GATE-X: A Graph and Transformer-Based Framework for Proactive Data Exfiltration Detection via User-Entity Behavior Analysis

L. Lanuwabang[1]* and S. Suprakash[2]

Department of Information Technology, Kalasalingam Academy of Research and Education, Anand Nagar, Krishna Koil, 626126, India
[1]*lanuwabang.l@klu.ac.in , [2]s.suprakash@klu.ac.in

Abstract. Due to the modern world's development, protecting sensitive information from sophisticated threats such as data exfiltration has emerged as a critical concern. This paper presents GATE-X (Graph and Transformer-based Exfiltration Detection), a novel algorithm for the proactive detection of potential data exfiltration attacks through smart analysis of User Entity Behaviour Analysis (UEBA). The approach creates a dynamic bipartite graph where users and devices are linked, and behavioural patterns are described by computing graph features such as degree centrality and clustering coefficients. A Graph Neural Network (GNN) is trained in an unsupervised fashion to reconstruct graph features and identify refined anomalies. The neural network is combined with an Isolation Forest to compute anomaly scores of user-device interactions, and adaptive thresholding techniques improve the precision of distinguishing normal from suspicious interactions. A transformer-based BERT model further classifies labelled anomaly context as malicious or benign. The model was tested on 100 users and 50 devices, producing 1000 synthetic logs and attaining over 91% accuracy. The proposed strategy presents a scalable, intelligent, and effective method for early identification and mitigation of data exfiltration threats.

Keywords: Data exfiltration · UEBA · Graph Neural Networks · Isolation Forest · Transformer · BERT · Anomaly detection · Cybersecurity

1 Introduction

The growing need for digital infrastructure has increased the chances of cyber threats, particularly data exfiltration, which refers to sensitive information being taken from a secure system without permission. Cybercriminals and insider threats exploit various methods, including unauthorized file transfers, hidden data tunnels, and cloud storage leaks, to steal critical data. These attacks can lead to severe financial losses, legal consequences, and breaches of data security regulations [1].

Traditional security systems, such as firewalls, anomaly detection, and Intrusion Detection Systems (IDS), rely on rule-based mechanisms to detect threats [2]. However, this method is ineffective against increasing cyberattacks, especially when

M. Sharma et al. (eds.), *Proceedings of the Third International Conference on Recent Advances in Computing Sciences (RACS 2025)*, Advances in Intelligent Systems Research 212,
https://doi.org/10.2991/978-94-6239-768-2_7

attackers exploit authorized users. To overcome these limitations, User and Entity Behavior Analytics (UEBA) is emerging as a solution to enhance system security, offering a more adaptive and intelligence-driven approach to detecting anomalies and threats.

The UEBA framework continuously monitors and analyzes user and system activities to create behavioral baselines and identify anomalies that may suggest malicious intentions [3]. Unlike conventional security tools that focus on known attack patterns, UEBA uses anomaly detection techniques to identify anomalous activities even in the absence of prior knowledge about the attack. UEBA can detect internal threats, compromised user accounts, and complex attacks using various models and machine learning algorithms [4]. The primary challenge in the effective implementation of UEBA is distinguishing between legitimate activities and potential security vulnerabilities.

For instance, a system administrator's access to sensitive files is frequent. However, if the same administrator immediately transfers a substantial amount of data to a third-party location at an unusual time, or repeatedly attempts to access restricted files, it may indicate a data exfiltration attempt. Traditional security systems may generate a high volume of false alarms, making it challenging for security teams to identify actual risks. UEBA enhances detection accuracy by reducing false positives through correlation of anomalous behaviors with historical and peer-group activities [5].

Graph Neural Networks (GNNs) help identify unusual activities by uncovering hidden patterns in complex relational data [6]. As organizations generate massive amounts of interconnected data through user, device, and application interactions, conventional security techniques struggle to examine these complex relationships. GNNs transform security data into a structured graph representation where nodes represent entities and edges map their interactions, enabling detection of insider threats, coordinated attacks, and previously undetected anomalies.

Furthermore, self-supervised learning techniques enable the system to continuously refine its threat detection models over time [7]. By adapting to evolving attack patterns rather than fixed rules, self-supervised learning strengthens recognition accuracy and reduces false positives. Transformer-based models additionally improve the analysis of security alerts, enabling rapid and precise incident management.

