RESEARCH ARTICLE OPEN ACCESS

Digital Forensics for Detecting and Investigating Cyber Malicious Activities

Marathi Muni Babu . M Sai Harshini . P V Koteswara Rao . R Gowtham . G Jamuna

Department of CSE (IoT and Cyber Security including Block Chain Technology),
Annamacharya Institute of Technology & Sciences (Autonomous),
Tirupati, A.P, India.

DOI: **10.5281/zenodo.18672461**
Received: 29 January 2026 / Revised: 12 February 2026 / Accepted: 17 February 2026
Corresponding Author: marathi.muni@gmail.com*
©Milestone Research Publications, Part of CLOCKSS archiving

Abstract – Cybersecurity is an urgent concern in this age of rapid expansion of digital infrastructures, especially due to insider threats. These are sophisticated threats where traditional signature-based detection methods have proven much less effective, since these attacks are by people who have legitimate access to sensitive data. In this paper, several ML models, namely Logistic Regression, Random Forest, Support Vector Machine, Decision Trees, and XGBoost, have been experimented with for detecting insider threats in cybersecurity. XGBoost proved to be the best among the compared ML models, with an accuracy of 94.2%. However, the proposed model CNN outperformed all other algorithms and achieved 95.0% accuracy along with the highest precision and F1-score. This confirms that deep learning techniques are much better at capturing complex patterns in cyber activities than ML techniques. While the proposed CNN resulted in excellent performance, several challenges remain to be explored, such as the problem of class imbalance, anomaly detection in real time, and explanation of anomalies. This paper presents a proposal that the integration of advanced machine learning and deep learning models is crucial for improvement in scalable, real-time, and accurate cybersecurity solutions.

Index Terms – Cybersecurity, Insider Threats, Machine Learning, Digital Forensics, CNN, Anomaly Detection.

I. INTRODUCTION

It may be noted that in recent years, with the development of digital infrastructures, cybersecurity threats have assumed great priority for organizations across various sectors. Cyber threats, particularly insider threats, continue to remain a major risk for data security. Unlike external threats from cyber attackers outside the organization, individuals inside the organization with valid rights to access sensitive information tend to misuse such rights for malicious purposes, which may be difficult to detect. Traditional detection systems based on signature scanning may find this difficult [1].

Recent developments in AI and ML have presented encouraging results when related to cybersecurity. Almost all the solutions based on AI use different models of Machine Learning for the detection of malicious activities. The solutions like Convolutional Neural Networks (CNN)Logistic Regression (LR), Random Forest (RF), Support Vector Machine (SVM), Decision Trees (DT), and XGBoost are being widely used for the automation of anomaly detection, prediction of threats, and increasing system security. These models can process large volumes of data, mine patterns, and generate real-time insights-skills that are very key to insider threat detection and data breach prevention [2][3]. There are several reports about the effectiveness of these models in dealing with cyber threats. CNN, GBoost, due to this advantage, has been identified as an effective model in detecting insider threats with high accuracy [4]. On the other hand, Random Forest and Decision Trees are considered effective models due to their robustness and ability to interpret the patterns and means of the given data, particularly in low-resource environments [5]. There are also additional AI-based techniques to incorporate deep learning to enhance the efficacy of these models in detecting cyber threats. Nonetheless, despite all these developments and considerations in dealing with cyber threats, there still exists a gap in dealing with insider threats in real-world environments [6].

Although existing machine learning models have demonstrated success in predicting and detecting various cyber-attacks, insider threats, especially those involving subtle and complex patterns in user behavior, continue to challenge traditional methods [7]. The solutions that have already been implemented possess some drawbacks or limitations, for example, in dealing with imbalanced datasets, understanding the actual reason for the anomaly, and being able to perform real-time detection. In addition, it has become important to develop solutions that can easily cope with the rising data volumes and rising complexity found within modern organization infrastructures, while providing a comprehensive cybersecurity environment that includes detection capabilities, among other advantages [8]. The existing security mechanisms possess certain limitations or disadvantages, such as being designed to detect specific types of attacks, such as outsider attacks, or failing to offer accurate, real-time threat identification capabilities, among others. Consequently, this research aims at developing a more effective model using a deep learning-based approach that can easily cope with huge data volumes, while offering accurate detection capabilities, potential, and promise within scalable, real-time, and evolving contexts [9], [10] that consider threats, digital systems, and attacks by insiders.

