International Journal of Computer Applications Technology and Research
Volume 13–Issue 10, 52 – 61, 2024, ISSN:-2319–8656
DOI:10.7753/IJCATR1310.1004

A Theory-Based Deep Learning Approach for Insider Threat Detection and Classification

Everleen Nekesa Wanyonyi
Jaramogi Oginga Odinga University of Science and Technology, Bondo, Kenya

Newton Wafula Masinde
Jaramogi Oginga Odinga University of Science and Technology, Bondo, Kenya

Silvance Onyango Abeka
Jaramogi Oginga Odinga University of Science and Technology, Bondo, Kenya

Abstract: Insider threats are a substantial concern to organizational security, often leading to grave financial and reputational damage. Classical insider threat detection methods rely on predefined rules and signatures and struggle to keep pace with these attacks' sophisticated and evolving nature leading to dismal performances. This research introduces a deep learning-based approach for insider threat detection, leveraging user network behavior as the primary data source. Our technology detects deviations in user network activity that might indicate harmful insider activities. We use a Gated Recurrent Network (GRU) that captures user behavior's temporal and spatial characteristics. The proposed model is validated using a synthetic CERT r4.2 dataset and exhibits higher detection rates based on accuracy, Recall, Precision, and f-measure. Additionally, the Social Bond Theory (SBT) and the Situational Crime Prevention Theory (SCPT) are used to elaborate effective ways to control insider threats. This study also presents solutions for dataset imbalance and high dimensionality that adversely hinder common insider threat datasets from giving accurate predictions during model training and validation. Our findings show that deep learning and data preprocessing approaches can considerably improve the ability to detect insider threats, giving organizations a reliable defense mechanism against insider threats.

Keywords: Insider; threat detection; theory-based; information security; deep learning; Gated Recurrent Unit; network behavior

1. INTRODUCTION

As a result of enterprises switching to remote working during the pandemic, insider threats have recently increased globally (Griffiths, 2024). Insider threat actors have benefited from misaligned networks, leading to a high increment of 358% over the previous years. By 2021, insider threats had risen by 125% globally; to date, most enterprises and individuals are threatened. Industry research reveals that insider threats, or harmful actions committed by unsatisfied workers who abuse their authorized access to networks, systems, and data, account for 79% of security threats (Bin Sarhan & Altwaijry, 2022). This has led to the cost of insider attacks increasing by 31% globally, reaching $11.45 million (Saxena et al., 2020).

The elusive nature of insider threats has made it difficult for classical techniques to control them. For example, firewalls, IDS, and IPS focus more on the outsider because the insider possesses authorized access, is a trusted entity, and fully knows how systems operate and their locations (Saxena et al., 2020). Other controls, such as signature-based systems, rely on storing past attacks, which suffer when encountering zero-day attacks. They also need large storage spaces and expertise to update the databases (CISA, 2024). Machine Learning (ML) models rely heavily on feature engineering and struggle to accurately distinguish between insider and normal user behavior due to data characteristics such as complexity, heterogeneity, sparsity, lack of labeled insider threats, and hidden and adaptable threats (Yuan & Wu, 2020a).

DL techniques have been proposed as practical solutions to insider threats (Yuan & Wu, 2021). In DL, multiple hidden layers are organized in deeply nested network architectures with advanced neurons that enhance detection and classification activities (Janiesch et al., 2021). DL technology is gaining popularity due to its efficiency in working with large heterogeneous datasets and combining several layers, such as input, hidden, and output, to improve performance (Al-Shehari & Alsowail, 2023; Alsowail et al., 2022). Although DL models outperform classical and ML insider threat detection models, they struggle to detect insider threats (Yuan & Wu, 2020). Despite the significant advancement and substantial work on DL technology for insider threat detection, there are still numerous chances to advance and improve the existing models into state-of-the-art systems for insider threat detection and prevention (Le & Zincir-Heywood, 2019). This is because the existing models still face challenges with imbalanced and highly dimensional datasets. In addition, poorly validated DL models have also exhibited poor detection rates (Tuor et al., 2017).

This research proposes an insider threat detection and classification model that integrates the Gated Recurrent Unit (GRU), SMOTE, and Adaptive Moment Estimation (Adam) algorithms for detection, data imbalance correction, and model training optimization respectively. The model is evaluated on four metrics using the popular CERT r4.2 dataset containing synthetic user network behavioral characteristics. Data pre-processing and feature engineering techniques are performed to enhance the data quality before model training and validation. The study recommends a layered approach to insider threat mitigation by introducing theoretical explanations of controlling insider threats within organizations. The Social Bond Theory (SBT) and the Situational Crime Prevention Theory (SCPT) have been utilized to illustrate the factors that prevent people from engaging in crime and hardening systems to reduce opportunity and motivation respectively. Practical solutions for SCPT may include the combined security policy approach, logging and monitoring, conducting periodic vulnerability assessments, and actively safeguarding information infrastructure from insider threats (Dawson & Omar, 2015).

The DL-based insider threat detection and classification model validation results indicate higher performance on the metrics compared to the Vanilla RNN, DNN, and LSTM. These results show that data preprocessing is a key step in improving DL models' performance. The study faced challenges with model training resources because of the big data used for training and validation. This study made the following contributions:

1. Advances a more accurate proactive tool for monitoring user network behavior to detect threats.
2. Catalyzes multidisciplinary research by integrating concepts from computer science, psychology, sociology, economics, and law to control insider threats.

3. Enhances the defense-in-depth strategy to encompass internal threats to improve the theoretical basis of comprehensive security models.

2. LITERATURE REVIEW

Research on insider threats attracts interest from numerous government entities, cybersecurity companies, and individuals. This is due to the damaging effects malicious employees cause on organizational computer networks and the difficulty distinguishing malicious from insiders' benign activities (Le & Zincir-Heywood, 2019). In 2006, the American Institute of Computer Security (CSI) reported that insider threats, such as malicious abuse of authority, pose a more significant threat to enterprises than classic attacks, such as Trojans (CERT, 2014). These factors make insider threats more dangerous to organizations' business continuity, requiring proactive security techniques to evade them. Motivations for insider threats are indicated in Table 1.