Contributions of this Paper

- A UEBA framework that continuously monitors and analyzes user activities in real time to detect potential threats, including insider attacks and data exfiltration.
- A GNN-based approach that models user-system interactions as a graph structure, capturing complex relationships between users, devices, and applications.
- A self-supervised learning approach that allows the system to continuously refine detection parameters based on historical user behavior.
- Comparison of the proposed approach with DBN-OCSVM [8] and Matrix Profile [9] baselines.

2 Related Work

UEBA has become a crucial component of cybersecurity systems designed to detect insider threats and prevent data exfiltration. As attacks grow more sophisticated, researchers have increasingly focused on behavioral analytics and machine learning for advanced threat detection.

Rengarajan and Babu [10] visualized user behavior using a UEBA framework leveraging Tableau dashboards, demonstrating how behavioral deviations such as anomalous IP activity or unusual access times can be effectively classified. Their research emphasizes the importance of continuous monitoring with machine learning to dynamically identify anomalies.

Tian et al. [11] developed a multimodel-based system (MBS) for user activity analysis in urban big data settings. Employing Long Short-Term Memory (LSTM) networks to learn typical behavioral patterns and identify anomalies indicative of insider threats, the study demonstrates how deep learning can efficiently recognize periodic behavioral patterns for real-time complex anomaly detection.

Khan et al. [12] proposed a UEBA framework for enterprise networks, highlighting the need for scalable anomaly detection in systems without ground-truth labels. The described approach reduces high false-positive rates by providing transparency in user monitoring over time.

Yousef and Jazzar [13] explored the accuracy of UEBA tools specifically designed for countering insider threats, measuring defined parameter thresholds through detection system accuracy testing. Their results emphasize the need for UEBA systems with low and consistent error values alongside wide threat coverage.

Al-Bataineh and White [14] analyzed traffic patterns for data exfiltration detection, developing a classification algorithm based on entropy and byte frequency distribution of HTTP POST content, providing evidence for accurate detection of exfiltration attempts disguised within web traffic.

Steadman and Scott-Hayward developed DNSxD [16], a detection system focused on anomalous DNS query behavior using Software-Defined Networking (SDN) principles. The same authors subsequently proposed DNSxP [15], which adds Data Plane Programmability through eBPF and P4, supporting real-time DNS traffic classification while reducing both exfiltration availability and network resource consumption.

Collectively, these works reveal a trend towards more sophisticated behavioral analytics and traffic-aware detection systems, emphasizing the need for continuous monitoring, machine learning, and programmable networks.

3 System Model

This section presents the problem formulation with mathematical equations and the proposed GATE-X algorithm.

3.1 Problem Formulation

Data exfiltration is a significant cybersecurity threat where sensitive data is illicitly transferred outside an organization's network. The proposed multi-layered detection approach integrates GNNs, Isolation Forest anomaly detection, and transformer-based log classification. The detection pipeline consists of data collection, preprocessing, graph-based feature extraction, anomaly detection, and classification.

3.2 Data Collection and Preprocessing.

The first step in the detection process collects logs capturing user-device interactions, represented by:

$$L = \{ (u_i, d_i, f_i) \mid i = 1, 2, ..., N \}$$

where u_i represents the user ID, d_i the device ID, f_i the access frequency, and N the total number of log entries. To ensure consistency in numerical features, standardization is applied to normalize access frequency:

$$f'_i = (f_i - \mu f) / \sigma f \quad (1)$$

where μf and σf are the mean and standard deviation of access frequency.

3.3 Graph Construction.

The collected data is structured into a bipartite graph G = (V, E), where nodes represent users and devices, and edges capture access relationships weighted by access frequency w_i. Degree centrality and clustering coefficient are computed as:

$$Cdn = degn / (|V| - 1) \quad (2)$$

$$Ccn = 2Tn / (kn(kn - 1)) \quad (3)$$

where Tn denotes the number of triangles associated with node n, and kn the number of direct connections of node n. Each node is then represented using the feature vector:

$$Xn = [Cdn, Ccn] \quad (4)$$

which serves as input for subsequent anomaly detection methods.