Our main contributions are as follows:

- This paper introduces a Convolutional Neural Network (CNN)-based model, which significantly outperforms other models, achieving an accuracy of 95.0%, with the highest precision and F1-score, demonstrating the potential of deep learning in detecting complex insider threats.
- We present the integration of deep learning techniques, such as Convolutional Neural Networks (CNN), to enhance the effectiveness of insider threat detection, which represents a major step forward in AI-based cybersecurity solutions.

- This work contributes to digital forensics by developing scalable solutions based on machine learning, capable of adapting to the increasing complexity of cyber threats, providing the foundation for more complete and adaptable digital forensics solutions.

II. LITERATURE SURVEY

Cybercrime has developed very rapidly as a significant menace to the digital infrastructures that necessitate sophisticated detection tools to facilitate the surveillance of cybersecurity as well as forensic examination. The signature-based methods are becoming less and less efficient against advanced attacks that include phishing, malicious URLs, ransomware, and distributed denial-of-service (DDoS). Alsubaei et al. [11] presented a new phishing detector deep learning system and called RNT-J that builds on the ResNeXt framework but adds a GRU unit to learn sequential features. The model is also enhanced by the incorporation of SMOTE to balance the data, an autoencoder-ResNet-based feature extraction method (EARN), and Jaya optimization to enhance the performance of classification. Through experiments, it was proven that the proposed framework had an accuracy between 83% to 98 %, beating current phishing detection methods by 11%-19. Moreover, the use of SMOTE boosted the detection accuracy by a great extent on the models without any oversampling.

Kesarwani and Rajesh [12] introduced a machine learning based method to detect malicious URLs by doing a comparative analysis of SVM, random forest, and logistic regression classifiers. Their research marked that the accuracy of Random Forest was the greatest, with 96.6%, which indicates a high ability to separate between malicious and legitimate URLs. The assumed model will usher in the direction of proactive automated threat detection that will provide an effective way to identify malicious web resources. Nevertheless, the framework is limited by its use of benchmark datasets and needs further modification to adequately address the changes in URL obfuscation and evasion methods. It is suggested by Kajjam et al. [13] that their framework of intrusion detection is based on the Random Forest and should be used to block the threat of data leakage by both DDoS and phishing attacks. The model was found to be more effective than traditional methods like SVM and SNORT by yielding 98% accuracy and a true positive rate near 99% with mixed network traffic properties paired with URL-related characteristics. The research meets a requirement of scalable multi-vector detection of cybercrime in real-time settings.

Puchalski et al. [14] presented Trustworthy Cyberattack Detector (TCAD) as an AI-based solution aimed at identifying and describing cyberattacks in real-time and during offline forensic investigation. The framework is meant to improve the investigation of cybercrime by minimizing the false positives and enhancing flexibility in contrast to the traditional signature-based and anomaly-based detection systems of intrusions. Even though TCAD offers an important contribution to the field of forensics, in terms of quantitative performance data, the study does not offer any specific results, which means that it can not be compared to the state-of-the-art models of cyberattack detection, which casts doubt on its extrapolability to the continuously changing patterns of attacks. Khan and Alkhathami [15] relies on machine learning models, including Random Forest, AdaBoost, Logistic Regression, Perceptron, and deep learning networks. Their experimental analysis on the CIC IoT datasets indicated that the highest accuracy was recorded at ~ 99.55 % with respect to Random Forest, whose response time was better in real-time intrusion detection. The work bridges a significant gap in securing the healthcare IoT infrastructures against cyber threats. Nevertheless, this method is still constrained by data set reliance and difficulty in extrapolation between heterogeneous IoT systems that are implemented in real healthcare environments.

Ghozi et al. [16] created an XGBoost-based ransomware detection model to detect zero-day ransomware attacks as opposed to the conventional signature-based ransomware detectors. The gradient-based model proposed got an F1-score of 97.60%, which is better than other ensemble techniques, which include Gradient Boosting(97.20%), Random Forest (96.94%)t, and AdaBoost(96.50%). The contribution of this work to adaptive ransomware detection is that it enhances robustness against the emerging ransomware variants. However, weaknesses are that data availability is limited and that additional validation should be done in real-life working scenarios. Almansouri et al. [17] examined the predictability of machine learning in cybercrime investigations by using Decision Trees and Feed-Forward Neural Networks to predict the outcome of the investigation using Kuwait 3CD data. Their results showed that brute-force feature selection offered better predictive capabilities than officer-guided feature selection, which brings to the fore the possibilities of data-driven methods of forensic analytics. Nevertheless, the research is confined due to the use of a national dataset, which makes it less applicable in other legal and investigation settings.

