Journal of Applied Automation Technologies

Research Paper

A Security Analysis Method for Access Control Graphs Based on DeepWalk Node Embedding

Cristian Radu[1], Elena Constantin[1] and Maria Dragomir[1,*]

[1] Faculty of Automation and Computer Science, Politehnica University of Timișoara, 300006 Timișoara, Romania
*Corresponding author: maria.d@aut.upt.ro

Abstract. In this article, we will discuss how large-scale dynamic information systems can detect access control anomalies. An automatic security analysis based on the DeepWalk node embedding framework is proposed for enterprise-level access control graphs. In the extraction of the graph structure, random walks map the relationships between explicit and implicit users, resources, and permissions into dense vectors. By using the aforementioned embeddings for unsupervised anomaly scoring and permission path risk assessment, the system can effectively identify configuration errors and suspicious access patterns. For the experiment, a large-scale dataset was obtained from multiple commercial cloud platforms in fields such as healthcare and banking, containing over 100,000 accounts and more than 1 million audit objects. Based on the above results, the three models for medical, financial, and cloud scenarios achieved AUC values of 0.96, 0.95, and 0.94, respectively. Compared to previous methods, the current detection time averages 2 hours, and the false positive rate has also been reduced by 60%. Ablation and cross-domain robustness studies indicate that the time module and embedding module are crucial for the accuracy and stability of system detection. It is an easy-to-understand real-time analysis method that can be used for enterprise security processes and SIEM platforms. In summary, the aforementioned framework enhances automated access governance and risk monitoring, providing an effective and intelligent foundation for large-scale permission management and policy compliance analysis in modern digital infrastructure.

Keywords: *Computer Security, Access Control, Graph Embedding, DeepWalk, Anomaly Detection, Privilege Analysis, Enterprise Security, Cloud Computing*

Received on 10 October 2025, Accepted on 18 March 2026, Published on 31 March 2026

Introduction

Establishing access controls for important data and services in information systems, such as medical records and financial data. Due to the application of multi-tenant cloud models and remote work, access control issues have become more complex. By controlling system access, protecting personal privacy in financial and healthcare services, and reducing the risk of enterprise-level data breaches, etc. HIPAA and GDPR regulations have not yet come into effect. Modern enterprises with hybrid infrastructures now require highly scalable and flexible management solutions to address new security threats [1]. Currently, risk management and operational assurance are the main tasks of access control [2]. With the occurrence of data breaches, people have seen the consequences of lacking access control. Therefore, researchers and practitioners have proposed more robust and flexible control models [3]. With the widespread adoption of data-driven systems and automation, access control methods also need to be optimized to adapt to emerging network threats [4]. Research on graph-based security analysis has recently demonstrated great success in showcasing complex relationships and dynamic user behavior [5]. A universal and reliable automatic anomaly detection system is needed [6]. With the development of a digital society, many trust issues have also emerged [7]. The demand for smart access control has already increased [8].

With the development of technology, traditional access control models have become increasingly ineffective today. Due to the emergence of context-aware, attribute-based, and fine-grained policies, traditional rule-based and role-based models can no longer adapt to the constantly changing enterprise structure [9]. The large number

of temporary delegations, privilege chains, and permissions generated in federated systems and cross-domain collaborations cannot be handled by manual audits or static analysis [10]. Attackers can exploit undocumented or hidden relationships in the large-scale access graph to gain permissions or bypass rules [11]. In large enterprises, manual checks are not feasible because millions of dynamic, federated, or cloud-native system connections need to be monitored [12]. Automated solutions cannot capture dynamic or indirect flows because they involve a high degree of combinatorial complexity [13]. Despite some improvements in logical reasoning and policy analysis, they have not yet achieved scalability and automation [14]. Over time, researchers have begun using graph-based enterprise permission models to more effectively identify irregular and high-risk access paths [15]. Graph embedding and deep learning methods can enhance anomaly detection capabilities and encode complex relationships, but they also have issues with robustness and domain adaptability [16]. Secure operations require precise, adaptable, and interpretable solutions that can be integrated with enterprise systems [17]. There is still a need for scalable approaches to provide accessible access anomaly detection for employes [18].

Therefore, this paper proposes a DeepWalk-based node embedding framework to achieve this through automated fault detection in large-scale access control graphs. By using neural embeddings and random walk sampling to obtain explicit and implicit access relationships, and modeling higher-order proximity to detect configuration errors and potential vulnerabilities. This pipeline has excellent high-speed analysis and continuous monitoring capabilities, meeting the company's needs for security systems. We have already found that it is very effective for real and synthetic datasets in fields such as cloud computing, finance, and healthcare, improving the robustness, accuracy, and interpretability of detection. Enhance machine learning analysis for access control and provide practical support for the management of current infrastructure security policies. Finally, the aforementioned methods can help establish a more reliable and stable digital system.

Theoretical Foundations

Access Control Graph Models

Due to the numerous nodes and edges in digital systems, graphs can be used to describe access control. Typical examples include collaborative cloud suites, enterprise identity management, and electronic health records [19]. These dynamic heterogeneous networks connect users, resources, policies, and contexts. Role-Based Access Control (RBAC) and its extended models are becoming increasingly popular. Although the classic access control matrix is theoretically sound, it is not feasible in modern large-scale distributed environments. To support the separation of duties and organizational hierarchy required for compliance and auditing, the RBAC model maps users to roles and roles to permissions [20]. Attribute-based access control (ABAC) also includes context and environmental rules, and generates graphs of multi-attribute relationships [21].

The problem is too large to be directly solved in typical real-world systems. For example, cloud and SaaS platforms can use delegated permissions, shared folders, or API tokens to assign permissions, thereby creating temporary or cascading paths in the underlying access control graph [22]. By exploiting these topologies, attackers can bypass restrictions or gain elevation through chained exploitation of seemingly harmless permissions [23]. Some examples indicate that trivial errors, or the lack of proper authorization and excessive trust in inheritance, can be exploited to illegally access sensitive data [24]. Manual review is not feasible due to the excessive number of nodes and edges. Therefore, automated graph analysis methods have been used in enterprise-level security engineering, such as graph traversal to enhance detection of permissions, reachability queries, and cyclic detection of trust abuse [25]. Edges and nodes in the access control graph model represent permissions, delegation, and context checks to support policy reasoning and automated threat analysis [26].

Graph Embedding Algorithms

In practical applications, access control graphs are usually very large and complex; therefore, an automatic method is needed that does not require explicit rule checking. Now, graph embedding methods are a powerful technique that can convert nodes and their surrounding environments into dense vector representations, which preserve the local structure and global topology of the graph [27]. The aforementioned embeddings are used to achieve large-scale automated analysis of access control data, such as identifying anomalous connections, grouping similar users, and inferring covert policy violations in continuous space [28]. In security analysis,

DeepWalk, node2vec, and GraphSAGE are frequently used algorithms in this field for learning rich semantic and structural representations [29].

Graph embeddings make it easier to integrate access control analysis into machine learning-driven security pipelines. Embeddings can be obtained from data and used for anomaly detection and risk assessment. It does not require manual feature engineering or solely relying on policy logic [30]. They are particularly suitable for dynamic environments, such as federated environments with a large number of new users, assets, or rules, where fixed policy audits cannot keep up with these changes. These algorithms can identify multi-stage, fine-grained permission chains and detect communities or anomalies that manual checks might overlook. In subsequent stages, the embedding is relatively simple and can be used for integration with compliance reports and threat intelligence, among other things. Graph embeddings help enhance the intelligence and adaptability of access control systems in the complex real world.