Table 1. Motivations for insider threats (Author, 2024)

Motivating Factor	Reason	Example
Financial gain (Kont et al., 2021); Personal gain (SEI, 2022).	Inadequate payouts	Greedy employee sells restricted information to competitors.
Revenge (Kanellopoulos, 2024)	Unfair treatment/grudge against a colleague	Disgruntled employee deletes organizational data
Political/ideological (CyberArk, 2017)	Having different ideologies from others	Hacking to destroy information or disrupt production
Desire to please/show off (Kont et al., 2021)	Pride	Hack and destroy systems to show capability to peers
Anger (CyberArk, 2017)	Feeling betrayed/unmet expectations	Delete databases to hurt those in charge
Depression and anxiety (Nurse et al., 2014)	Divorce/stress/sickness	Delete and disrupt processes to feel better

2.1 Insider Threats to Information Systems

Insider threats are currently one of the biggest concerns for intranets, as they can cause system failure, data exfiltration, and information loss (Hu et al., 2019). They are caused by perpetrators with authorized access who have knowledge of underlying sensitive systems and are trusted by the organization. They are also aware of the organization's safety facilities' regulations, such as firewalls and IDS, and can easily avoid them (Kanellopoulos, 2024).

Insider threats have three main features: transparency, concealment, and high risks. Identifying insider threats is more challenging because insiders are acquainted with the organization's information system and can readily avoid surveillance systems. Furthermore, fraudulent activities by insiders are frequently disguised as a wide range of legitimate actions, making detection difficult (Jiang et al., 2018b). Moreover, most insiders are employees who deal with critical assets for their daily assignments. As a result, the harm is significant compared to that caused by exterior attacks (Alsowail & Al-Shehari, 2022).

Insider threats can be grouped into five main profiles which are discussed in the following.

2.1.1 IT Sabotage

Such incidents are highly sophisticated and are majorly committed by insiders with sophisticated IT skills, privileged access to systems or networks, and knowledge of how they are configured (Saxena et al., 2021). These attacks range from malware, worm, or Trojan insertion to tampering and disruption of information resources. The attacker intentionally uses technical methods to disrupt or cease normal business operations. Approximately 90% of perpetrators are system administrators with a motive of harming the organization or a specific person (Nurse et al., 2019).

2.1.2 Intellectual property (IP) theft:

Crimes against IP are committed by employees who directly work with or are in charge of the same information they are supposed to protect. IP includes valuable company data, trade secrets, programming code, and customer information. 75% of IP thefts are performed by technical staff who use file transfers, remote access, and emails to violate security against product information, source code, and proprietary software (Nurse et al., 2019).

2.1.3 Insider Fraud

This is the most frequent attack within the IT environment, with more than 61% of managers rating it as the most prevalent insider threat. Fraud can range from stealing organization funds to trading in organizational data for personal gain (Nurse et al., 2019). In 2018, all companies hit by fraud indicated an insider as a perpetrator and financial gain as the primary motivating factor (Saxena et al., 2021).

2.1.4 Espionage

IT espionage, also known as cyber espionage, is a form of IP theft that involves obtaining personal, sensitive, or proprietary information from individuals without their knowledge or consent (Nurse et al., 2019). This attack can be committed by technical and non-technical staff who act on behalf of the "employer." This second employer may be a competitor organization or sometimes for their gain (Freet & Agrawal, 2017).

2.1.5 Unintentional insider

An accidental insider is an employee, contractor, or business partner who has authorized access to an organization's network, system, or data and who acts without malicious intent and unwittingly causes harm or substantially increases the probability of severe future harm to the CIA of the organization's information system resources (Khan's et al., 2021). Common attacks include the loss of laptops and auxiliary storage devices and careless e-mail and web browsing practices that lead to the downloading of worms and Trojans. It is noted that unintentional attacks occur more frequently than their malicious counterparts (Saxena et al., 2020).

2.2 Insider Threat Mitigation

Mitigating insider threats requires a complex, diverse, and comprehensive approach due to the variety of threat sources and motivations (Singh et al., 2023). Many organizations focus on external attacks when designing their network security while overlooking insider threats which tend to cause more severe damage due to the secrecy and concealment of user activities (Alsowail & Al-Shehari, 2022). The main concern then lies in identifying which authorized users are attacking or planning to attack the organization due to the elusive nature of these threats (Saxena et al., 2020).

Traditional security controls primarily focus on external threats, making it easier for insiders familiar with the organization to elude detection (Kont et al., 2021). Honeypots, decoy machines designed to fool an attacker, are one method of identifying insider attackers. However, as security awareness grows, insider attackers adopt more subtle methods to perpetrate the attacks, which calls for more advanced detection and protection strategies (Legg et al., 2017). Signature-based techniques, compare user actions against a database of known attacks to detect deviations (Kong & Bashir, 2022). This technique often leads to high false positives when encountering new or benign user activity. In addition, maintaining a database of past attacks requires significant storage resources (Wei et al., 2021).

Anomaly-based Intrusion Detection Systems (IDS) work as a behavior-based model, assuming that a user's current activity closely resembles their previous and next action sequence (Aldairi et al., 2019). The systems create user behavior profiles from the user activity sequences that serve as a checkpoint in detecting anomalies (T. et al., 2024). Currently, these methods leverage ML technology, utilizing user network behavior to identify inconsistencies and detect anomalies (Nicolaou et al., 2020). In ML, a computer "learns" an algorithm to determine the most relevant performance criteria from training data to complete assigned tasks (Jiang et al., 2022). Nevertheless, these models struggle to handle Big Data from fast-growing networks and rely on linear models, which perform poorly with complex and heterogeneous data (Saxena et al., 2020).

Recently, deep learning (DL), a subset of ML, has gained importance in its use due to its ability to learn and extract complex patterns from massive volumes of data. DL offers a new framework for developing sophisticated models from intricate datasets (Al-Mhiqani et al., 2021). DL models make use of a multi-layer architecture to acquire knowledge of data representation, with the lower layers capturing low-level data characteristics. In contrast, the upper layers extract high-level abstract information which improves anomaly detection (Yuan & Wu, 2020). Despite these advancements, DL models face various challenges due to common anomaly detection data characteristics like high dimensionality, complexity, heterogeneity, sparsity, absence of labeled data, and insider threats' nuanced and adaptive nature (Yuan & Wu, 2020). To compensate for the weaknesses of the two methods, hybrid models that combine signature-based and anomaly-based characteristics have emerged. Table 2 presents common insider threat mitigation strategies' characteristics, strengths, and weaknesses.