Dananjana et al. [18] suggested a machine learning-based criminal behavior analysis system that integrates LSTM networks and autoencoders to recognize unusual browsing behavior patterns that are associated with intent to commit cybercrimes. The paper recommends that automated sequence-based behavioral modeling has the potential to increase the speed and smartness of forensic investigations over traditional instruments. The work is new, yet it does not present quantitative performance outcomes in detail and creates problems with privacy and generalization of the outcomes across different behavioral patterns of the user. Pandian et al. [19] when studying illegal cyber activity in terms of evidence gathering on laptops, mobiles, network traces, and cloud services, as well as removable drives. Their results placed malware among the most widespread types of attacks and highlighted the relevance of the forensic methods, including disk imaging, packet analysis, and memory forensics. Although the research has useful practical implications, it is mostly descriptive, and it lacks integration of automated machine learning-based detection systems. Oh et al. [20] came up with a journal-based timestamp manipulation detection algorithm in NTFS file systems. The approach maximizes the identification of forged file metadata incidences with minimum false classification of normal operations. The suggested method shows a better output than the current methods used in detecting the manipulations of timestamps that were not reported before.

III. METHODOLOGY

A. Dataset Description

In this work, we used the Cybercrime Forensic Dataset, a Kaggle dataset provided by user jimohyusuf, which numerous researchers and practitioners have used in the fields of cybersecurity and forensic analysis. The dataset consists of 7,400 rows, all in the form of a CSV file simulating cyber activities in pertinent tables. The dataset contains various features as it is believed to represent and display different aspects of these simulated events. Some of these features are related to suspicious activity that is suspected to have occurred and aspects of network traffic, as well as other forensic-related features.

Besides feature columns, there is also information on one or more class labels that can be used for supervised learning. Such labels can help identify normal behavior and possible cybercrimes, fitting into categories involving classification, anomaly detection, and research for forensics.

B. Data Preprocessing

In this study, a systematic preprocessing pipeline was used to the Cybercrime Forensic Dataset to assure data consistency, minimize noise, prevent information leakage, and prepare the dataset for accurate machine learning classification.

- Missing Value Handling: Missing values are often found in behavioral and forensic logs because monitoring and recording aren't always complete. We used feature-type-specific imputation strategies to ensure the integrity of the dataset by avoiding the loss of any records; thus, we imputed numerical features where necessary and categorical features. Lastly, after cleaning, we saved the processed dataset for reproducibility and auditability.
- Target Variable Encoding: The textual labels (Normal and Suspicious) in the dataset serve as classification targets. Because supervised learning algorithms require numerical outputs, the labels were encoded in binary form:

$$\mathrm{y} = \begin{cases} 0, & \textit{Normal activity} \\ 1, & \textit{Suspicious activity} \end{cases}$$

  The class distribution was Normal (0) and Suspicious (1).
- Categorical Feature Encoding: Machine learning methods are unable to directly interpret nominal behavioural features. As a result, selected categorical predictors such as activity_type, resource_accessed, and action were transformed using label encoding:

$$\mathrm{f}\,(\mathrm{x}) = \mathrm{k}, \;\; \mathrm{k} \in \{0, 1, 2, \ldots\ldots\ldots, \mathrm{n}\}$$

  where each unique category is assigned an integer identifier.

  Importantly, post-event forensic descriptors, such as anomaly_type, were purposefully removed to prevent data leakage, as they contain information that would not be available during real-time detection.
- Feature Selection and Leakage Prevention: To ensure generalization, identifier-based data, including timestamps, user IDs, IP addresses, file names, and forensic comments, were deleted. These characteristics may lead models to memorize patterns rather than learning behavioral indications. The final feature-target separation was specified as follows:

$$\mathrm{X} = \frac{D}{\{ID,\;\; anomaly, y\}}$$

$$\mathrm{y} = \mathrm{label}$$

  where, X represents behavioural predictors, and y defines the encoded classification target.
- Dataset Partitioning: The cleaned dataset was split into training and test sets using an 80:20 stratified split. Stratification guarantees that class proportions are uniform across both sets:

$$\mathrm{D} = D_{train} \cup D_{test}$$

- Feature Scaling: Because methods like Logistic Regression are sensitive to feature size, numeric attributes were normalized using z-scores. To prevent leakage, scaling was applied only to the training data. The standardization procedure is defined as:

$$x_{scaled} = \frac{\mathrm{x} - \mu}{\sigma}$$

D. Proposed Methodology

In this study, we present a trainable deep learning architecture based on a Convolutional Neural Network (CNN) for identifying and classifying cyber harmful actions. Deep learning systems are made up of several sequential processing modules, each of which performs a specialized transformation on the incoming data. CNN-based architectures learn discriminative representations automatically from training data, as opposed to classic machine learning algorithms that rely largely on handcrafted feature extraction. The proposed deep classifier is trained end-to-end, which means that all trainable parameters across all layers are jointly optimized to reduce the difference between the projected output and the intended ground truth label. Figure 4 depicts the proposed model architecture.