3.4 GNN-Based Anomaly Detection.

A two-layer GNN model performs representation learning using graph-based features. Node embeddings are computed as:

$$H^1 = ReLU(W_1X + b_1) \quad (5)$$

$$H^2 = W_2H^1 + b_2 \quad (6)$$

where W_1, W_2 are learnable weight matrices and b_1, b_2 are bias terms. ReLU (Rectified Linear Unit) is defined as ReLU(x) = max(0, x). The training objective minimizes reconstruction loss:

$$L = \|H_i^2 - X_i\|^2 \tag{7}$$

3.5 Isolation Forest-Based Anomaly Detection.

An Isolation Forest model computes an anomaly score S(x) based on the expected isolation depth of a sample:

$$S(x) = 2\text{^}(-E[h(x)]/c(n)) \tag{8}$$

where h(x) is the average isolation depth of x and c(n) is the expected path length. A threshold T is determined to classify anomalies:

$$T = P_{95}(S) \tag{9}$$

where P_{95} denotes the 95th percentile of scores. The decision rule is:

y = 1 if S(x) < T (anomaly), 0 otherwise (normal) (10)

3.6 Transformer-Based Log Classification.

Logs are classified using a transformer-based model. For each log entry t_i, embeddings are computed using BERT:

$$E(t_i) = BERT(t_i) \tag{11}$$

The final classification function is:

$$\hat{y}cls = \sigma(W \cdot E(t_i) + b) \tag{12}$$

where W and b are trainable parameters.

3.7 Evaluation Metrics.

Detection effectiveness is assessed using a confusion matrix comprising True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) values. Performance metrics are computed as:

$$Accuracy = (TP + TN) / (TP + TN + FP + FN) \tag{13}$$

$$Precision = TP / (TP + FP) \tag{14}$$

$$Recall = TP / (TP + FN) \tag{15}$$

$$F1 = 2 \times (Precision \times Recall) / (Precision + Recall) \tag{16}$$

The ROC curve is plotted using the True Positive Rate (TPR) and False Positive Rate (FPR):

$$TPR = TP / (TP + FN) \tag{17}$$

$$FPR = FP / (FP + TN) \tag{18}$$

The Area Under the Curve (AUC) is computed as:

$$AUC = \int_0^1 TPR(FPR)\, d(FPR) \quad (19)$$

3.8 Proposed Algorithm

Algorithm: GATE-X (Graph and Transformer-based Exfiltration Detection)

Input: Log data entries (User, Device, Frequency)

Output: Anomaly classification and ROC evaluation

1. Initialize (u_i, d_i) and generate (u_i, d_i, f_i).
2. Initialize graph G. For each entry (u_i, d_i, f_i): add edge (u_i, d_i) to G with weight f_i; compute Cdn and Ccn.
3. Construct node feature matrix F using Cdn and Ccn (Eq. 4).
4. Initialize Simple GNN (input dim = 2, hidden dim = 8). Train with MSE loss and Adam optimizer (lr = 0.01, epochs = 50).
5. Train Isolation Forest (contamination c = 0.01) on standardized logs. Compute anomaly scores (Eq. 8) and threshold T (Eq. 9). Classify each entry by Eq. 10.
6. Compute BERT embeddings (Eq. 11). Classify each embedding as Malicious if mean output exceeds predefined threshold; otherwise Benign (Eq. 12).
7. Compute Confusion Matrix. Calculate F1 score (Eq. 16) and ROC/AUC (Eq. 17–19). Plot Confusion Matrix and ROC Curve.

4 Experimental Study

This section discusses the experimental setup, simulation environment, and performance results obtained from the execution of the proposed algorithm.

4.1 Experimental Setup

Experiments were conducted on a MacBook Air (Midnight) equipped with an Apple M3 chip (8-core CPU, 10-core GPU, 16-core Neural Engine), 16 GB unified memory, and 256 GB SSD storage. The algorithm was implemented in Python 3.12. Simulation parameters are listed in Table 1.