Table 2. Features, strengths, and weaknesses of insider threat detection models (Author, 2024)

Algorithm	Characteristics	Strengths	Weaknesses
Signature-Based Detection Models	- Need for domain expert - Database quality determines performance - Detects known attacks - Inflexible	- Less false alarms - Superior at detecting known attacks. - Simple design	- Need for regular database updates - Misses unknown threats - Resource intensive - Slow
Statistical Anomaly-based Intrusion Detection Models	- Newer technique for anomaly detection - Based on ML, AI, and statistics - Relies on behavioral changes to detect anomalies - Classified into supervised, unsupervised and semi-supervised	- Effective against new threats - No database needed - Highly flexible models	- Difficult to develop and maintain - High false negatives - Costly and complex algorithms - Affected by data quality
Hybrid Intrusion Detection Systems	- Combine anomaly-based and signature-based features - Integrates algorithms - Emphasize data preprocessing	- Enhanced detection rates - Dynamic models - Objective evaluation	- Complex designs - Resources intensive - Expensive to develop - Challenging to train - Affected by data quality

3. PSYCHOSOCIAL THEORETIC CONSIDERATIONS

Solving the insider threat problem requires a multidisciplinary approach as the technical controls alone may not solve the problem comprehensively. An understanding of the behavior of individuals may also play a significant part in addressing the problem. To this end, this work takes into consideration two psycho-social theories: Social Bond Theory (SBT) and Situational Crime Prevention Theory (SCPT).

3.1 The Social Bond Theory (SBT)

Travis Hirschi (Hirschi, 1969) introduced this theory in 1969 to explain criminal and delinquent behavior in society. The theory suggests that humans are inherently selfish and asocial, with this self-interest potentially leading to illegal, delinquent, and deviant behavior driven by the desire for instant gratification (Cullen & Wilcox, 2010). Under this theory, social ties are influenced by four elements: attachment, commitment, belief, and involvement, each influencing deviant behavior both individually and collectively. These elements can deter individuals from being deviant, promoting conformity to societal conduct (Kotlaja & Meier, 2018).

The theory assumes that people are inherently inclined and capable of committing crimes, but the social costs act as a deterrent. It hypothesizes that stronger social links to family, organization, church, civic, and other groups, reduce the likelihood of committing a crime. Hirschi argues that social relationships foster compliance with the shared community ideals and customs (Nickerson, 2024). Attachment, commitment, involvement and belief are the main factors to foster within an organization to help in controlling defiant behavior. Therefore, in the design of insider threat controls, it is essential not only to focus on motivation and opportunity but also to understand why individuals avoid crime. This approach will help an organization to create an environment that discourages insider threats.

3.2 The Situational Crime Prevention Theory (SCPT)

The Situational Crime Prevention Theory (SCPT) posits that crime happens as a result of two factors; motivation and opportunity, and eliminating either or both factor(s) can reduce criminal activities significantly (Ruohonen & Saddiqa, 2024). In the case of insider threats, opportunity reduction can be achieved by using fine-grained authentication and authorization procedures, strong access controls, and other relevant defensive cyber security measures. On the other hand, to reduce motivation and hold perpetrators accountable, implementing rigorous logging, monitoring, and auditing can be helpful (Safa et al., 2018). Other strategies that can reduce the potential rewards from an attack include digital signatures and watermarking, information and hardware segregation, encryption, automatic data deletion schemes, and minimizing of reconnaissance information. SCPT thus emphasizes system hardening in increase the difficulty of insiders compromising the information systems. The proposed model introduces detection that reduces the motivation for insider threats.

4. RESEARCH METHODOLOGY

This study aims to develop a more accurate insider threat detection and classification model using Deep Learning techniques. The study goes beyond technical solutions by using theoretical explanations on other methods of controlling insider threats. The study adopts a mixed research design. A review of related literature was done to establish threats and related research to assist in coming up with a more accurate model. Design science was the main research design supported by simulation and modeling. The outcome is a classification model that differentiates user benign behavior from malicious ones. Other strategies are also proposed to control insider threats.

5. EXPERIMENTAL SETUP

The test model was developed and trained on the Kaggle platform (https://www.kaggle.com/). Kaggle provides a customizable and configuration-free environment for Jupyter Notebooks and enables writing and running Python code via a browser. The Virtual Machine (VM) used for the experiment had 12.7 GB RAM, 78.2 GB HDD, 3-5 GHZ CPU, and 12GB of GPU. The essential libraries imported for model development include Scikit-learn, NumPy, Pandas, and Torch. The proposed model's performance was ascertained by comparing detection rates with a vanilla Recurrent Neural Network (RNN), Deep Neural Network (DNN), and Long Short-Term Memory (LSTM).

The model was evaluated using four metrics based on the confusion matrix. These include Accuracy, Precision, Recall, and F1 score. The metric formulae are shown below.

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$$

$$Precision = \frac{TP}{TP+FP}$$

$$Recall = \frac{TP}{TP+FN}$$

$$F_1 score = \frac{2 * Precision * Recall}{Precision + Recall}$$

TP-True Positive

TN-True Negative

FP-False Positive

FN-False Negative

6. THE PROPOSED MODEL

The proposed model utilizes the Gated Recurrent Unit (GRU) as a classifier, Synthetic Minority Oversampling Technique (SMOTE) combined with RandomUnderSampler, for data imbalance correction, the Kernel Principal Component Analysis (KPCA) for dimensionality reduction while the Adaptive Moment Estimation (Adam) is used as a model training optimization algorithm. Utilizing five files (e-mail, file access, device, login/off and LDAP) from the CERT r4.2 dataset to simulate different user network behavior characteristics, the proposed model goes through three significant development phases: data management, training, and validation.

6.1 Data Management

This step includes data selection, pre-processing, and imbalance correction. The details of each step follow.