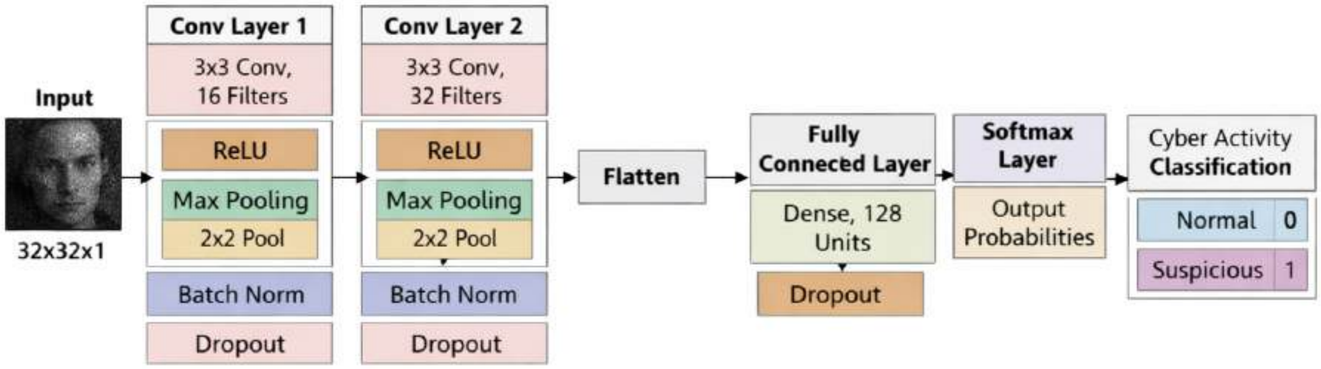

Fig. 1: Graphical representation of the proposed model architecture

- Hierarchical Feature Learning in Deep Networks: A deep neural network can be considered a stacked composition of nonlinear transformations:
  $$f(x) = f_L(f_{L-1}(\ldots\ldots f_2(f_1(x))\ldots\ldots))$$
  where, x defines the input sample, and L is the total number of layers.
  This hierarchical structure allows the network to learn low-level patterns in early levels before gradually building higher-level abstract representations in subsequent layers.
- CNN-Based Feature Extraction: CNNs are particularly effective because they leverage local correlations and weight sharing to dramatically reduce the number of trainable parameters while maintaining learning capacity. The fundamental operation in a convolutional layer is defined as:
  $$z_k = W_k * x + b_k$$
  where, W_k is the convolution kernel, b_k is the bias term, and z_k defines the generated feature map.
- Activation Function (ReLU): A nonlinear activation is added after convolution to add flexibility and prevent saturation. This study makes use of the Rectified Linear Unit (ReLU):
  $$a_k = \max(0, z_k)$$
  This activation improves learning stability while preventing negative feature suppression.
- Pooling for Dimensionality Reduction: Max-pooling is used to compress feature representations while lowering computing costs:
  $$p_k = \underset{(i,j)\in R}{max}(a_k(i,j))$$
  where, R represents the pooling region, which is typically 2x2 in size. Pooling enhances invariance to minor distortions and noises.

- Regularization Techniques: Two regularization approaches are combined to promote generalization and reduce overfitting.
  - Batch Normalization: Batch normalization helps to stabilize training by normalizing intermediate activations.

$$\hat{x} = \frac{x - \mu_B}{\sqrt{\sigma_B^2 + e}}$$

  - Dropout: Dropout randomly inhibits neurons during training:

$$h' = h.r, \qquad \text{r} \sim \text{Bernoulli(p)}$$

  This decreases reliance on specific neurons and increases robustness.
- Softmax Output Layer: Finally, classification is carried out using the Softmax function:

$$\text{P (y = j|x)} = \frac{e^{uj}}{\sum_{c=1}^{C} e^{uc}}$$

  where, C is the number of activity classes, and P (y = j|x) is the predicted probability for class j. The predicted label is obtained as:

$$\hat{y} = \underset{j}{arg\ max}\ P\ (y = j|x)$$

- Model Objective Function: The training objective is to minimize the cross-entropy loss:

$$\mathcal{L} = - \sum_{i=1}^{N} y_i \log\ (\hat{y}_i)$$

  where, y_i is the true label, $\hat{y}_i$ is the predicted probability, and the N is the total number of samples.

