Comprehensive Journal of Science
Volume (11), Issue (41), (JUNE, 2026)
ISSN: 3014-6266

مجلة العلوم الشاملة
المجلد (11) العدد 41 يونيو 2026
ردمد: 3014-6266

A Behavioral Anomaly Detection Framework for Academic Information Systems Using Isolation Forest: A Case Study of Students and Academic Staff

Lutfia Mohamed Hagrassi
Computer Department, The Higher Institute of Industrial Technology, Tripoli, Libya
Email: lutfia.hagrassi.cs@gmail.com

تاريخ الاستلام:2026/05/06 –تاريخ المراجعة:2026/05/28– تاريخ القبول: 2026/06/08– تاريخ للنشر: 2026 /06/30

Abstract
Academic Information Systems (AIS) have become essential platforms for managing educational and administrative activities, making them attractive targets for cybersecurity threats. Although conventional authentication mechanisms verify user identities during login, they provide limited capability for detecting abnormal user behavior after successful authentication. This study proposes a machine learning-based behavioral anomaly detection framework for Academic Information Systems using the Isolation Forest algorithm. The proposed framework integrates user session records, audit logs, and user role information into a unified behavioral dataset extracted from a real operational Academic Information System. A database view was developed to aggregate behavioral information from multiple relational tables, resulting in a dataset containing 3,090 authenticated user sessions described by 15 behavioral attributes. After feature engineering and preprocessing, 12 behavioral features were used to train the Isolation Forest model. Three contamination values (0.01, 0.03, and 0.05) were experimentally evaluated to investigate their influence on anomaly detection performance. The final model, configured with a contamination value of 0.05, identified 155 anomalous sessions, corresponding to an anomaly detection rate of 5.02%.

Statistical analyses demonstrated that the proposed framework successfully distinguished behavioral deviations using temporal characteristics, session properties, user roles, and database activity patterns without requiring labeled attack data. The findings indicate that the proposed framework can serve as an effective early-warning behavioral monitoring mechanism to support cybersecurity in Academic Information Systems while preserving the advantages of unsupervised machine learning.

Keywords: Academic Information Systems; Cybersecurity; User Behavior Analysis; Anomaly Detection; Isolation Forest; Machine Learning; Behavioral Analytics.

1. Introduction

Academic Information Systems (AIS) have become essential components of modern educational institutions, supporting a wide range of academic and administrative services such as student registration, course enrollment, grade management, examination processing, and user administration. As universities continue their digital transformation, these systems process large volumes of sensitive academic and personal information, making them attractive targets for cybersecurity threats [4], [5], [7]

Traditional authentication mechanisms verify users during login but provide limited capability for monitoring user behavior after authentication [4], [6].

Consequently, ensuring the security and integrity of Academic Information Systems has become a critical requirement for protecting institutional assets and maintaining the continuity of educational services.

Despite the widespread adoption of authentication mechanisms such as usernames and passwords, traditional access control methods primarily verify user identity during the login process and provide limited capability for monitoring user behavior after authentication. As a result, unauthorized account usage, insider misuse, privilege abuse, and compromised user accounts may remain undetected while operating under valid credentials. Conventional security mechanisms are therefore insufficient for identifying behavioral anomalies that emerge during normal system operation.

Recent advances in Machine Learning have introduced behavior-based anomaly detection techniques capable of identifying unusual activities without relying on predefined attack signatures[3], [4]. Among these techniques, the Isolation Forest algorithm has demonstrated strong performance in detecting anomalies within unlabeled datasets by isolating observations that significantly differ from normal behavioral patterns [1], [2]. This characteristic makes Isolation Forest particularly suitable for academic environments, where labeled cybersecurity incidents are scarce and obtaining representative attack datasets is often impractical.

This study proposes a machine learning-based framework for detecting anomalous user behavior in an Academic Information System using behavioral features extracted from user sessions, audit logs, and user role information. A real institutional dataset was constructed from an operational academic information system and used to train an Isolation Forest model capable of identifying abnormal user activities without requiring labeled attack data. The proposed framework is evaluated through comprehensive behavioral analysis, statistical evaluation, and visualization of the detected anomalies to assess its effectiveness as an early-warning cybersecurity mechanism for academic institutions.