Table 1. System Specifications and Hyperparameters

Total Entries	1000 logs
Number of Users	100 Users
Number of Devices	50 Devices
Features per Entry	3 (User, Device, Frequency)
Hidden Layer Dimension	8
Activation Function	ReLU
Optimizer	Adam

Learning Rate (lr)	0.01
Epochs	50

4.2 Results and Discussion

This section evaluates the effectiveness of the proposed approach in detecting data exfiltration. The framework was tested on user-device interaction logs, and performance was assessed using multiple evaluation metrics. By combining GNNs for feature extraction, Isolation Forest for anomaly detection, and transformer-based log classification, the approach enhances detection accuracy while minimizing false positives.

4.2.1 Confusion Matrix Analysis.

The confusion matrix evaluates the performance of the GNN in classifying data into Normal and Abnormal categories. Accuracy is calculated using Eq. 13, precision by Eq. 14, recall by Eq. 15, and the F1 score by Eq. 16. By analyzing the confusion matrix, strengths of the model are identified alongside areas requiring further enhancement. The confusion matrix for the proposed method is shown in Fig. 1.

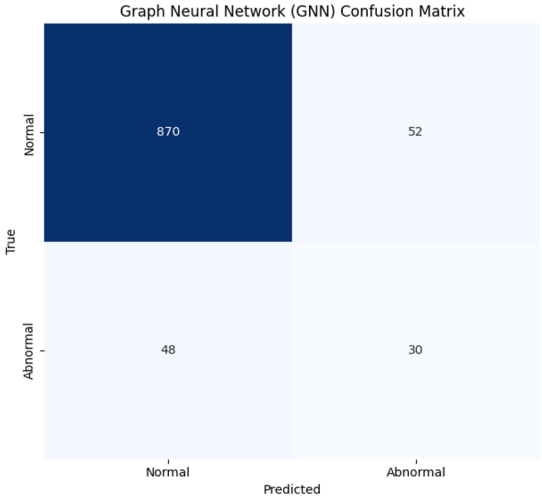

Fig. 1. Confusion Matrix

4.2.2 Performance Comparison.

The proposed approach shows significant improvement over existing techniques in both accuracy and precision. It achieves 3.65% higher accuracy than DBN-OCSVM and 8.46% higher accuracy than the Matrix-Profile, demonstrating its ability to classify normal and abnormal activities more effectively. In terms of precision, the proposed method outperforms DBN-OCSVM by 11.6% and Matrix-Profile by 29.9%, reducing false positives and improving threat detection. These results confirm that the proposed approach is more reliable in identifying exfiltration attempts while maintaining a lower false positive rate. A comparison of accuracy and precision across approaches is provided in Table 2.

Table 2. Result Comparison Across Approaches

Approach	Accuracy	Precision
DBN-OCSVM	**87.79%**	**32.79%**
Matrix-Profile	**83.9%**	**28.18%**
Proposed Approach	**91%**	**36.6%**

4.2.3 ROC Curve Analysis.

The proposed approach demonstrates superior detection accuracy while keeping false positives low. Its position on the ROC curve confirms that it identifies exfiltration attempts more effectively than the two comparison methods. Although all three models exhibit similar trends, the higher AUC of the proposed method confirms its improved performance. By integrating GNN, Isolation Forest, and transformer-based classification, this approach enhances detection reliability. The ROC curve analysis for the proposed and compared algorithms is shown in Fig. 2.