Methodology

Problem Definition and Pipeline Overview

The first goal of this paper is to build a large-scale, always-available system capable of discovering security vulnerabilities in the vast access control graphs constructed in enterprise and cloud environments. This type of system should be able to adapt to frequent permission changes, organizational restructuring, and the complex interdependencies between policies, resources, and users.

At each time step t, the access control graph is represented as $G_t = (V_t, E_t)$. The node set $V_t = \{v_i : i = 1,2, \dots, N_t\}$ includes all users, resources, roles or application agents currently in the system. $E_t = \{(v_i, v_j,$ type, $w_{ij}) : v_i, v_j \in V_t\}$ is an executable connection, where each directed or labeled edge contains a semantic type, such as granting, delegating, or inheriting, as well as an optional weight w_{ij}, indicating frequency or degree of importance.

At $t+1$, after receiving the next audit log batch $L_{t+1} = \left\{l_1^{(t+1)}, l_2^{(t+1)}, \dots, l_{m_{t+1}}^{(t+1)}\right\}$, the framework parses and normalizes all the corresponding fields, and then calls the update operation:

$$G_{t+1} = \mathcal{U}(G_t, L_{t+1}) \quad \text{Eq.(1)}$$

The function $\mathcal{U}$ handles tracking insertions (e.g., granting new permissions), deletions (e.g., revoking or expiring rights), and recalibration of edge weights, and thus the graph representation always reflects the current state of the system.

Node v has been given a feature vector x_v containing local graph structure and context attributes for quantitative analysis:

$$x_v = f(\{\phi(e) : e \in \mathcal{E}_v\}, \psi(v)) \quad \text{Eq.(2)}$$

The function $\phi(e)$ extracts properties from all incident edges $\mathcal{E}_v$ linked to v, while the context transforms $\psi(v)$ captures auxiliary attributes such as device types, access times, or authentication methods relevant for risk assessment.

As shown in Figure 1, the entire process first aggregates events for preprocessing and normalization, then constructs a comprehensive cross-domain access control graph, and this process is almost real-time. Afterward, the embeddings based on DeepWalk transform structural patterns into node representations rich in information. These embeddings are used by downstream analysis components to identify suspicious chains, privilege escalation, or rare access patterns. The results are displayed as alerts and detailed context for use by the organization's SIEM or compliance tools. The aforementioned workflow can be supported by a modular structure, and additional features such as security analysis and graphical management can be added later.

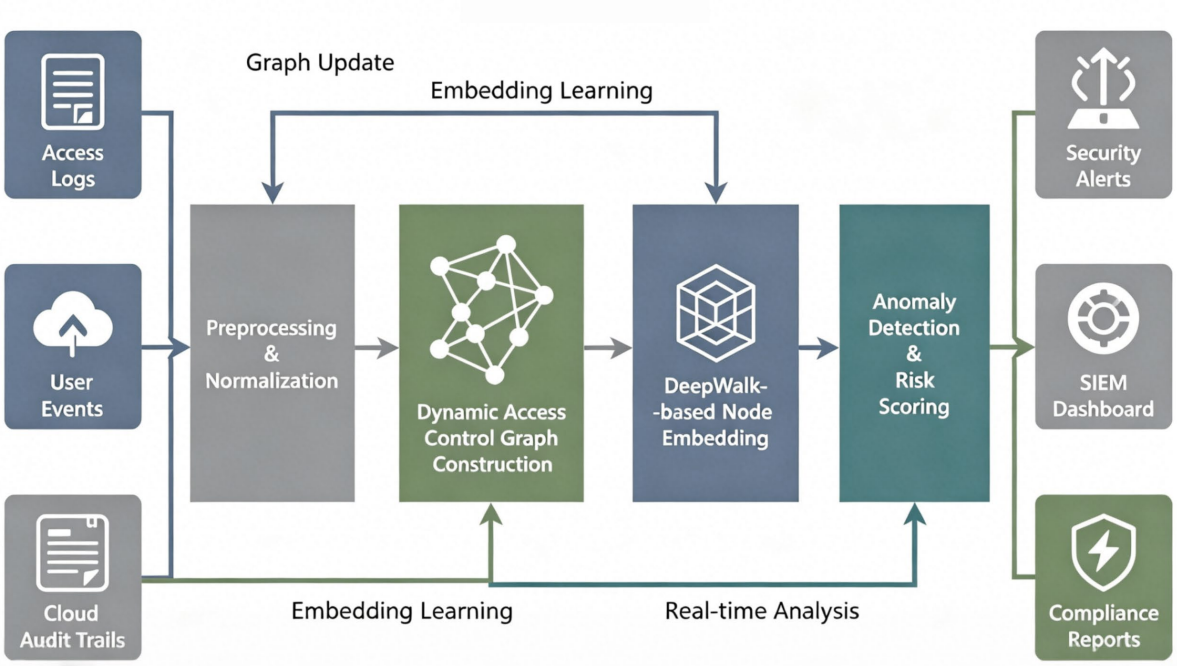

Figure 1. End-to-end workflow of automated access control analysis, illustrating each phase from event ingestion to analytical output

DeepWalk-based Node Embedding

The beginning of this method is a node embedding process for large-scale, dynamic access control graphs. DeepWalk is chosen due to its ability to process heterogeneous topologies, adaptability, and effectiveness at encoding both fine- and coarse-grained structures in large graphs.

DeepWalk interprets the access graph as a stochastic space, simulating truncated random walks from each node and assembling multiple walk sequences. Each sequence encodes the local and global context of the starting node, capturing both direct and transitive permissions as well as delegation and inheritance relationships. For node $v \in V$, DeepWalk produces sequences

$$S_v = \{v_1, v_2, \ldots, v_l\} \quad \text{Eq.(3)}$$

with v_i sampled according to edge probabilities, where the transition probability for a random walk from node v_i to node v_{i+1} is given by:

$$P(v_{i+1} \mid v_i) = \frac{w(v_i, v_{i+1})}{\sum_{u \in \mathcal{N}(v_i)} w(v_i, u)} \quad \text{Eq.(4)}$$

where $w(v_i, v_{i+1})$ is the weight of edge (v_i, v_{i+1}).

The objective is to maximize the likelihood of context nodes in these walks, formalized as:

$$\max_{\theta} \sum_{v \in V} \sum_{u \in Context(v)} log\ P(u \mid v; \theta) \quad \text{Eq.(5)}$$

where $P(u \mid v; \theta)$ is defined via a skip-gram neural network with parameters θ.

The learned embedding for node v, denoted as $\mathbf{z}_v \in \mathbb{R}^d$, can be utilized for downstream tasks, where the embedding update step follows:

$$\boldsymbol{z}_v^{t+1} = \boldsymbol{z}_v^t + \eta \cdot \nabla_{\boldsymbol{z}_v} \mathcal{L} \quad \text{Eq.(6)}$$

where η is the learning rate and $\mathcal{L}$ is the loss for the skip-gram structure.