1.1 Research Questions

The increasing complexity of user activities within Academic Information Systems raises several important cybersecurity challenges related to behavioral monitoring and anomaly detection. Based on these challenges, this study addresses the following research questions:

RQ1: Can the Isolation Forest algorithm effectively detect anomalous user behavior in an Academic Information System using behavioral features extracted from user sessions, audit logs, and user role information?

RQ2: Which behavioral features contribute most significantly to distinguishing normal user activities from anomalous behaviors within an Academic Information System?

RQ3: Do behavioral patterns differ among students, academic staff, and administrative users, and how do these differences influence anomaly detection?

RQ4: How effective is the proposed behavioral analysis framework as an early-warning mechanism for improving cybersecurity in Academic Information Systems?

1.2 Research Hypothesis

Based on the characteristics of behavioral data and the capabilities of unsupervised anomaly detection techniques, this study investigates the following hypothesis:

H1: Behavioral features extracted from user sessions, audit logs, and user role information provide sufficient information for the Isolation Forest algorithm to identify anomalous user activities without requiring labeled attack data.

This hypothesis is evaluated experimentally using a real-world dataset collected from an operational Academic Information System.

1.3 Research Contributions

The main contributions of this study can be summarized as follows:

1. A comprehensive behavioral dataset was constructed from a real Academic Information System by integrating user sessions, audit logs, and user role information into a unified analytical dataset.
2. A database view was designed to automatically aggregate behavioral information from multiple relational tables, simplifying data extraction while preserving data consistency.
3. An unsupervised anomaly detection framework based on the Isolation Forest algorithm was developed to identify abnormal user behavior without relying on labeled cybersecurity incidents.

4. A comprehensive experimental evaluation was conducted using real institutional data, including statistical analysis, anomaly score visualization, sensitivity analysis of the contamination parameter, and behavioral analysis across different user roles and login characteristics.

5. The proposed framework demonstrates the feasibility of integrating machine learning techniques into Academic Information Systems to provide an early-warning cybersecurity mechanism capable of detecting potentially abnormal user activities before security incidents escalate.

3. Proposed Framework

This study proposes a behavioral anomaly detection framework designed to identify abnormal user activities within an Academic Information System using unsupervised machine learning techniques. The framework integrates multiple sources of behavioral information, including user session records, audit logs, and user role information, to construct a comprehensive behavioral dataset capable of representing user activities during system operation.

As illustrated in Figure 1, the proposed framework consists of six sequential stages. First, behavioral data are collected from the operational Academic Information System, where user sessions, audit logs, and user role information are extracted from the relational database. These heterogeneous data sources are then integrated through a database view that provides a unified representation of user activities.

In the second stage, feature engineering is performed to derive meaningful behavioral attributes describing login characteristics, session duration, device information, operating systems, user roles, and database activities. The extracted features are subsequently preprocessed through categorical encoding, removal of identifier attributes, and elimination of non-informative fields to produce a machine-learning-ready dataset.

The processed behavioral dataset is then supplied to the Isolation Forest algorithm, which isolates observations exhibiting behavioral characteristics significantly different from the majority of user sessions. The algorithm assigns an anomaly score to each session, allowing user activities to be classified as either normal or anomalous without requiring labeled attack data.

Finally, the detected anomalies are analyzed statistically using behavioral characteristics, user roles, login times, device information, operating systems, and anomaly score distributions in order to evaluate the effectiveness of the proposed framework as an early-warning cybersecurity mechanism for Academic Information Systems.

Figure 1.