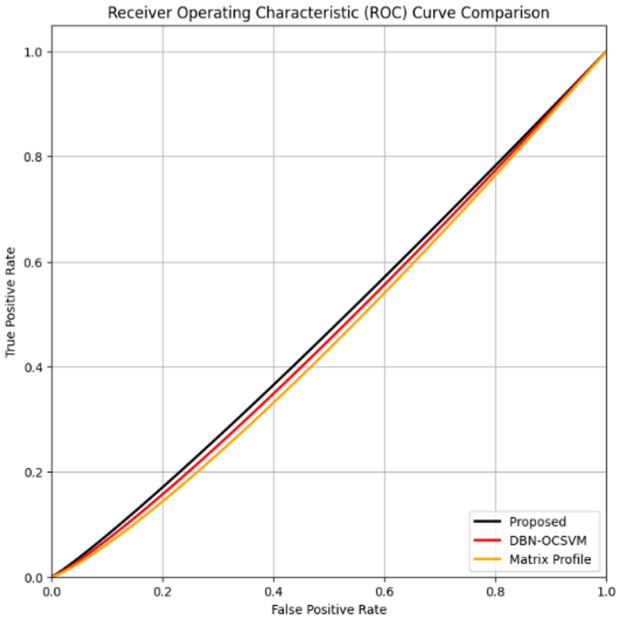

Fig. 2. ROC Curve Analysis

5 Conclusion

This study introduces GATE-X (Graph and Transformer-based Exfiltration Detection), a framework designed to detect data exfiltration threats in user-device interaction ecosystems for UEBA. The framework constructs a user-device graph and, through node centrality and clustering metrics, implements a GNN to capture behavioral patterns. An Isolation Forest model identifies behavioral anomalies, while a transformer language model classifies suspicious activities based on log-file semantic analysis. Experimental evaluation using synthetic data from 100 users and 50 devices demonstrates over 91% classification accuracy, a balanced confusion matrix, and a low false positive rate. The combination of graph and language models augments the scalability and comprehensiveness of the real-time anomaly detection solution.

In future work, the framework will be tested on operational networks and optimized for resource-limited edge computing environments.

References

1. Yuan, S., Wu, X.: Deep learning for insider threat detection: review, challenges and opportunities. *Computers & Security* 104, 102221 (2021)
2. Detection, I.: Using machine learning algorithms in intrusion detection systems: a review. Tikrit Journal of Pure Science 29(3) (2024)
3. Sharma, G., Thakur, A., Tiwari, C.: Developing a comprehensive framework for user and entity behavior analytics (UEBA): integrating advanced machine learning and contextual insights. Journal of 14(2) (2024)
4. Datta, J., et al.: Real-time threat detection in UEBA using unsupervised learning algorithms. In: 5th International Conference on Electronics, Materials Engineering & Nano-Technology (*IEMENTech*), pp. 1–6. IEEE (2021)
5. Deepa, S., et al.: Deep belief network-based user and entity behavior analytics (UEBA) for web applications. *International Journal of Cooperative Information Systems* 33(02), 2350016 (2024)
6. Altaf, T., et al.: GNN-based network traffic analysis for the detection of sequential attacks in IoT. *Electronics* 13(12), 2274 (2024)
7. Mohanty, R.K., et al.: Deep learning for analyzing user and entity behaviors: techniques and applications. In: Consumer and Organizational Behavior in the Age of AI, pp. 219–250. IGI Global (2024)
8. Lin, L., et al.: Insider threat detection based on deep belief network feature representation. In: *International Conference on Green Informatics (ICGI)*, pp. 54–59 (2017)
9. Khan, M.Z.A., Khan, M.M., Arshad, J.: Anomaly detection and enterprise security using user and entity behavior analytics (UEBA). In: 3rd International Conference on Innovations in Computer Science & Software Engineering (*ICONICS*). IEEE (2022)
10. Rengarajan, R., Babu, S.: Anomaly detection using user entity behavior analytics and data visualization. Amrita Vishwa Vidyapeetham, Bangalore (2022)
11. Tian, Z., et al.: User and entity behavior analysis under urban big data. *ACM Transactions on Data Science* 1(3), 1–19 (2020)
12. Khan, M.Z.A., Khan, M.M., Arshad, J.: Anomaly detection and enterprise security using UEBA. In: 3rd International Conference on Innovations in Computer Science & Software Engineering (ICONICS). IEEE (2022)
13. Yousef, R., Jazzar, M.: Measuring the effectiveness of user and entity behavior analytics for the prevention of insider threats. J. Xi'an Univ. Arch. Technol. 13, 175–181 (2021)
14. Al-Bataineh, A., White, G.: Analysis and detection of malicious data exfiltration in web traffic. In: 7th International Conference on Malicious and Unwanted Software, pp. 1–6. IEEE (2012)
15. Steadman, J., Scott-Hayward, S.: DNSxP: enhancing data exfiltration protection through data plane programmability. *Computer Networks* 195, 108174 (2021)
16. Steadman, J., Scott-Hayward, S.: DNSxD: detecting data exfiltration over DNS. In: IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN). IEEE (2018)