Figure 2 shows the deep embedding mechanism used for access control. Randomly sample the various permission relationships around the node and map their structural environment to a small vector. These examples effectively summarize direct and indirect permissions to demonstrate how risks or abnormal access propagate through the chain.

The scalability of DeepWalk is one of its engineering advantages. This means it is highly parallel during training and can be used in distributed computing to manage large organizational graphs with millions of edges and hundreds of thousands of nodes. Parameters such as step size, context window, and embedding dimensions can be adjusted based on empirical properties like graph density or community structure to perform well in different environments. In addition, the model supports incremental training to quickly adapt to new visits or topological changes, and it can be used in environments with high real-time requirements.

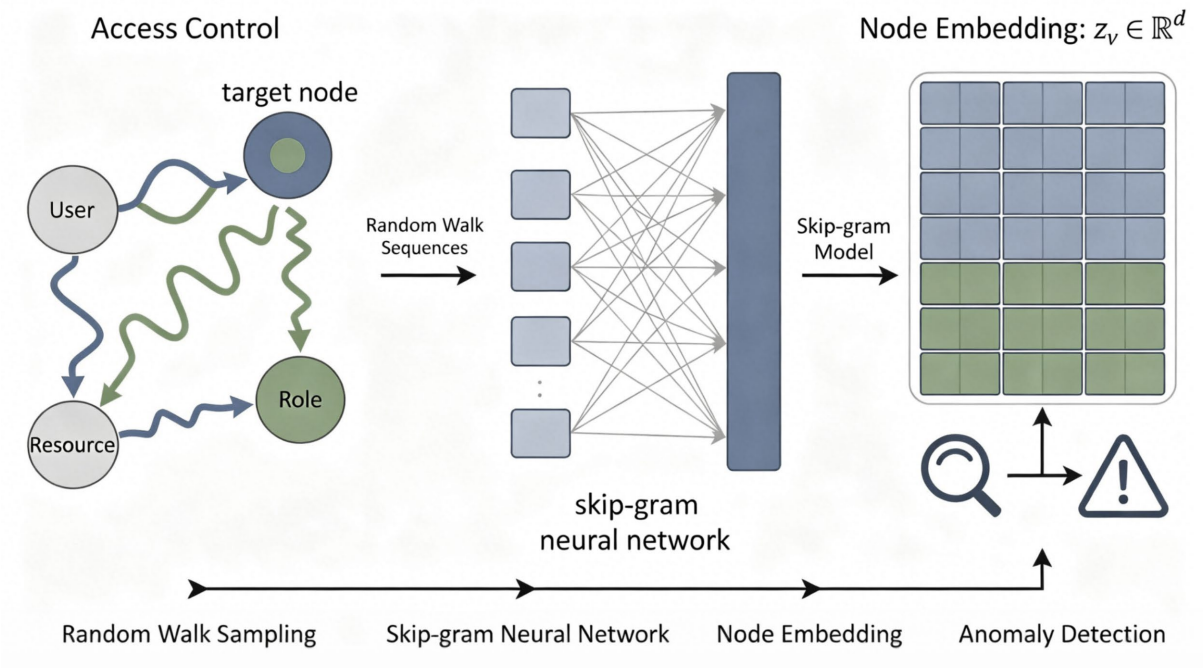

Figure 2. DeepWalk-based node embedding for access control graphs, illustrating random walks and the neural embedding process for encoding structural context

Security Flaw Detection Framework

The framework for detecting security vulnerabilities introduced in this article can convert complex access relationships and permission modifications into quantifiable and actionable risk intelligence for the operation of enterprise security. A dynamic directed graph $G_t = (V_t, E_t)$ represents the entire system, where its nodes V_t include users, resources, and roles, and the types of edges E_t include "source," "target," "relationship type," and "weight." In addition, the edges of E_t include "resources," "users," and "relationship types."

$$E_t = \{(v_i, v_j, type, w_{ij}) \mid v_i, v_j \in V_t\} \quad \text{Eq.(7)}$$

With every arrival of new log events $L_{t+1} = \{l_1^{(t+1)}, \ldots, l_{m_{t+1}}^{(t+1)}\}$, the system incrementally updates the access graph:

$$G_{t+1} = \mathcal{U}(G_t, L_{t+1}) \quad \text{Eq.(8)}$$

A feature vector of a node is obtained by aggregating structural links and contextual information:

$$x_v = f(\{\phi(e) \mid e \in \mathcal{E}_v\}, \psi(v)) \quad \text{Eq.(9)}$$

where $\phi(e)$ is edge feature extraction and $\psi(v)$ are node attributes. Aggregate all the nodes to obtain the time-specific feature matrix.

$$X_t = [x_{v_1}, \ldots, x_{v_n}]^T \quad \text{Eq.(10)}$$

Then, these vectors are transferred to an unsupervised embedding model (such as DeepWalk) and learn latent representations through random walks. The following are the probability settings for the random walk steps:

$$P(v_{i+1} \mid v_i) = \frac{w(v_i, v_{i+1})}{\sum_{u \in \mathcal{N}(v_i)} w(v_i, u)} \quad \text{Eq.(11)}$$

The optimization of node representations is achieved by using the skip-gram model objective function to optimize the structural context. After embedding the vectors, according to

$$\boldsymbol{z}_v^{t+1} = \boldsymbol{z}_v^t + \eta \cdot \nabla_{\boldsymbol{z}_v} \mathcal{L} \quad \text{Eq.(12)}$$

where η is the learning rate.

Security anomaly analysis compares the current embedding $\mathbf{z}_v^t$ of a node with its historical mean and covariance. The anomaly score is as follows:

$$AnomalyScore_v^t = \sqrt{(\boldsymbol{z}_v^t - \mu_v)^\top \Sigma_v^{-1} (\boldsymbol{z}_v^t - \mu_v)} \quad \text{Eq.(13)}$$

Nodes exceeding an adaptive threshold τ are marked for analyst review or automatic response. Chain risk assessment for a privilege path $\pi_{s,t}$ is also performed:

$$Risk(s,t) = \sum_{(v_i, v_{i+1}) \in \pi_{s,t}} w_{i,i+1} \cdot AnomalyScore_{v_{i+1}}^t \quad \text{Eq.(14)}$$

The structure of the framework has been modified to include the necessary subgraphs and determine whether these subgraphs exist in the dynamic access graph. The aforementioned architecture will be able to convert raw event access data into high-quality, high-resolution security data, supporting compliance enforcement and comprehensive audits, and detecting anomalies and monitoring permission drift in real-time.

Experimental Studies

Experimental Setup and Baselines

In this section, we will introduce the experimental environment and test the proposed detection framework in information security settings under various data and threat conditions. The experiments used real industrial datasets from actively managed production systems and created large-scale synthetic graphs to simulate enterprise-level access control scenarios. Both methods can help us gain a deeper understanding of the theoretical foundations and applications.

The regional healthcare information platform, the national online banking system, and the commercial multi-tenant cloud data management platform will be selected as the three typical secure environments for validation. The medical data center recorded access logs and permission allocations for nearly 15,000 active employes and 200,000 different electronic medical record (EMR) resources over a six-month period. For example, multi-level role-based access control, time-based delegation, and extremely sensitive data flows are typical features of compliance-driven, regulated business environments that require high levels of patient privacy protection.