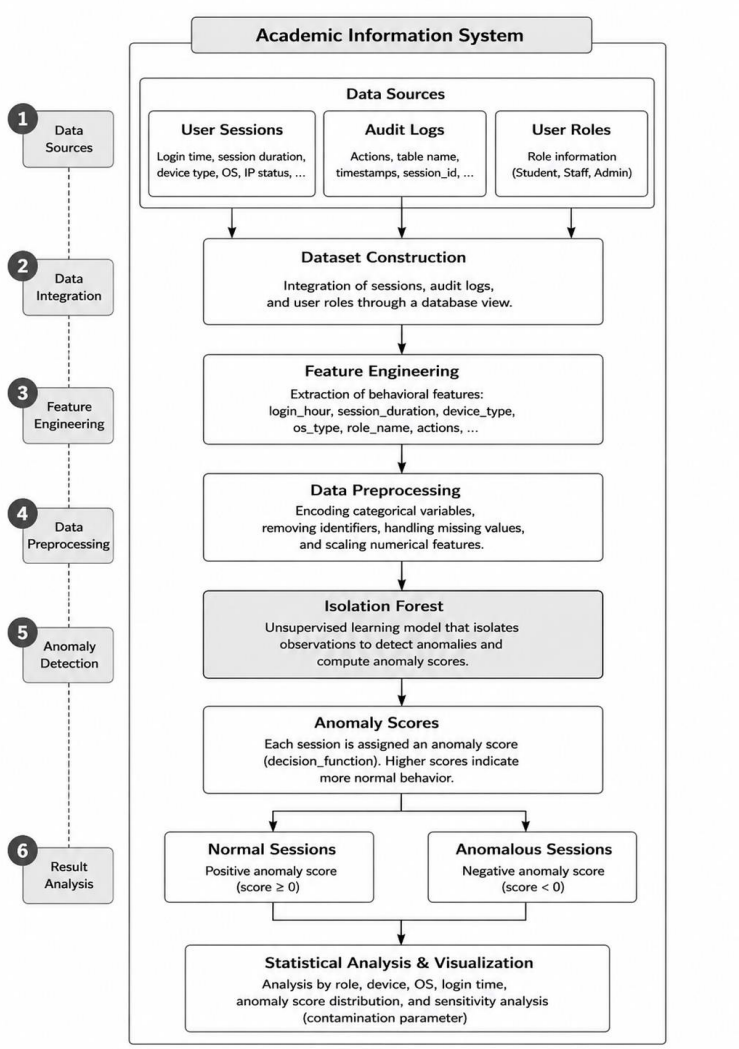

Figure 1. Proposed Behavioral Anomaly Detection Framework

2 Related Work

2.1 Machine Learning for Cybersecurity

Machine learning has become one of the most promising approaches for enhancing cybersecurity through the automatic analysis of large volumes of security-related data. [4], [5]

Unlike traditional rule-based security mechanisms, machine learning techniques are capable of identifying hidden behavioral patterns and detecting previously unseen attacks without relying solely on predefined signatures. Consequently, behavioral analysis has become an important research direction for protecting information systems against insider threats, compromised accounts, and abnormal user activities [3], [6]. Recent studies have demonstrated the effectiveness of anomaly detection techniques across various cybersecurity domains, including network monitoring, business process analysis, web server protection, and user behavior analytics.

2.2 Isolation Forest for Anomaly Detection

Among unsupervised anomaly detection algorithms, Isolation Forest has attracted considerable attention because of its computational efficiency and its ability to detect anomalies without requiring labeled training data [1], [2]. Instead of constructing a model describing normal behavior, Isolation Forest isolates observations through

recursive random partitioning, allowing anomalous observations to be identified based on shorter path lengths within isolation trees[1].

Recent research has further extended the Isolation Forest algorithm to improve its applicability in large-scale environments, graph-based behavioral analysis, and non-independent data distributions[2]. These studies demonstrate the flexibility of Isolation Forest across different application domains while maintaining its effectiveness as a lightweight unsupervised anomaly detection technique.

2.2 Isolation Forest for Anomaly Detection

Among unsupervised anomaly detection algorithms, Isolation Forest has attracted considerable attention because of its computational efficiency and its ability to detect anomalies without requiring labeled training data. Instead of constructing a model describing normal behavior, Isolation Forest isolates observations through recursive random partitioning, allowing anomalous observations to be identified based on shorter path lengths within isolation trees.

Recent research has further extended the Isolation Forest algorithm to improve its applicability in large-scale environments, graph-based behavioral analysis, and non-independent data distributions. These studies demonstrate the flexibility of Isolation Forest across different application domains while maintaining its effectiveness as a lightweight unsupervised anomaly detection technique.

2.3 Behavioral Anomaly Detection

Behavior-based anomaly detection has recently emerged as an effective strategy for identifying suspicious user activities after successful authentication[3], [6]. Rather than focusing solely on login credentials, behavioral approaches analyze user interactions, session characteristics, temporal information, and operational activities to distinguish normal behavior from unusual behavioral patterns.

Several studies have successfully applied Isolation Forest to behavioral datasets collected from web server logs, business process event logs, and mobile user activities[1], [2]. These investigations reported that combining behavioral features with unsupervised anomaly detection algorithms enables effective identification of abnormal activities while reducing dependence on predefined attack signatures.