E. Deployment Architecture

A deployment diagram. 2 also shows where different components or pieces of software are located in the real world. It provides a physical overview of the hardware organization of the system by showing the different ways the organizational pieces of the forensic investigation system are deployed or located. Here, nodes represent physical components such as user workstations and application or database servers where the system can be deployed or run. Components refer to the essential pieces of system software used for processing investigations, report writing, and managing data. Artifacts refer to databases used for forensic data storage, data used for investigations, and system configuration used for proper operation. Associations refer to the different software components deployed or located on the system nodes, whereas communication paths show the different paths through which communication occurs between different client systems, application systems, and databases during the investigation process.

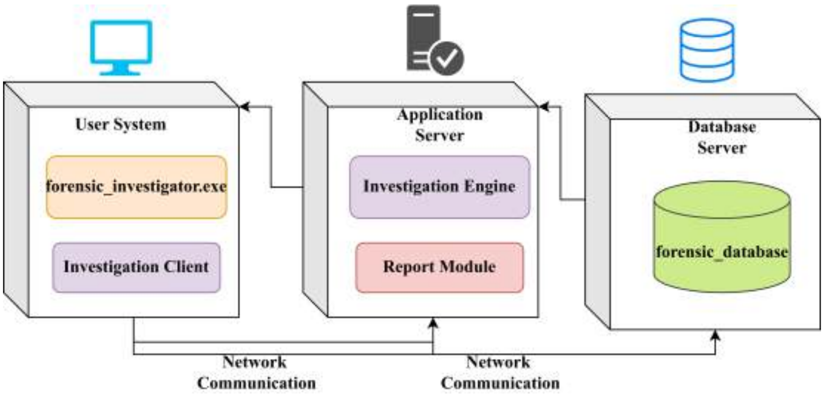

Fig. 2: Deployment architecture

IV. HARDWARE AND SOFTWARE SPECIFICATIONS

It needs to be an x86-64 (64-bit) processor with at least 2 GHz of processing capacity for optimal performance. Furthermore, it needs at least 512 MB of memory to be freely available as RAM and at least 5 GB of memory freely available for disk space. If one needs to run it on Windows, it supports Microsoft Windows 7/8/10/11 and needs to have Microsoft .NET Framework 4.5 or later. For development environments, it supports Microsoft Visual Studio 2012 or later, Python IDLE, and Anaconda with Spyder as the recommended Python code development platform due to its feature-rich support. For macOS, the platform needs to be at least macOS 10.13 or later. For XCode and GNU Make version requirements, it needs XCode 9.3 and GNU Make version 3.81 to develop applications. Linux systems require a Linux 3.10 kernel or newer, along with glibc 2.17, gcc 4.8, and GNU Make 3.81. For software stacking, the essential Python IDEs such as Spyder and Anaconda; machine learning libraries: NumPy, Pandas, scikit-learn, Matplotlib, Seaborn, Flask, PyMySQL, and the core algorithms on machine learning, which include logistic regression, decision trees, random forest, support vector machine classification, naïve Bayes, and gradient boosting classifier. These aforementioned specifications set up a really strong pedestal for the development and deployment of machine learning models to detect and analyze the cyber threats effectively.

V. RESULT & DISCUSSION

The results presented in Table 1 provide a simple comparison between our suggested CNN model and different classifiers used by other researchers in the field, including the Decision Tree Classifier (DT), Support Vector Machine Classifier (SVM), Random Forest Classifier (RF), and XGBoost Classifier. We used the aforementioned classifiers while keeping various metrics such as precision, recall rate, F1 score, and accuracy scores in mind for a comprehensive comparison.