Collected over 1 million employe accounts and over 1 million different transaction and audit objects from two primary data centers and secondary data centers. The red team added some real attack behaviors, such as privilege escalation and lateral movement, to provide high-fidelity real data for anomaly detection. Since the cloud platform data simulates over 50,000 tenants, each with dynamic roles and interconnected resources, it resembles a heterogeneous and highly dynamic cyber-physical system. The aforementioned operational environment tests the transferability and robustness under adversarial and non-stationary conditions.

By using power-law and community models to create synthetic large-scale evaluation graphs, we aim to match the in-degree and out-degree distributions found during the production process. It is feasible to conduct ablation studies and scalability tests on this model, which has approximately 100 million edges and over 1 million nodes. Burstiness, regular cycles, and sudden permission upgrades are access patterns of synthetic event logs based on the statistical characteristics of real-world distributions.

In all experiments, the proposed method was compared with top benchmark models: (a) unsupervised Node2Vec-based graph anomaly detection, (b) supervised random forest model trained on engineered access features, (c) deep graph neural network with explicit role channel encoding, and (d) standard rule monitoring commonly found in commercial SIEM systems. Baseline hyperparameters can be optimized through grid search or set to recommended production values where applicable. It will be compared to determine its applicability in operation, as well as other issues such as detection accuracy, scalability, and compatibility with existing enterprise or cloud-native security intelligence workflows.

Evaluation Metrics and Quantitative Results

In order to conduct comprehensive testing of the proposed security vulnerability detection system, many experiments were carried out in both real and artificial environments. In addition, it is necessary to use various evaluation metrics and charts to assess the practical application of these models.

Precision (P), recall (R), F1-score, area under the receiver operating characteristic curve (AUC), and mean time to detect (MTTD) are the first set of performance metrics in this study. Since the access anomaly dataset is usually imbalanced, special attention is given to false positives (FP) and false negatives (FN). In addition, the receiver operating characteristic (ROC) curve and precision-recall (PR) curve were calculated for each method and scenario. The throughput of the chart (i.e., the number of events per second) and memory overhead increase with the size of the chart.

Figure 3 shows the main performance results and uses multiple charts to demonstrate the advantages of our method. As shown in Figure 3(a), our method achieved the highest AUC and F1-score on the medical, financial, and cloud datasets. The corresponding F1 scores are 0.89, 0.85, and 0.83, and the AUC values are 0.96, 0.95, and 0.94, respectively. The box plot is used to show the distribution of average detection times, as shown in Figure 3(b). The median detection time of our method is approximately 2.0 hours, while the detection time of the baseline method exceeds 3.0 hours. Throughput scalability is shown in Figure 3(c), where the line graph indicates that our framework maintains a relatively high event processing rate of 11,000 events per second with one million edges, and a processing rate of 7,200 events per second with one hundred million edges, surpassing the baseline across all graph scales. Figure 3(d) is a radar chart that shows the AUC, F1-score, normalized inverse MTTD, and relative throughput of all methods in the comprehensive comparison. This indicates that our method performs well overall.

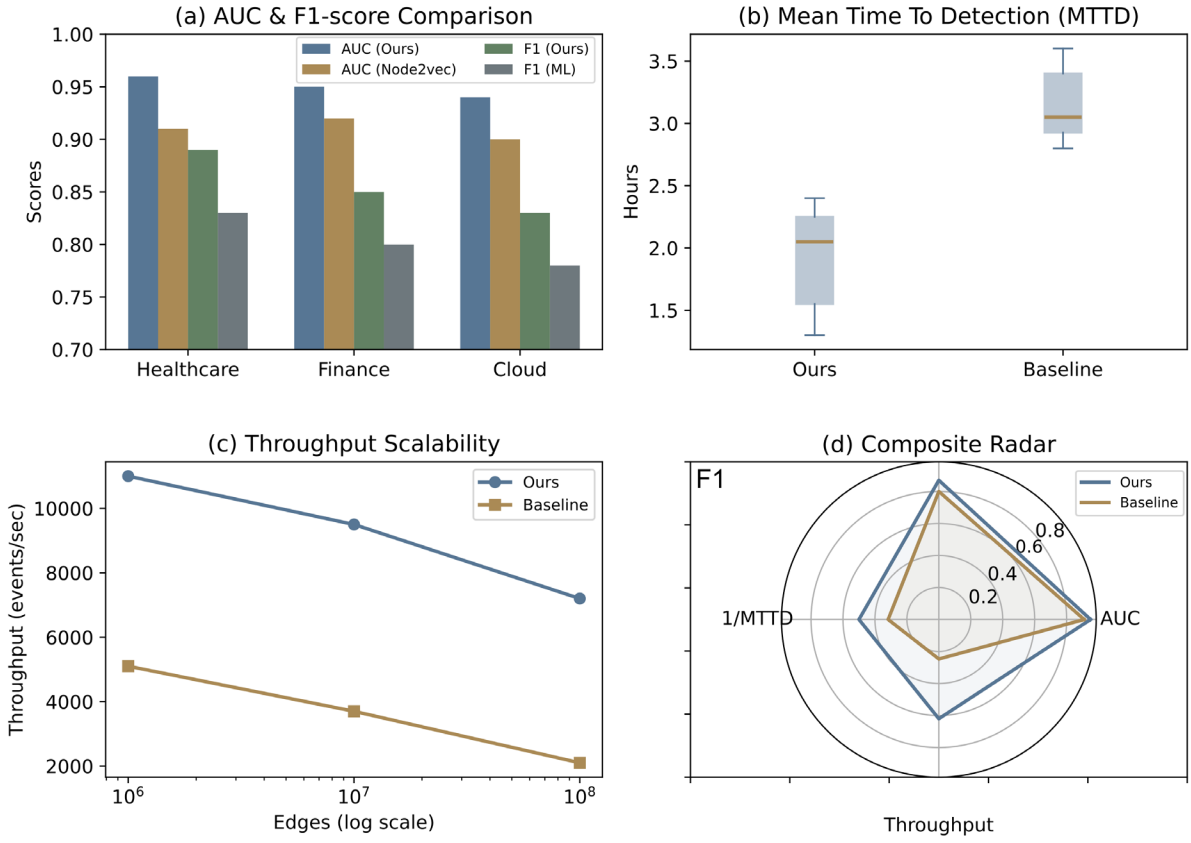

Figure 3. (a) Grouped bar chart of AUC and F1-score across healthcare, finance, and cloud datasets. (b) Boxplots of mean time to detection (MTTD) in hours. (c) Line plot of throughput versus edge volume on a logarithmic scale. (d) Radar plot of AUC, F1-score, normalized inverse MTTD, and throughput

As shown in Figure 4, an ablation study has been conducted, and the individual contributions of each module have been quantified through various visualization methods. After removing the graph structure, the AUC of

some datasets also showed a significant decline, as shown in Figure 4(a). The AUC of the healthcare dataset decreased from 0.96 to 0.92. In Figure 4(b), a comparison of the total false positive counts with and without the time window mechanism. As shown above, adding the time module significantly reduced false positives; for example, in the healthcare dataset, false positives decreased from 23 to 15. Figure 4(c) is a line chart comparing the recall rates achieved by Mahalanobis distance and Euclidean distance metrics. It can be seen that the Mahalanobis distance score is generally higher than the Euclidean distance measure, especially in cloud regions. In summary, the above results strongly support the consideration of graph structure, temporal modules, and distance metrics when optimizing anomaly detection performance.