2.4 Research Gap

Despite the significant progress achieved by previous studies, several research gaps remain.
First, most existing studies evaluate anomaly detection algorithms using publicly available benchmark datasets or simulated behavioral data rather than operational data collected from real Academic Information Systems.
Second, previous research typically focuses on a single source of behavioral information, such as network traffic, web server logs, or business process event logs, while limited attention has been given to integrating user sessions, audit logs, and user role information into a unified behavioral dataset.

Third, relatively few studies investigate behavioral anomaly detection specifically within Academic Information Systems, despite the increasing cybersecurity challenges faced by educational institutions.

To address these limitations, the present study proposes a behavioral anomaly detection framework based on the Isolation Forest algorithm using a behavioral dataset extracted from a real Academic Information System. Unlike previous approaches, the proposed framework integrates user sessions, audit logs, and user role information into a unified analytical dataset and evaluates anomaly detection using operational data collected from an educational environment.

4. Methodology

This section presents the methodology adopted to develop the proposed behavioral anomaly detection framework. The framework consists of five sequential stages: data collection, dataset construction, feature engineering, data preprocessing, and behavioral anomaly detection using the Isolation Forest algorithm. Figure 1 illustrates the

complete workflow, beginning with behavioral data extraction from the Academic Information System and ending with the statistical analysis and visualization of detected anomalous user sessions.

4.1 Data Collection

The experimental data used in this study were collected from a real-world Academic Information System developed to support student registration, course enrollment, examination management, and academic administration within a higher education institution. The system is actively used in daily operations and stores transactional and behavioral information in a MySQL relational database.

User activities generated during normal system operation are continuously recorded through several relational tables, including authentication records, user sessions, audit logs, and user role information. These operational records provide detailed behavioral information describing how different categories of users interact with the system throughout their daily activities.

Unlike many previous studies that rely on publicly available cybersecurity datasets, this research utilizes operational behavioral data collected from an actual academic environment. Consequently, the proposed anomaly detection framework is evaluated using realistic user behavior rather than simulated attack scenarios.

To preserve user privacy, personally identifiable information was not used during model training. Identifier attributes were retained only during dataset construction to maintain record consistency and were subsequently removed before training the machine learning model. Similarly, IP addresses were excluded from the final training dataset to reduce feature dimensionality and avoid unnecessary dependence on network-specific information.

The resulting dataset therefore represents authentic behavioral activities performed by students, academic staff, and administrative personnel, providing a realistic basis for evaluating behavioral anomaly detection within Academic Information Systems.

4.2 Dataset Construction

The behavioral dataset was constructed by integrating information distributed across multiple relational tables into a unified analytical dataset suitable for machine learning. Rather than repeatedly joining several tables during data analysis, a dedicated database view named dataset_user_behavior was developed to automatically aggregate behavioral information from different components of the Academic Information System.

The constructed dataset combines three primary categories of behavioral information:

- **User session information**, including login time, session duration, device type, operating system, and session status.
- **Audit log information**, including aggregated counts of database operations such as INSERT, UPDATE, and DELETE activities performed during each authenticated session.
- **User role information**, representing the organizational responsibilities of users, including students, academic staff, and administrative personnel.

Several behavioral attributes were also derived directly within the SQL view, reducing preprocessing complexity while ensuring consistency between the operational database and the machine learning workflow.

Each record in the resulting dataset represents a single authenticated user session together with its corresponding behavioral characteristics. The final dataset contains 3,090 authenticated user sessions described by 15 behavioral attributes, providing a comprehensive representation of user behavior suitable for anomaly detection.

Table 1 summarizes the behavioral features extracted from the Academic Information System.

4.3 Feature Engineering

Feature engineering was performed to transform raw operational data into meaningful behavioral indicators suitable for anomaly detection. Rather than relying directly on transactional database records, a set of behavioral features describing user interactions with the Academic Information System was extracted from user sessions, audit logs, and user role information.

The selected features represent multiple dimensions of user behavior, including temporal characteristics, session properties, user privileges, device information, operating system characteristics, and aggregated database activities. These behavioral indicators were specifically selected because they describe user behavior rather than user identity, making them suitable for unsupervised anomaly detection.