6.1.1 Dataset Selection

Table 3 provides a comparison of various candidate datasets for insider threat detection.

Table 3. Common datasets for insider threat detection (Yuan & Wu, 2021)

Dataset	Category	Statistics
RUU	Masquerader	34 normal users and 14 masqueraders
Enron	Traitor	Half a million emails from 150 employees
Schonlau	Substituted masquerader	Unix Shell commands from 50 users
Greenberg	Authentication	Full Unix Shell commands from 168 users
TWOs	Miscellaneous malicious	24 users, 12 masqueraders, and five traitor sessions
CERT v6.2	Miscellaneous malicious	3,995 normal users and 5 Insiders
CERT r4.2	Miscellaneous malicious	> 1,000,000 normal users with < 100 malicious instances

The CERT r4.2 dataset contains a higher number of malicious instances than other datasets. It comprises the activity records of more than 1000 users in a company collected over 17 months. Less than 100 malicious insider threats were purposely introduced by experts. The CERT r4.2 dataset contains seven files out of which five were utilized (see Table 4), eliminating hypertext transfer protocol (HTTP) and psychometric files as most organizations allow bring-your-own-device (BYOD), making it challenging to track private gadgets. Also, psychometric data has legal implications that may be challenging to achieve. Table 4 illustrates the five files.

Table 4. The CERT r4.2 dataset files (Author, 2024)

File	Description	Features
Device.csv	Log of user's activity regarding connecting and disconnecting a thumb drive	ID, date, user, PC, activity (connect/disconnect)
Email.csv	Log of user's e-mail communication	ID, date, user, PC, to, cc, bcc, from, size, attachment count, content
File.csv	Log of user's activity regarding copying files to removable media devices	ID, date, user, PC, filename, content
Logon.csv	Log of user's workstation logon and logoff activity	ID, date, user, PC, activity
LDAP.csv	Eighteen (18) files for users and their roles	Employee name, ID, email, role, business unit, functional unit, department etc

6.1.2 Data Preprocessing

The ML model's performance is dependent on this step (Amato & Lecce, 2023). Preprocessing includes data cleaning, conversion, normalization, and feature selection/extraction. Categorical and non-numerical data were transformed into numerical values using one hot encoding procedure. Data normalization is achieved using the StandardScaler (minimum-maximum values are normalized to remove extremes). Other steps in the data preparation process for model training and validation are shown in Figure 1.

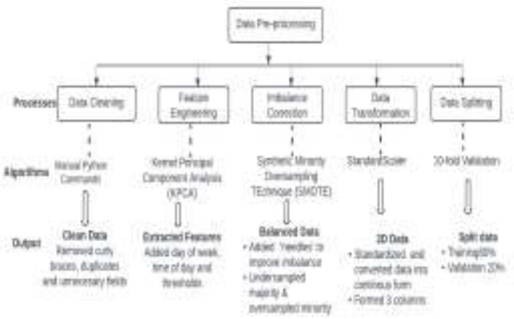
Data Pre-processing
Processes
Data Cleaning
Feature Engineering
Imbalance Correction
Data Transformation
Data Splitting
Algorithms
Manual Python Commands
Kernel Principal Component Analysis (KPCA)
Synthetic Minority Oversampling TEchnique (SMOTE)
StandardScaler
10-fold Validation
Output
Clean Data
Removed curly braces, duplicates and unnecessary fields
Extracted Features
Added day of week, time of day and thresholds
Balanced Data
• Added 'Needles' to improve imbalance
• Undersampled majority & oversampled minority
3D Data
• Standardized and converted data into continuous form
• Formed 3 columns
Split data
• Training 80%
• Validation 20%

Figure 1. Data Management process (Author, 2024)

6.1.3 Imbalanced Dataset Correction

The CERT r4.2 dataset contains significantly fewer anomalous samples than standard samples (Bin-Sarhan & Altwaijry, 2022). Evidence of imbalance is calculated by:

$$Imbalance\ Ratio\ (IR) = \frac{Minority\ Instances}{Majority\ Instance}$$

The IR was 0.00083 meaning that for every single anomaly, there are 83000 genuine records. Training the model using this unbalanced dataset will result in a model skewed towards the majority group. By employing the SMOTE and RandomUnderSampler the dataset was balanced at ratio 1:1 to reduce biases.

6.1.4 Feature Extraction

The Email, Device, File, and Logon/logoff files contain non-significant parameters for model training by removal or merging to create more comprehensive ones and generate new parameters to enhance model learning. The resultant parameters were parsed through the Kernel Principal Component Analysis (KPCA) algorithm to create a 3D dataset comprising of timestamp, activity, and target as required by the GRU algorithm.

To boost the detection accuracy of the insider threat detection and classification model, threshold setting was done for the selected files as follows:

a. *Email activity*: the recipient of emails and the number of emails sent per day was set. A user's activity is flagged if a user sends emails to external recipients (outside the domain), especially at odd hours or exceeding the number of emails sent in a day.

b. *File activity*: Files without headers or with mismatching headers are flagged.

c. *Device activity*: Abnormal use of drives, such as downloading and saving large files, using drives at odd hours, or moving drives from one PC to another, is also flagged.

d. *Logon activity*: User behavior and activities are monitored from the logon time to logoff and compared with set thresholds for specific activities.

The final result of data pre-processing is a 3D dataset containing the timestamp, activity, and target, which is to be utilized by the classifier. The dataset is split into 80% training and 20% validation using 10 cross validation split for model training and validation respectively.

6.2 Model Development

The insider threat detection and classification model development phase is a cyclic process entailing four steps, as illustrated in Figure 2. The steps entail model selection, model training, hyperparameter tuning, and transfer learning, which are further discussed.