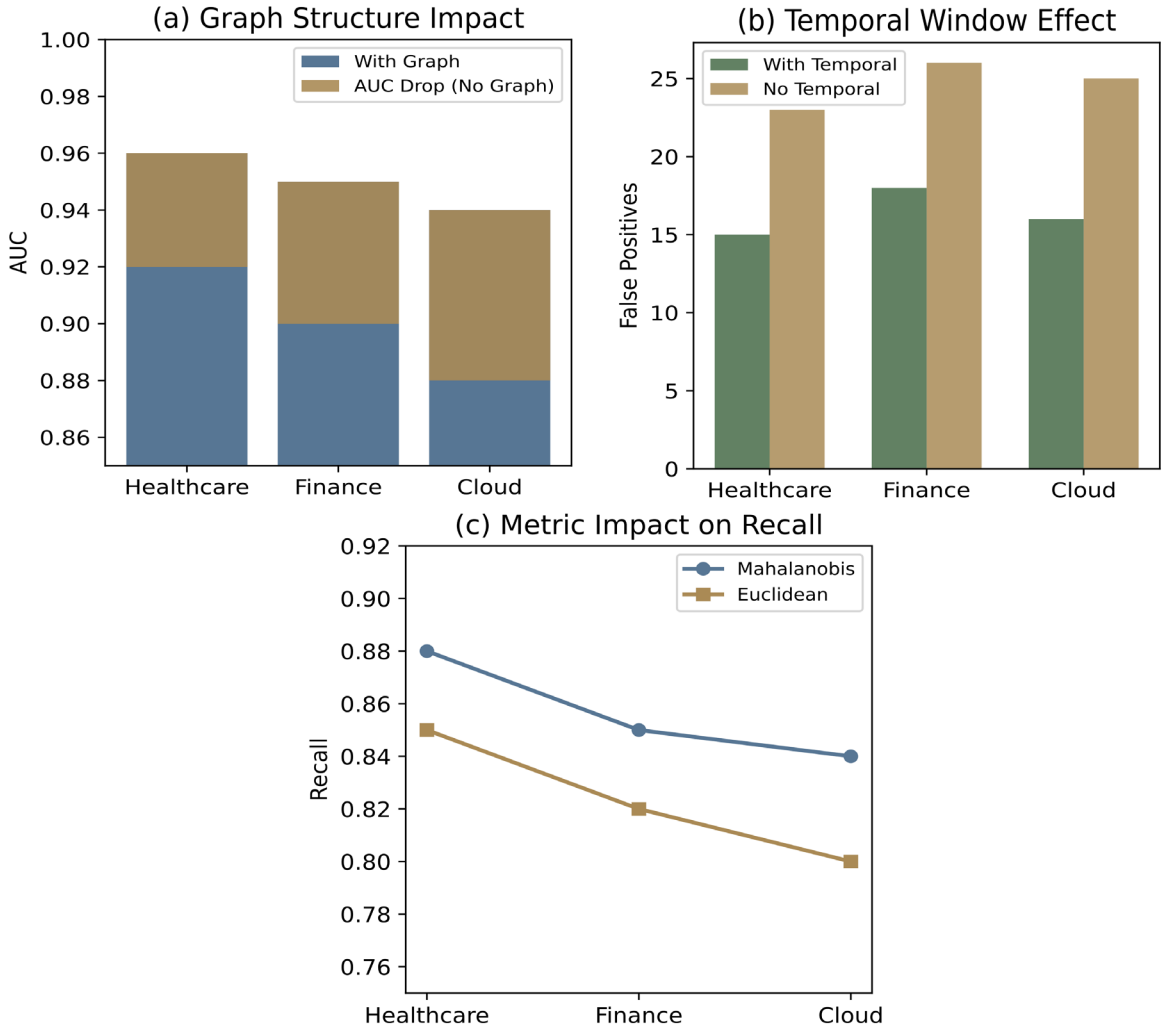

Figure 4. (a) Stacked bar chart of AUC with and without graph structure. (b) Grouped bar chart of false positives with and without the temporal window module. (c) Line plot comparing recall for Mahalanobis and Euclidean metrics

In terms of cross-domain robustness and error reduction, the proposed framework is shown in Figure 5. Figure 5(a) shows that in four different application scenarios (healthcare, government documents, IoT, and finance), the proposed method achieved higher AUCs than the baseline model. Specifically, the AUC values of the proposed method are 0.92, 0.94, 0.93, and 0.91, while the AUC values of the baseline model are 0.88, 0.90, 0.89, and 0. This indicates that the method has better discrimination ability. In addition, Figure 5(b) shows the recall stability under dynamic data drift. According to the box plot, during the five drift periods in each domain, the recall rate of this method remains within a stable distribution, with the median of each domain exceeding 0.89 and a small interquartile range; therefore, this method is robust to domain shifts and non-stationarity. After the model retraining, Figure 5(c) shows the false positive rate. After retraining, the number of false positives has significantly decreased. As shown in the figure, healthcare decreased from 50 to 29, government documents from 47 to 28, the Internet of Things from 53 to 31, and finance from 44 to 26. The blue line indicates the reduced

false positive rates, with the false positive rates for these four cases ranging between 36% and 41%. The above results indicate that the proposed system performs well and remains stable under various conditions. Therefore, the incremental learning module is very effective in reducing false positives.