Table 1: Performance analysis of the proposed and several baseline models

Model	Precision (%)	Recall (%)	F1-Score (%)	Accuracy (%)
Decision Tree (DT)	88.4	86.9	87.6	88.1
Support Vector Machine (SVM)	90.2	89.5	89.8	90.0
Random Forest (RF)	93.1	92.4	92.7	93.0
XGBoost	94.0	93.6	93.8	94.2
Proposed CNN Model	95.3	94.8	95.0	95.0

Starting with DT, it had the lowest accuracy of 88.1% and an F1 of 87.6%. DT is an easy classifier, being very interpretable but risking overfitting because of its reliance on a single decision boundary. It is especially risky in the case of forensic data, as the patterns are intricate. Thus, DT failed to effectively distinguish between normal and suspicious behaviors. SVM has enhanced the performance of DT. Its accuracy was 90.0%, and it had an F1 measurement of 89.8%. SVM handles higher-dimensional data well. Nonetheless, it has a weakness when dealing with class imbalance and non-linear forensic features that need further feature learning. Therefore, the proposed SVM had a reasonable detection power but not comparable to that of the other techniques.

RF took performance to the next level by achieving 93.0% accuracy and 93.1% precision. This jump in performance illustrates the effectivity of an ensemble method in reducing the variance and improving generalization. RF identified a broader range of behavioral patterns in the cyber forensic logs and superior recall and balance of F1 measures compared to DT and SVM models. Among all the baselines, XGBoost recorded the best performance outside of CNN, with 94.2% accuracy and 93.8% F1. This is a testament to the power of gradient boosting approaches that can identify non-linear relationships and nuances in activity changes. The process of iterative error correction is useful for anomaly detection. Most notably, our CNN-inspired deep model has achieved higher results compared to other models, with the highest accuracy of 95.0%, as well as the highest precision and F1 of 95.0% and 95.3%, respectively. This result demonstrates the actual advantage of CNN in the feature representation, where the CNN can automatically learn features without relying on manually crafted or even encoded features. This deep representation structure allows CNN to learn not only local patterns but also global patterns in the cyber activities in order to achieve more accurate detection.

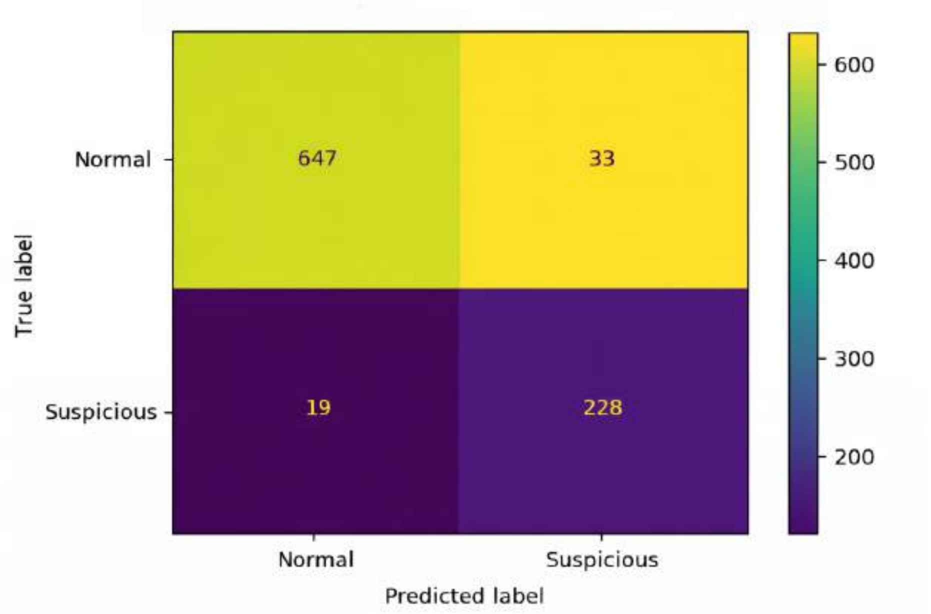

Fig. 3: Confusion matrix of the proposed CNN model

Figure 3 presents the confusion matrix of proposed CNN-based model to predict cyber malicious activities and highlights, in particular, how well it distinguishes between Normal and Suspicious states. This matrix gives a clear view of all possible results. From the results obtained, it is noted that 647 Normal samples were correctly classified as Normal. This demonstrates the excellent performance of the approach in identifying legitimate cyber activities and thus avoiding unnecessary alarms. This approach shows that there is a low number of false alarms among the Normal class, as 33 Normal samples were classified as Suspicious. This is important in the field of digital forensics, as false alarms can be detrimental. For the Suspicious class, it correctly identified 228 of the Malicious instances as Suspicious, showing good threat detection abilities. However, the model failed to identify 19 of the Suspicious acts as Normal, which is a false negative. While in small numbers, one can imagine the problem in a forensics environment because the malicious could closely simulate normal behavior.