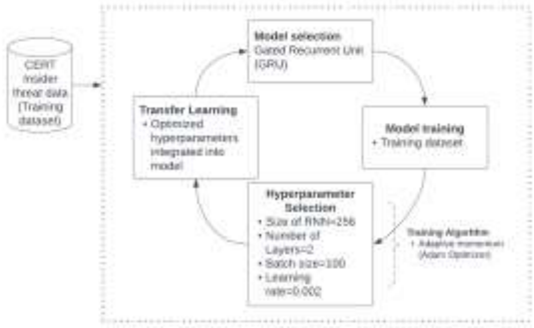

Figure 2. Model development cycle

The DL techniques result in models that can autonomously perform detection and classification (Yan & Han, 2018). Based on this premise, the GRU model is selected. Compared to LSTM, GRU is a more lightweight algorithm, using only two gates, input, and reset, to achieve efficient handling of intricate and multi-dimensional data (Malaiya et al., 2019). During training, the input layer of the GRU model feeds parsed data into the hidden layers, where recurrent computations are done. At each iteration, the hidden state is updated based on the current input and the preceding hidden state. The reset gate determines how much the preceding hidden state is altered. The gate accepts the previous hidden state and the current input as its input and generates a vector of values ranging from 0 to 1. This vector determines the extent to which the previous hidden state is "reset" during the current time step. On the other hand, the update gate determines the proportion of the candidate activation vector that should be included in the new hidden state. The candidate activation vector is a modified iteration of the preceding hidden state, which undergoes a "reset" process through the reset gate and is subsequently mixed with the current input. The computation involves using a tanh activation function, which restricts the output from -1 to 1. The output layer receives the ultimate hidden state as its input and generates the neural network output. In this case, classification of either malicious or benign user (binary classification).

Additionally, it has been established that GRU’s simple internal structure eases the training process by minimizing the computational load associated with updating the hidden state (Al-Mhiqani et al., 2021). GRU’s network has demonstrated strong performance in various applications, such as natural language processing, speech recognition, and music production.

7. RESULTS AND DISCUSSION

7.1 Model Training

This phase requires an optimization algorithm, a loss function, a metric to measure accuracy, and setting stopping criteria. An optimization algorithm is a mechanism used in DL to adjust the model's parameters and reduce a given loss function, to enhance overall model performance by reducing the objective function value. A loss function quantifies the modeling accuracy by calculating the variance between a model's predictions and the correct, actual predictions. To establish the models’ performance, an evaluation metric is used while the stopping criteria is the condition on which when the model reaches during training, the optimization process ends. This study met the requirements as follows:

7.1.1 *Optimization/training algorithm:* Adaptive Moment Estimation (Adam).

7.1.2 *Loss function:* Categorical Cross entropy.

7.1.3 Training evaluation metric: classification accuracy.

7.1.4 Stopping criteria: 3.

During training, the model goes through 20 epochs. The model’s learning capability during training is illustrated by the training and validation loss curves (Figure 3).

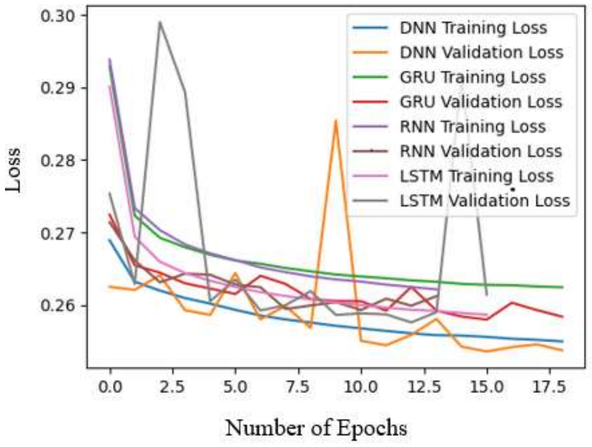

Figure 3. Training and validation loss curves for selected models (Author, 2024)

During training, the losses of the selected models are monitored as the number of epochs increases until an optimal point is reached. Loss demonstrates how effectively the model is learning to forecast the final results of the training dataset. The higher the loss value, the further the model is to predict correct results. Therefore, small values of loss are preferred.

At the beginning of training, the models are not familiar with the dataset, and therefore, high false negatives are realized and that is why all the curves start slightly above 0.26. As the number of epochs increase, the models learn and become better and reduce predicting false positives and negatives, hence improving the detection accuracy. Generally, the training loss curves measure the error (or dissimilarity) between the models’ predicted and actual output, giving insights into how performance improves over time. Sharp curves indicate the models’ instability. For all the models, learning using the training dataset was smooth until validation data was introduced. For example, LSTM and DNN models were not able to cope well with the new dataset (validation) since they encountered unknown variables and hence they performed poorly.

Other sets of curves that help to establish the learnability of the DL model are the training and validation accuracy curves. Figures 4(a)-4(d) show the training and accuracy curves for GRU, LSTM DNN, and RNN respectively.