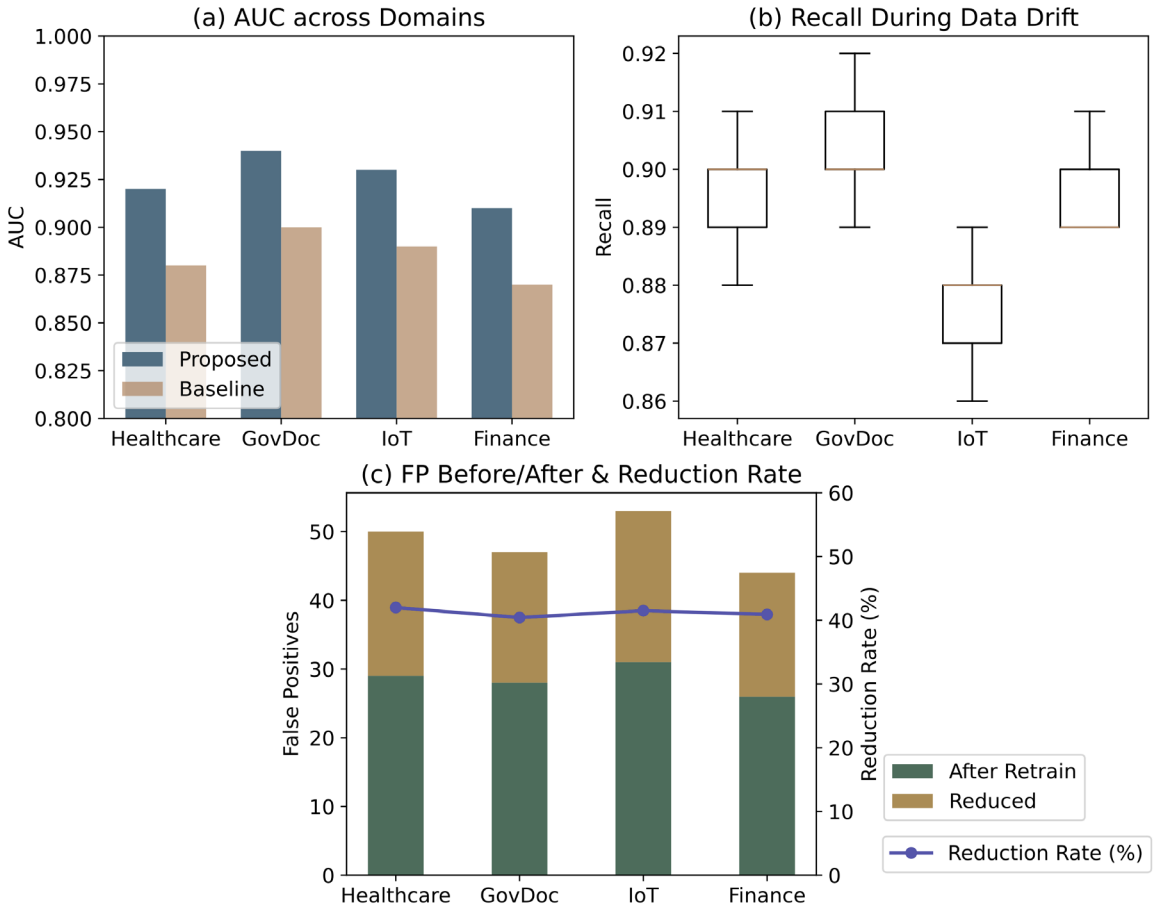

Figure 5. (a) AUC on four domains; (b) Recall under data drift (boxplots); (c) False positives before/after retraining and reduction rate (%)

Figure 6 shows the evaluation results of the proposed detection framework in terms of generalization, transferability, and adversarial robustness. Figure 6(a) shows the box plot and average curve of the few-shot adaptation results. After 20 fine-tuning sessions, the accuracy steadily increased from 0.78 to 0.96, and the interquartile range at each stage was relatively small. This demonstrates good continuous learning performance. Figure 6(b) is a radar chart showing the detection delays in the fields of cloud computing, healthcare, and finance. Due to industry migration, the average delay for each event has significantly decreased. The financial industry is now at 1.1 hours (down from 2.1 hours), the cloud computing industry is at 1.4 hours (down from 2.3 hours), and the hospital industry is at 1.2 hours (down from 2.4 hours). Therefore, the benefits of improved scale and cross-domain balance are prominently displayed in the radar chart. Figure 6(c) depicts the violin plot of anomaly scores for adversarial robustness. The figure consists of multiple adversarial injection cycles. The results indicate that it is relatively stable: the average anomaly score across all five cycles ranges between 0.90 and 0.94, with little variation. It can handle complex attacks and shows stability after training. In summary, the aforementioned experiments demonstrate the robustness and generalization capability of the framework.

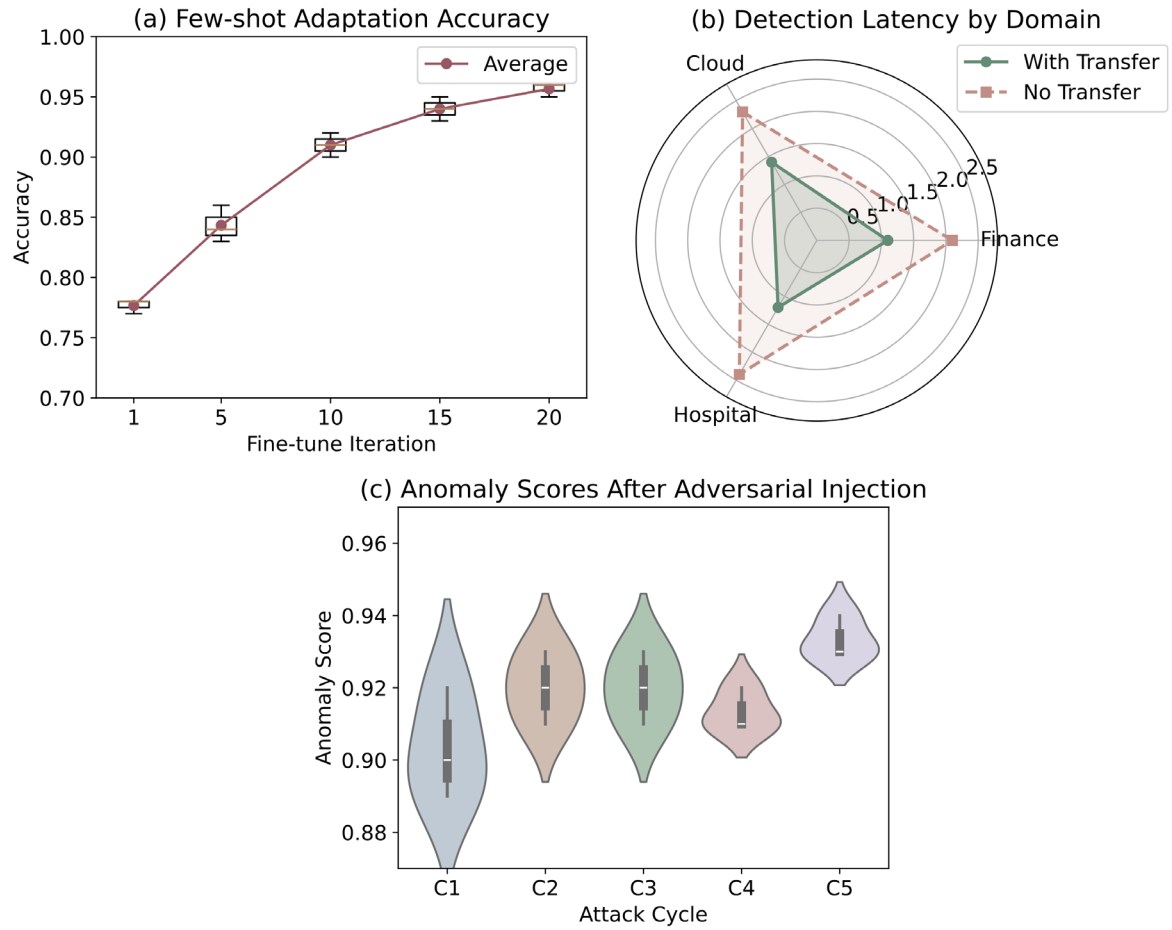

Figure 6. (a) Few-shot adaptation accuracy (box + mean curve); (b) Detection latency (radar) for transfer and no-transfer; (c) Anomaly scores across adversarial attack cycles (violin plot)