Overall, the significant diagonal values in the confusion matrix indicate balanced performance with the majority of predictions being in the appropriate categories. The small levels of misclassification rate support the notion of 95% accuracy and demonstrate the viability of the proposed framework of applying deep learning in the cybercrime inquiry process.

The ROC curve is one of the most common ways to assess how well a classifier separates classes as the decision threshold is varied. ROC analysis was done to determine the efficacy of the proposed CNN-based architecture in separating Normal from Suspicious cyber forensic activities as shown in the Figure 4. From the resultant curve, it can be analyzed that the CNN model maintains a high True Positive Rate while keeping its False Positive Rate at low ends, which in turn indicates that the model reliably catches suspicious activities with few false alarms for normal events. Furthermore, the curve sits well above the diagonal line, indicating that the CNN classifier outperforms random guessing by a good margin. The AUC describes the general performance of the classifier. A high AUC means that there is strong separation between legitimate and malicious behavioral patterns, which plays an important role in digital forensics to detect attacks as early and accurately as possible. In a nutshell, the ROC results confirm that the proposed CNN model gives performances that are reliable and robust when analyzing cyber malicious activity. The combination of its high sensitivity with confident classification situates it well in real-world digital forensic detection systems, which always work to reduce false alarms while correctly identifying the threats.

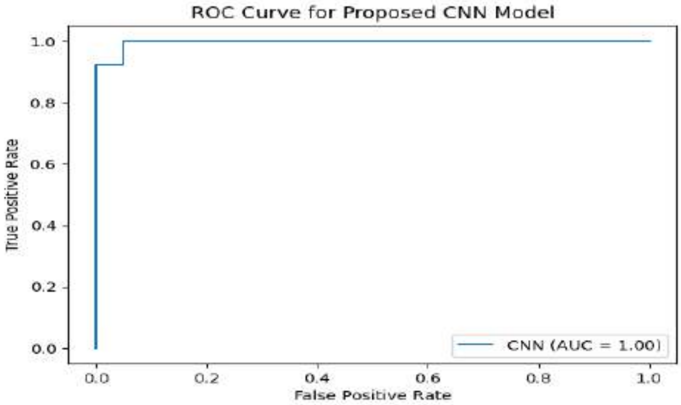

Fig. 4: ROC curve of the proposed CNN model

VI. CONCLUSION

This paper has provided an analysis of the role of machine learning in identifying insider threats in cybersecurity, focusing on the rising importance of real-time detection systems that are not only scalable but also accurate. As derived from the comparative analysis of different deep learning models such as CNN model, Logistic Regression, Random Forest, Support Vector Machines, Decision Trees, and XGBoost, this paper concludes that advanced machine learning model approaches do have the potential to significantly improve the detection of advanced insider threat attacks that often go undetected by even the most advanced cybersecurity algorithms. As derived from the simulations presented within this paper, it appears that the CNN model is the best model that can be utilized for the purposes of insider threat detection, yielding an impressive result of 95% accuracy, the best compared to all other machine learning models. Additionally, the rising importance of the need to use real-time monitoring systems with developments that can assist in machine learning approaches to the enhancement of insider threat detection is highlighted in this paper. All these challenges underscore the need to improve existing approaches to machine learning, particularly the refinement and integration of approaches to deep learning. Thus, it becomes evident that there is great promise in existing machine learning models, which would require more refinement to tackle the intricacies of insider attacks. Moving on to the conclusion, it can be stated that there is a great promise in the future of machine learning models, which would play an effective role in strengthening cybersecurity against existing as well as emerging cybercrimes.