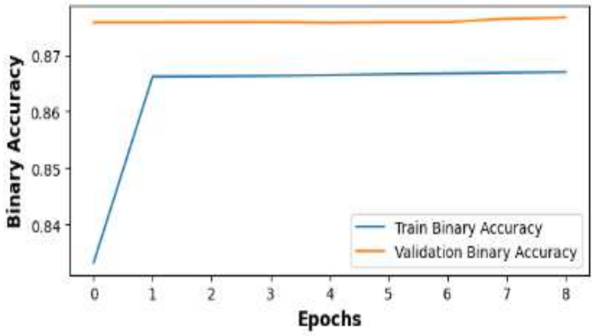

Figure 4(a). GRU Training and Validation Accuracy Curve

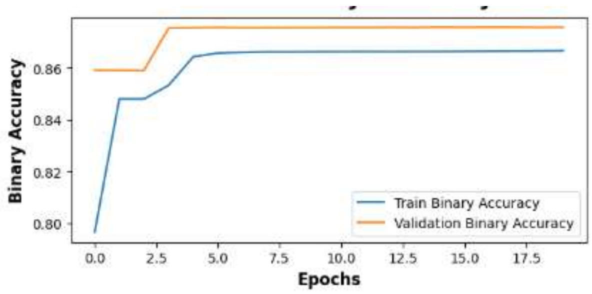

Figure 4(b). LSTM training and validation accuracy curves

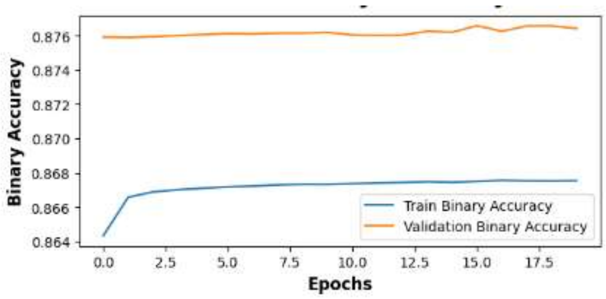

Figure 4(c). DNN Training and validation accuracy curves

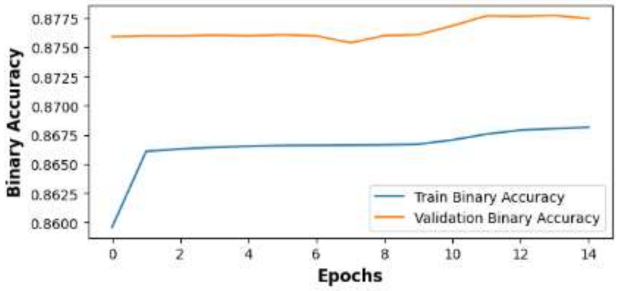

Figure 4(d). RNN training and validation accuracy curves

These figures show how the predictive accuracy of the models improves against the number of epochs. It should be noted that in all the graphs, the training accuracy is lower than the validation accuracy. This signifies that the models have learned well and can be generalized to identify threats within unknown datasets. Among the four models, the GRU model (4(a)) has a higher accuracy greater than 0.88 followed closely by LSTM, DNN, and finally RNN. This is because GRU is an improved version of LSTM which outperforms both vanilla RNN and LSTM.

A significant difference in the distance between the training and validation lines is seen among the GRU & LSTM curves with the DNN & RNN curves. The distance between the training and validation curves illustrates the DL model's learnability. Better models have a smaller gap between the two curves than poor ones. A bigger distance between the curves means the models might be overfitting which is a concern when accurate predictions are required.

In all the four curves, validation results are better than training results. For example, for the GRU- based insider threat detection and classification model, the training curve optimizes at 0.865 while the validation curve at approximately 0.882 which translates to an increment. This means that although there is a difference in the accuracy rates, the selected DL models fit to be used for insider threat detection but the higher the value, the better the DL model. While the training and validation loss curves decrease drastically, the training and validation accuracy curves increase steadily. This shows that the models are generalizing well with the unknown datasets.

7.2 Hyperparameter Tuning

Hyperparameter tuning involves the manipulation of the settings until the model's learning capabilities are optimal and stabilized. The baseline hyperparameters for the Insider Threat Detection and Classification (ITDC) model are given in Table 5.

Table 5. ITDC model hyperparameters (Author, 2024)

Hyperparameter	DL Models	Remark
Batch size	256	To utilize GPU power
Validation split	0.2	80% used for training
Stopping criteria	3	# of epochs set to terminate model training.
Number of epochs	20	No. of times the entire dataset is passed through
Learning rate	0.001	The pace of the model's learning

All the selected models were trained using the same hyperparameters. These are the baseline configurations of the DL models and should never be confused with parameters (variables) that belong within the dataset. This stage was very challenging because of inadequate computing resources. The virtual Machine (VM) subscribed to performed dismally and might have affected the models' training results. In addition, not all the models perform well with these baseline results and weights assigned to them. Forcing them to start training at the same level decreases the chances of having correct predictions.

During training, there is a need for varying the hyperparameter settings to gauge the performance of different levels. This ensures that the changes in performance are noted with different hyperparameters until you reach the optimized level where the model makes highly accurate detection predictions.

7.3 Transfer Learning

To assess the models' learnability and generalizability, they were evaluated using the 20% validation dataset acquired from the 80-20 cross-validation split. The ITDC model is evaluated alongside the other three selected DL models using the four metrics: Recall, Precision, Accuracy, and f-measure. Having multiple tests ensures the model's overall robustness is comprehensively tested. The evaluation results are presented in Table 6.

Table 6. Selected DL models Evaluation Table (Author, 2024)

Classifier	Precision	Recall	f_1 *score*	Accuracy
RNN	0.9331	0.7838	0.8524	0.8668
DNN	0.9285	0.7881	0.8525	0.8672
LSTM	0.9358	0.7850	0.8527	0.8676

GRU	0.9393	0.7890	0.8533	0.8683

The detection and classification performance of the selected DL models in Table 6 indicate superior performance by the GRU-based model just as indicated by the training and validation accuracy curves. GRU algorithm has only two gates that enable it to train faster and perform better than the rest. Although they do not have storage for long-term dependencies, they tend to converge faster during training.

The performance of our model was evaluated using four metrics: Accuracy, Precision, Recall, and $f1$ $score$ and compared to other DL techniques as indicated in Table 6. A classification Accuracy of 0.8683 denotes that our model accurately predicted approximately 87% of all the predictions made. A high Precision of 0.9393 implies out of the predictions made, 93% were accurately predicted and were actual threats; a Recall of 0.7890 means the model correctly remembers 78.90% of the threats learned. Recall is also known as True Positive Ratio (TPR). Lastly, the f-measure of 0.8533 is a harmonic mean that expresses the balance between recall and Precision, but it is interpreted depending on the nature of the model. This is whether false positives are costlier than false negatives or vice versa. Our ITDC model was balanced, making it more effective in insider threat detection.

If a DL model exhibits high Precision but poor Recall (our case), it accurately identifies threats and fails to detect a few that it should have. This approach may be deemed appropriate if the intention is to prevent users from being irritated by false alerts. However, it also risks exposing users to additional undesirable and potentially detrimental threats that were overlooked. Conversely, when Precision is low, but Recall is high, it indicates that your ML model excels at detecting threats, but it also mistakenly identifies several acceptable actions as threats. While prioritizing user safety is commendable, implementing such stringent measures may inadvertently lead to user dissatisfaction due to frequent false alarms and erode their confidence in the system.

8. CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

Insider threats have been thriving as more organizations continue to digitize their data. Classical solutions such as firewalls, IDS and IPS have failed to prevent this vice due to the characteristics that insiders possess (trusted, aware of systems, and enjoy authorized access). Anomaly detection has been used in other fields such as fraud detection by operating on the belief that the user’s current behavior resembles past behavior and hence deviation implies a threat. This technique has been adopted by ML techniques to detect changes in behavior among computer network users. Due to the vast data generated on the network, ML models have failed to correctly identify threats.