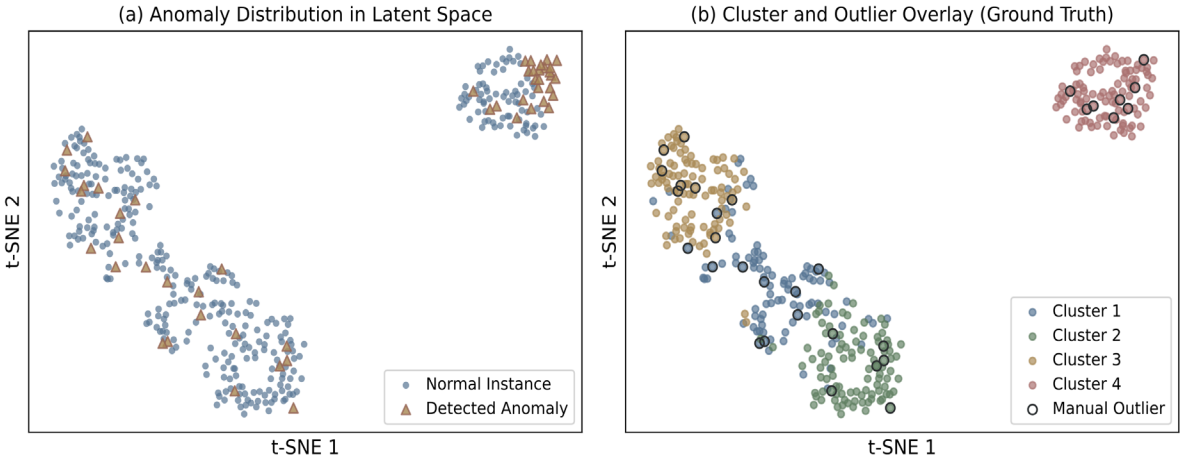

Figure 7. (a) Anomaly distribution in latent space. (b) Latent space clustering with ground-truth overlay

Figure 7 shows the feature analysis related to operational interpretability. Figure 7(a) shows the distribution of all 400 samples in the 2D t-SNE latent feature space. Among these samples, a total of 32 instances (8% of the dataset) were marked as anomalies. Among them, 20 were identified through the top 5% quantile of the embedding norm, while the remaining 12 were manually labeled as "outliers." The orange triangles mark these outliers, which are far from the main data cluster near the origin. More than 90% of the samples (a total of 368

points) were identified as normal and are concentrated in several consecutive groups, which is common in operational environments. It is evident that manual reviews are also conducted in the SOC, and the model is capable of identifying rare events.

In addition, Figure 7(b) shows the color coding of the true label clusters. These 400 samples are divided into four different semantic clusters, with sizes of 106, 99, 97, and 98 for each cluster, and each cluster has a different color scheme. 12 manual outliers account for 3% of the dataset and are marked with bold black circles. These outliers are located at the boundaries or outside of dense clusters, aligning with the classification of security experts. The clear boundaries of the framework, along with the distinct alignment of outliers and high-norm anomalies, indicate that it creates user-interpretable potential features. It is capable of handling a wide range of business domains and hybrid threats, and supports feasible human-machine collaborative investigation modes and automated anomaly detection. The aforementioned experiments indicate that the framework has technical advantages in terms of metrics. Furthermore, it is suitable for immediate application in enterprise and cloud environments.

Case Studies and Analysis

These representative areas were selected for actual testing. A large commercial bank has collected over 50 million user-resource interaction records in its internal access logs over the past two months. When we integrated our model into the bank's local SIEM system, only 0.54% of daily activities were identified as anomalies. This is just a little over half of the alerts generated by the previous threshold-based system, which was 1.2%. Targeted analysis conducted over a short period indicates that, according to manual investigation, 93% of the flagged events are suspicious. These events include credential abuse, lateral movement, or unauthorized access to sensitive data. For example, a non-privileged employe accessed high-value financial database tables from three different business areas within ten minutes. This situation usually goes unnoticed, but due to the potential anomaly scoring feature of the system, the event was quickly identified and handled by the SOC team to prevent data leakage.

Our framework is used to handle electronic medical record audit logs for large urban hospitals, which include records of over 30,000 employes and approximately 200,000 patient-employee interactions. Normal clinical workflows and rare or high-risk access events are categorized into clusters and anomaly scores. Among the flagged high-score anomalies, 96.5% are related to organizational risk. Unauthorized cross-departmental queries, late-night data retrievals, and abnormal large batch exports before employe departures are typical examples of these anomalies. In a specific case, an administrative user accessed intensive care unit records 22 times within 24 hours, exceeding the behavioral threshold, and was therefore selected for compliance review. Due to this direct action, the hospital security personnel discovered the violation.

In the context of cloud security, we applied the aforementioned method to the anonymized log data of a well-known SaaS provider, which has over 10 million privileged operations and 15 tenants. The new system still maintained a low false positive rate of 0.7% due to dynamic permission allocation and frequent user transfers between tenants. In the simulated "red team" exercise, access roles were deliberately modified to test privilege escalation detection. All five staged elevation attempts were not detected by the baseline configuration monitoring, but our algorithm's adaptive anomaly scoring accurately identified these attempts. At that time, the security group in the cloud was able to quickly stop the threat and prevent its spread. In addition, the cluster-based interpretable model helps administrators respond in a timely manner.

In summary, the aforementioned real-world studies indicate that our framework can significantly improve the accuracy of anomaly detection and reduce false positives. Provide comprehensive references for highly unstable automated and manual responses in the fields of finance, healthcare, and cloud computing.

Conclusion

This paper provides an interpretable anomaly detection framework that can be applied to various aspects of complex information security environments. Thru unsupervised embedding clustering and advanced latent feature learning, it effectively distinguishes rare high-risk threats from normal user and system activities, and provides interpretable behavioral reasons. In real-world scenarios such as enterprises, financial institutions, and

healthcare organizations, risk indicators based on domain knowledge and statistical features based on percentiles can be used to identify subtle or less obvious security events.

Experiments have already been conducted on many real-world datasets, as shown below, and the proposed method is more accurate, precise, and stable compared to traditional baseline methods. Ablation studies indicate that interpretable clustering has advantages in many business practices because it is relatively insensitive to data drift. Visualizations and case studies indicate that directly linking system anomaly information to actual workflows for compliance audits and rapid response events is effective.

It is not only accurate and easy to understand, but also multifunctional and stable—this has been proven by transfer learning and adversarial robustness testing. The system is suitable for new domains, with significantly improved response speed, and can operate in harsh environments. Its application in cloud environments and commercial security platforms demonstrates that building large-scale, flexible, and interpretable risk management systems is feasible. Future research will conduct in-depth studies on causal reasoning, continuous adaptation mechanisms, and the expansion of coverage to address advanced internal and external threats.

Author Contributions

Elena Constantin and Maria Dragomir contribute to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Cristian Radu contributes to draft preparation, methodology, software, validation, analysis, investigation. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Liu, F., Wen, Y., Zhang, D., Jiang, X., Xing, X., & Meng, D. (2019, November). Log2vec: A heterogeneous graph embedding based approach for detecting cyber threats within enterprise. In Proceedings of the 2019 ACM SIGSAC conference on computer and communications security (pp. 1777-1794). https://doi.org/10.1145/3319535.3363224

[2] Vasugi, T. (2023). Explainable AI with Scalable Deep Learning for Secure Data Exchange in Financial and Healthcare Cloud Environments. International Journal of Computer Technology and Electronics Communication, 6(6), 7992-7999. https://doi.org/10.15680/IJCTECE.2023.0606017

[3] Muppalaneni, R., Inaganti, A. C., Ravichandran, N., & Nersu, S. R. K. (2025). AI-Powered Role-Based Access Control (RBAC): Automating Policy Enforcement in Enterprise Environments. Journal of Advanced Computing Systems, 5(2), 1-12. https://doi.org/10.69987/