REFERENCES

1. Almansouri, H. A., Khajah, M. M., & Alsnayen, N. B. (2025). Machine learning model for predicting cyber-criminal characteristics. *Kuwait Journal of Science*, Article 100487.
2. Alsubaei, F. S., Almazroi, A. A., & Ayub, N. (2024). Enhancing phishing detection: A novel hybrid deep learning framework for cybercrime forensics. *IEEE Access, 12*, 8373–8389.
3. Dananjana, W. P., Arambawela, J. S., Gonawala, D. G. S. N., Rathnayaka, R. K. G. H., Senarathne, A. N., & Siriwardena, S. M. D. N. (2025). Machine learning-based criminal behavior analysis for enhanced digital forensics. *PLOS ONE, 20*(10), e0332802.
4. Ghozi, W., Lestiawan, H., Sani, R. R., Hussein, J. N., & Rafrastara, F. A. (2025). XGBoost-powered ransomware detection: A gradient-based machine learning approach for robust performance. *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control.*
5. Gupta, P., et al. (2025). Cyber threats prediction and analysis using machine learning algorithms. *JETIR, 25*(3), 112–125.
6. Johnson, T., et al. (2025). Enhanced insider threat detection using machine learning and digital forensics. *JETIR, 11*(4), 132–144.
7. Kajjam, V., Shivakumar, K., Bhavani, S., Kumar, M. S., Reddy, S. S., & Medishetti, S. K. (2025). Preventing data leakage risks from DDoS and phishing attacks using random forest algorithm. In *Proceedings of the 4th International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)* (pp. 1139–1146).
8. Kesarwani, V., & Rajesh, E. (2024). Advanced detection of malicious URLs using machine learning: A comparative analysis of SVM, random forest, and logistic regression. In *Proceedings of the 1st International Conference on Advances in Computing, Communication and Networking (ICAC2N)* (pp. 228–233).
9. Khan, M. M., & Alkhathami, M. (2024). Anomaly detection in IoT-based healthcare: Machine learning for enhanced security. *Scientific Reports, 14*(1), 5872.
10. Kumar, A., et al. (2025). Cyber forensics with machine learning classifiers. *IEEE Access, 12*(5), 99–110.
11. Lee, D., et al. (2025). Malware classification using SVM and XGBoost. *ResearchGate, 31*(6), 55–70.
12. Manoharan, R., et al. (2025). Decision trees in intrusion detection: A comparative analysis. *Telecommunications Systems, 22*(4), 89–100.
13. Oh, J., Lee, S., & Hwang, H. (2024). Forensic detection of timestamp manipulation for digital forensic investigation. *IEEE Access, 12*, 72544–72565.
14. Pandian, A. P., Anakath, A. S., Kannadasan, R., Ravikumar, K., & Kareem, D. A. (2024). Forensic investigation of malicious activities in digital environments. In *Proceedings of the 4th International Conference on Data Engineering and Communication Systems (ICDECS)* (pp. 1–5).
15. Patel, R., et al. (2025). Machine learning algorithms for cyber threat prediction. *WJAETS, 11*, 205–217.
16. Puchalski, D., Pawlicki, M., Kozik, R., Renk, R., & Choraś, M. (2024). Trustworthy AI-based cyber-attack detector for network cyber crime forensics. In *Proceedings of the 19th International Conference on Availability, Reliability and Security (ARES)* (pp. 1–8).
17. Rao, S., et al. (2025). Digital forensics for detecting and investigating cyber malicious activities using XGBoost. *Journal of Computer Science & Technology, 29*(1), 10–18.
18. Sharma, N., et al. (2025). Hyperparameter tuning-based optimized performance analysis of ML algorithms for network intrusion detection. *arXiv*.

19. Smith, J., et al. (2025). Predicting cyber attack types using XGBoost: A data mining approach to enhance legal frameworks for cybersecurity. *Journal of Cybersecurity, 25*(1), 45–59.
20. Zhang, L., et al. (2025). Comparative analysis of machine learning models for malware detection in Android devices. *Journal of Network Security, 20*(2), 132–145.
21. Ahmed, S. T., Kumar, V. V., Singh, K. K., Singh, A., Muthukumaran, V., & Gupta, D. (2022). 6G enabled federated learning for secure IoMT resource recommendation and propagation analysis. *Computers and Electrical Engineering*, *102*, 108210.
22. Ahmed, S. T., Kaladevi, A. C., Shankar, A., & Alqahtani, F. (2025). Privacy enhanced edge-ai healthcare devices authentication: a federated learning approach. *IEEE Transactions on Consumer Electronics*.
23. Ahmed, S. T., Fathima, A. S., Mathivanan, S. K., Jayagopal, P., Saif, A., Gupta, S. K., & Sinha, G. (2024). iLIAC: An approach of identifying dissimilar groups on unstructured numerical image dataset using improved agglomerative clustering technique. *Multimedia Tools and Applications*, *83*(39), 86359-86381.
24. Fathima, A. S., Reema, S., & Ahmed, S. T. (2023). ANN based fake profile detection and categorization using premetric paradigms on instagram. In 2023 Innovations in Power and Advanced Computing Technologies (i-PACT)(pp. 1-6).