DL is a technique that employs more layers for refined detection and classification and has now been applied for insider threat detection. Despite the improved performance, these systems become highly biased when faced with imbalanced and highly dimensional datasets. Detection rates plunge drastically because the resultant models are usually skewed toward the majority class.

Data is the main component of ML and DL model development and hence, determines the models’ performance. To improve the insider threat detection and classification model’s detection accuracy, this study employed data enhancement techniques to improve the model’s insider threat detection rates. This involved using data imbalance correction techniques and data augmentation to improve the parameters for model training. Moreover, unlike other authors who use a single file from the CERT r4.2 (e.g file, email login/off etc), this study employed five files that represent more user characteristics to improve detection rates. This is because a file access activity when used to train a model may not establish threats in email exchange activities.

A layered approach to cybersecurity is always recommended. This study was a double technique strategy of controlling insider threats. Combining DL, Social Bond Theory, and Situational Crime Prevention Theory provides a robust framework for detecting insider threats. DL models thrive at analyzing and discovering complicated patterns across large behavioral datasets, making them ideal for detecting subtle indicators of insider threats. Social Bond Theory provides a psychological perspective by emphasizing the strength of an individual's attachment, commitment, involvement, and conviction inside the organization, which can indicate possible hazards. At the same time, Situational Crime Prevention Theory builds on this paradigm by highlighting the necessity of minimizing the possibilities for crime through methods such as increased surveillance, reduced anonymity, and strengthened organizational controls. Combining these theories, the model identifies possible risks based on behavior and social ties and considers the situational aspects that may permit insider threats. This complete strategy improves the ability to detect and mitigate insider threats, resulting in a more secure organizational environment.

8.2 Future Work

The CERT r4.2 dataset is a synthetic dataset that was injected with “fake” malicious activities to depict what happens within an organization. In the future, this study recommends the use of real organizational datasets. To enhance model performance, this study also recommends employing Natural Language Processing (NLP) applied to email data for email content analysis. This will ensure that the model can detect anomalies in emails using the content of the email in addition to other email characteristics used in this study.

9. ACKNOWLEDGMENTS

Thanks to my supervisors who guided me during this research and all friends and colleagues who assisted in proof reading this paper.

10. REFERENCES

[1] Aldairi, M., Karimi, L., & Joshi, J. (2019). A Trust Aware Unsupervised Learning Approach for Insider Threat Detection (p. 98). https://doi.org/10.1109/IRI.2019.00027

[2] Al-mhiqani, M. N., Ahmad, R., Abidin, Z. Z., Yassin, W., Hassan, A., & Mohammad, A. N. (2020). New insider threat detection method based on recurrent neural networks. Indonesian Journal of Electrical Engineering and Computer Science, 17(3), Article 3. https://doi.org/10.11591/ijeecs.v17.i3.pp1474-1479

[3] Al-Mhiqani, M. N., Ahmed, R., Zainal, Z., & Isnin, S. N. (2021). An Integrated Imbalanced Learning and Deep Neural Network Model for Insider Threat Detection. International Journal of Advanced Computer Science and Applications, 12(1). https://doi.org/10.14569/IJACSA.2021.0120166

[4] Alsowail, R., & Al-Shehari, T. (2022). Techniques and countermeasures for preventing insider threats. PeerJ

Computer Science, 8, e938. https://doi.org/10.7717/peerj-cs.938

[5] Amato, A., & Lecce, V. (2023). Data preprocessing impact on machine learning algorithm performance. Open Computer Science, 13. https://doi.org/10.1515/comp-2022-0278

[6] Bin Sarhan, B., & Altwaijry, N. (2022). Insider Threat Detection Using Machine Learning Approach. Applied Sciences, 13, 259. https://doi.org/10.3390/app13010259

[7] CERT. (2014). 2014 US State of Cybercrime Survey -3. https://insights.sei.cmu.edu/documents/3858/2014_017_001_298322.pdf

[8] Chalapathy, R., & Chawla, S. (2019). Deep Learning for Anomaly Detection: A Survey. https://doi.org/10.48550/arXiv.1901.03407

[9] CISA. (2024). Defining Insider Threats | CISA. https://www.cisa.gov/topics/physical-security/insider-threat-mitigation/defining-insider-threats

[10] Cullen, F. T., & Wilcox, P. (2010). Hirschi, Travis: Social Control Theory. In Encyclopedia of Criminological Theory. SAGE Publications, Inc. https://shorturl.at/zC7QP

[11] CyberArk. (2017). The Everyday Insider Threat. https://www.cyberark.com/resources/blog/the-everyday-insider-threat

[12] Dawson, M., & Omar, M. (Eds.). (2015). New Threats and Countermeasures in Digital Crime and Cyber Terrorism: IGI Global. https://doi.org/10.4018/978-1-4666-8345-7

[13] Fortinet. (2024). What is Defense in Depth? Defined and Explained. Fortinet. https://www.fortinet.com/resources/cyberglossary/defense-in-depth

[14] Griffiths, C. (2024). The Latest Cyber Crime Statistics (updated July 2024) | AAG IT Support. https://aag-it.com/the-latest-cyber-crime-statistics/

[15] Hirschi, T. (1969). Social Bond/Social Control Theory. Sage Publications. https://www.sagepub.com/sites/default/files/upm-binaries/36812_5.pdf

[16] Hu, T., Niu, W., Zhang, X., Liu, X., Lu, J., & Liu, Y. (2019). An Insider Threat Detection Approach Based on Mouse Dynamics and Deep Learning. Security and Communication Networks, 2019, 1–12. https://doi.org/10.1155/2019/3898951

[17] Jiang, W., Tian, Y., Liu, W., & Liu, W. (2018). An Insider Threat Detection Method Based on User Behavior Analysis. In Z. Shi, E. Mercier-Laurent, & J. Li (Eds.), Intelligent Information Processing IX (Vol. 538, pp. 421–429). Springer International Publishing. https://doi.org/10.1007/978-3-030-00828-4_43

[18] Jiang, Y., Luo, J., Huang, D., Liu, Y., & Li, D. (2022). Machine Learning Advances in Microbiology: A Review of Methods and Applications. Frontiers in Microbiology, 13. https://doi.org/10.3389/fmicb.2022.925454

[19] Kanellopoulos, A.-N. (2024). Insider threat mitigation through human intelligence and counterintelligence: A case study in the shipping industry. Defense and Security Studies, 5, 10–19. https://doi.org/10.37868/dss.v5.id261

[20] Kong, I., & Bashir, M. (2022). A Closer Look at Insider Threat Research. 53, 29–35.