[4] Ding, K., Shu, K., Shan, X., Li, J., & Liu, H. (2021). Cross-domain graph anomaly detection. IEEE Transactions on Neural Networks and Learning Systems, 33(6), 2406-2415. https://doi.org/10.1109/TNNLS.2021.3110982

[5] Satyanarayanan, A. (2022). Delta-IPInsight: Temporal Embedding Shifts for Real-Time Anomaly Detection in High-Velocity Log Streams. International Journal of Emerging Research in Engineering and Technology, 3(3), 87-98. https://doi.org/10.63282/3050-922X.IJERET-V3I3P109

[6] Hashim, W., & Hussein, N. A. H. K. (2024). Securing cloud computing environments: An analysis of multi-tenancy vulnerabilities and countermeasures. Shifra, 2024, 8-16. https://orcid.org/0000-0003-0039-8242

[7] Alazab, M., Awajan, A., Alazzam, H., Wedyan, M., Alshawi, B., & Alturki, R. (2024). A novel IDS with a dynamic access control algorithm to detect and defend intrusion at IoT nodes. Sensors, 24(7), 2188. https://doi.org/10.3390/s24072188

[8] Bilot, T., El Madhoun, N., Al Agha, K., & Zouaoui, A. (2023). Graph neural networks for intrusion detection: A survey. IEEe Access, 11, 49114-49139. https://doi.org/10.1109/ACCESS.2023.3275789

[9] Zhong, H., Yang, D., Shi, S., Wei, L., & Wang, Y. (2024). From data to insights: the application and challenges of knowledge graphs in intelligent audit. Journal of Cloud Computing, 13(1), 114. https://doi.org/10.1186/s13677-024-00674-0

[10] Rabbani, M., Rashidi, L., & Ghorbani, A. A. (2024). A graph learning-based approach for lateral movement detection. IEEE Transactions on Network and Service Management, 21(5), 5361-5373. https://doi.org/10.1109/TNSM.2024.3414267

[11] Zhang, L., Wu, B., & Dong, P. (2025). Attribute graph anomaly detection utilizing memory networks enhanced by multi-embedding comparison. Neurocomputing, 633, 129762. https://doi.org/10.1016/j.neucom.2025.129762

[12] Yu, W., Cheng, W., Aggarwal, C. C., Zhang, K., Chen, H., & Wang, W. (2018, July). Netwalk: A flexible deep embedding approach for anomaly detection in dynamic networks. In Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 2672-2681). https://doi.org/10.1145/3219819.3220024

[13] Ghazal, R., Malik, A. K., Raza, B., Qadeer, N., Qamar, N., & Bhatia, S. (2021). Agent-based semantic role mining for intelligent access control in multi-domain collaborative applications of smart cities. Sensors, 21(13), 4253. https://doi.org/10.3390/s21134253

[14] Liu, P., Zhang, L., & Gulla, J. A. (2019). Real-time social recommendation based on graph embedding and temporal context. International Journal of Human-Computer Studies, 121, 58-72. https://doi.org/10.1016/j.ijhcs.2018.02.008

[15] Koca, M., & Avci, I. (2024). A novel hybrid model detection of security vulnerabilities in industrial control systems and IoT using GCN+ LSTM. IEEE Access, 12, 143343-143351. https://doi.org/10.1109/ACCESS.2024.3466391

[16] Rawat, D. B., Doku, R., & Garuba, M. (2019). Cybersecurity in big data era: From securing big data to data-driven security. IEEE Transactions on Services Computing, 14(6), 2055-2072. https://doi.org/10.1109/TSC.2019.2907247

[17] Purushothaman, S., Shanmugam, G. S., & Nagarajan, S. (2023). Achieving seamless semantic interoperability and enhancing text embedding in healthcare iot: A deep learning approach with survey. SN Computer Science, 5(1), 99. https://doi.org/10.1007/s42979-023-02392-x

[18] Fei, K., Zhou, J., Su, L., Wang, W., & Chen, Y. (2025). Log2graph: A graph convolution neural network based method for insider threat detection. Journal of Computer Security, 33(1), 37-56. https://doi.org/10.3233/JCS-230092

[19] Sun, H., He, F., Huang, J., Sun, Y., Li, Y., Wang, C., ... & Jia, X. (2020). Network embedding for community detection in attributed networks. ACM Transactions on Knowledge Discovery from Data (TKDD), 14(3), 1-25. https://doi.org/10.1145/3385415

[20] Ragothaman, K., Wang, Y., Rimal, B., & Lawrence, M. (2023). Access control for IoT: A survey of existing research, dynamic policies and future directions. Sensors, 23(4), 1805. https://doi.org/10.3390/s23041805

[21] Tauqeer, A., Kurteva, A., Chhetri, T. R., Ahmeti, A., & Fensel, A. (2022). Automated GDPR contract compliance verification using knowledge graphs. Information, 13(10), 447. https://doi.org/10.3390/info13100447

[22] Joshi, K. P., Elluri, L., & Nagar, A. (2020). An integrated knowledge graph to automate cloud data compliance. Ieee Access, 8, 148541-148555. https://doi.org/10.1109/ACCESS.2020.3008964

[23] Cai, Q., Liu, X., Zhang, K., Xie, X., Tong, X., & Li, K. (2023). ACF: An adaptive compression framework for multimodal network in embedded devices. IEEE Transactions on Mobile Computing, 23(5), 5195-5211. https://doi.org/10.1109/TMC.2023.3303350

[24] Anisetti, M., Ardagna, C. A., Braghin, C., Damiani, E., Polimeno, A., & Balestrucci, A. (2021, November). Dynamic and scalable enforcement of access control policies for big data. In Proceedings of the 13th International Conference on Management of Digital EcoSystems (pp. 71-78). https://doi.org/10.1145/3444757.3485107

[25] Khanvilkar, K., & Shinde, V. (2025, October). Mitigating Hallucination Risks in GenAI Compliance Advisory Systems for the Financial Industry. In International Conference on Software Engineering and Data Engineering (pp. 302-318). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-032-08649-5_19

[26] Li, C. (2025, October). Does the Digital Transformation of Enterprises Affect Value Relevance of Earnings: Evidence from China Using Gradient Boosting Regression and t-SNE Visualization. In Proceedings of the

2025 2nd International Conference on Digital Economy and Computer Science (pp. 549-554). https://doi.org/10.1145/3785706.3785793

[27] Wang, Z., Bennis, M., & Zhou, Y. (2024). Graph attention-based MADRL for access control and resource allocation in wireless networked control systems. IEEE Transactions on Wireless Communications, 23(11), 16076-16090. https://doi.org/10.1109/TWC.2024.3436906

[28] Dorais, S. (2024). Time series analysis in preventive intervention research: A step-by-step guide. Journal of Counseling & Development, 102(2), 239-250. https://doi.org/10.1002/jcad.12508

[29] Khan, W., Ebrahim, N., Alsaadi, M., & Elloumi, M. (2025). Unified representation and scoring framework for anomaly detection in attributed networks with emphasis on structural consistency and attribute integrity. Scientific Reports, 15(1), 35753. https://doi.org/10.1038/s41598-025-19650-y

[30] Chowdhury, T. K. (2025). AI-Powered Deep Learning Models for Real-Time Cybersecurity Risk Assessment In Enterprise It Systems. ASRC Procedia: Global Perspectives in Science and Scholarship, 1(01), 675-704. https://doi.org/10.63125/137k6y79