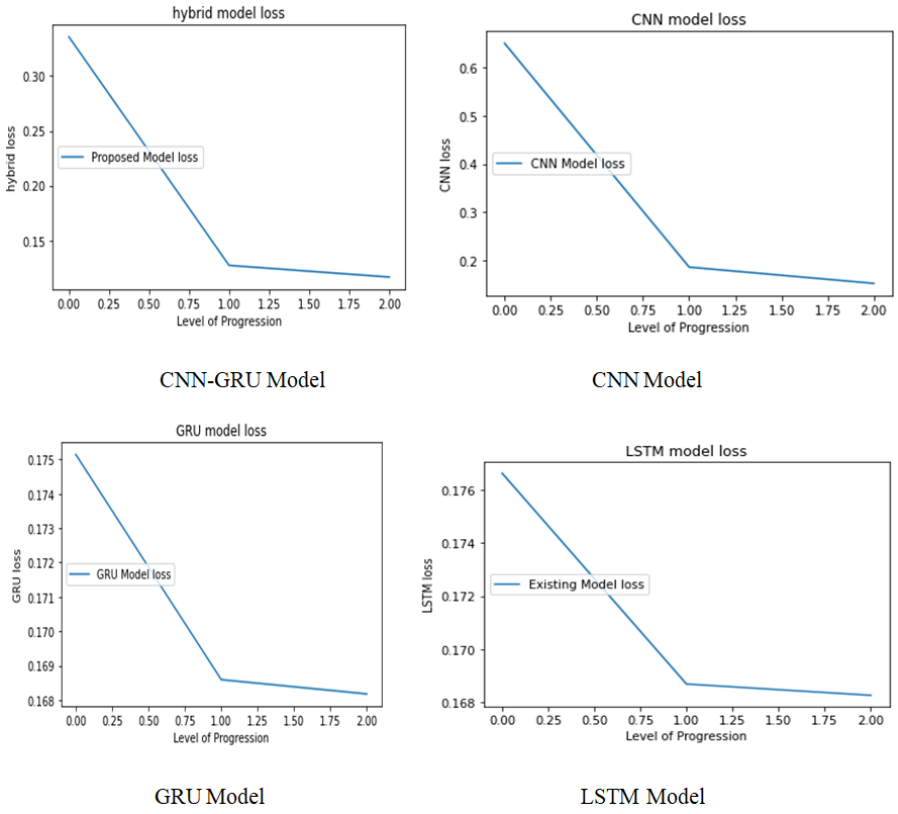

Figure 6. Loss Graphs for the Models