[21] Kont, M., Pihelgas, M., Wojtkowiak, J., Trinberg, L., & Osula, A.-M. (2021). Insider Threat Detection Study.

[22] Kotlaja, M., & Meier, R. (2018). Social Bonds Theory of Crime.

[23] Le, D. C., & Zincir-Heywood, A. N. (2019). Machine learning based Insider Threat Modelling and Detection.

[24] Legg, P. A., Buckley, O., Goldsmith, M., & Creese, S. (2017). Automated Insider Threat Detection System Using User and Role-Based Profile Assessment. IEEE Systems Journal, 11(2), 503–512. https://doi.org/10.1109/JSYST.2015.2438442

[25] Lv, Q., Zhang, S., & Wang, Y. (2022). Deep Learning Model of Image Classification Using Machine Learning. Advances in Multimedia, 2022, e3351256. https://doi.org/10.1155/2022/3351256

[26] Malaiya, R. K., Kwon, D., Suh, S. C., Kim, H., Kim, I., & Kim, J. (2019). An Empirical Evaluation of Deep Learning for Network Anomaly Detection. IEEE Access, 7, 140806–140817. https://doi.org/10.1109/ACCESS.2019.2943249

[27] Munir, M., Siddiqui, S. A., Dengel, A., & Ahmed, S. (2019). DeepAnT: A Deep Learning Approach for Unsupervised Anomaly Detection in Time Series. IEEE Access, 7, 1991–2005. IEEE Access. https://doi.org/10.1109/ACCESS.2018.2886457

[28] Nasir, R., Afzal, M., Latif, R., & Iqbal, W. (2021). Behavioral Based Insider Threat Detection Using Deep Learning. IEEE Access, 9, 143266–143274. https://doi.org/10.1109/ACCESS.2021.3118297

[29] Nicolaou, A., Shiaeles, S., & Savage, N. (2020). Mitigating Insider Threats Using Bio-Inspired Models. Applied Sciences, 10. https://doi.org/10.3390/app10155046

[30] Nurse, J., Legg, P., Buckley, O., Agrafiotis, I., Wright, G., Whitty, M., Upton, D., Goldsmith, M., & Creese, S. (2014). A Critical Reflection on the Threat from Human Insiders – Its Nature, Industry Perceptions, and Detection Approaches. 8533. https://doi.org/10.1007/978-3-319-07620-1_24

[31] Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P., Shyu, M.-L., Chen, S.-C., & Iyengar, S. S. (2019). A Survey on Deep Learning: Algorithms, Techniques, and Applications. ACM Computing Surveys, 51(5), 1–36. https://doi.org/10.1145/3234150

[32] Raval, M. S., Gandhi, R., & Chaudhary, S. (2018). Insider Threat Detection: Machine Learning Way. In M. Conti, G. Somani, & R. Poovendran (Eds.), Versatile Cybersecurity (pp. 19–53). Springer International Publishing. https://doi.org/10.1007/978-3-319-97643-3_2

[33] Ruohonen, J., & Saddiqa, M. (2024). What Do We Know About the Psychology of Insider Threats? https://arxiv.org/html/2407.05943v1

[34] Safa, N. S., Maple, C., Watson, T., & Von Solms, R. (2018). Motivation and opportunity based model to reduce information security insider threats in organisations. Journal of Information Security and Applications, 40, 247–257. https://doi.org/10.1016/j.jisa.2017.11.001

[35] Saxena, N., Hayes, E., Bertino, E., Ojo, P., Choo, K.-K. R., & Burnap, P. (2020). Impact and Key Challenges of Insider Threats on Organizations and Critical Businesses.

Electronics, 9, 1460. https://doi.org/10.3390/electronics9091460

[36] SEI. (2022). Common Sense Guide to Mitigating Insider Threats, Seventh Edition. Common Sense Guide to Mitigating Insider Threats, Seventh Edition. https://insights.sei.cmu.edu/library/common-sense-guide-to-mitigating-insider-threats-seventh-edition/

[37] Singh, M., Mehtre, B., S., S., & Govindaraju, V. (2023). User Behaviour based Insider Threat Detection using a Hybrid Learning Approach. Journal of Ambient Intelligence and Humanized Computing, 14, 1–21. https://doi.org/10.1007/s12652-023-04581-1

[38] T. N., N., & Pramod, D. (2024). Insider Intrusion Detection Techniques: A State-of-the-Art Review. Journal of Computer Information Systems. 106–123. https://doi.org/10.1080/08874417.2023.2175337

[39] Tian, Z., Shi, W., Tan, Z., Qiu, J., Sun, Y., Jiang, F., & Liu, Y. (2020). Deep Learning and Dempster-Shafer Theory Based Insider Threat Detection. Mobile Networks and Applications. https://doi.org/10.1007/s11036-020-01656-7

[40] Wei, Y., Chow, K.-P., & Yiu, S.-M. (2021). Insider threat prediction based on unsupervised anomaly detection scheme for proactive forensic investigation. Forensic Science International: Digital Investigation. https://doi.org/10.1016/j.fsidi.2021.301126

[41] Yan, B., & Han, G. (2018). LA-GRU: Building Combined Intrusion Detection Model Based on Imbalanced Learning and Gated Recurrent Unit Neural Network. Security and Communication Networks. https://doi.org/10.1155/2018/6026878

[42] Yuan, S., & Wu, X. (2020). Deep Learning for Insider Threat Detection: Review, Challenges and Opportunities. https://doi.org/10.48550/arXiv.2005.12433