The engineered feature set includes attributes such as login hour, day of the week, login month, session duration, user role, device type, operating system, risk weight, action count, insert count, update count, and delete count. Collectively, these behavioral attributes provide a comprehensive representation of user activities and serve as the input variables for the Isolation Forest model.

4.4 Data Preprocessing

Before training the anomaly detection model, several preprocessing operations were performed to improve data quality and prepare the behavioral dataset for machine learning analysis.

Initially, the dataset was imported directly from the dataset_user_behavior database view into Python using the Pandas library. The imported dataset contained 3,090 authenticated user sessions represented by 15 behavioral attributes.

A data quality assessment was then conducted to verify the completeness and consistency of the collected data. The inspection confirmed that no missing values were present in any attribute, indicating that the behavioral dataset was fully populated and suitable for further analysis without requiring data imputation.

During exploratory analysis, some observations appeared to share identical behavioral characteristics. Further investigation confirmed that these records represented different authenticated user sessions rather than duplicated database entries. Therefore, the unique session identifier (session_id) was retained solely for traceability purposes and excluded from model training.

Since session_id and user_id uniquely identify users and sessions rather than describing behavioral characteristics, both attributes were removed prior to training the machine learning model. Similarly, the ip_address attribute was excluded because it represents network identity rather than behavioral information and could introduce unnecessary bias into the anomaly detection process.

The remaining categorical attributes (role_name, device_type, and os_type) were transformed into numerical representations using the Label Encoding technique provided by the Scikit-learn library. Numerical behavioral attributes were retained without additional normalization because the Isolation Forest algorithm is relatively insensitive to differences in feature scales.

Following preprocessing, the final training dataset consisted of 12 numerical behavioral features, which were supplied directly to the Isolation Forest algorithm for behavioral anomaly detection.

4.5 Isolation Forest Model

The anomaly detection component of the proposed framework is based on the Isolation Forest algorithm, an unsupervised machine learning technique specifically designed for identifying anomalous observations within large datasets[1], [2]. Unlike supervised classification algorithms, Isolation Forest does not require labeled attack data during training, making it particularly suitable for cybersecurity applications where malicious activities are often unknown or unavailable[1].

Instead of learning a decision boundary between predefined classes, Isolation Forest isolates observations by recursively partitioning the feature space using randomly selected attributes and randomly generated split values. Since anomalous observations are relatively rare and exhibit behavioral characteristics different from the majority

of the data, they generally require fewer partitioning steps to become isolated. Consequently, anomalous observations produce shorter path lengths within the isolation trees than normal observations.

In this study, the Isolation Forest algorithm was applied to behavioral features extracted from authenticated user sessions, allowing the proposed framework to identify suspicious behavioral patterns without requiring prior knowledge of cyberattacks or manually labeled anomalies.

4.5.1 Principle of Isolation Forest

The Isolation Forest algorithm constructs an ensemble of randomly generated binary trees known as Isolation Trees (iTrees)[1], [2]. During tree construction, one feature is selected randomly, followed by a random split value within the range of that feature. This recursive partitioning continues until every observation becomes isolated or a predefined tree depth is reached.

The underlying assumption is that anomalous observations are easier to isolate because they differ substantially from normal behavioral patterns[1]. Consequently, anomalous sessions appear closer to the root of the isolation trees, whereas normal user sessions require a larger number of recursive partitions before becoming isolated.

The anomaly score assigned to each observation depends on the average path length computed across all isolation trees within the forest. Shorter average path lengths indicate higher probabilities of anomalous behavior.

4.5.2 Mathematical Formulation

The average path length of an observation is first computed across all isolation trees.

$$E(h(x)) = \frac{1}{t} + \sum_{i=1}^{t}(h_i(x))$$

Where:

- $E(h(x))$is the average path length of observation x.
- $t$is the total number of isolation trees.
- $h_i(x)$is the path length of observation $x$in the i^{th}isolation tree.

The anomaly score is then calculated as follows:

$$s(x,n) = 2^{-\frac{E(h(x))}{c(n)}}$$

Where:

- $s(x,n)$is the anomaly score of observation x.
- $E(h(x))$is the average path length.
- $n$is the number of training instances.
- $c(n)$is the average path length of unsuccessful searches in a Binary Search Tree and is calculated a

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n}$$

where:

- $H(n)$is the harmonic number, approximated by

$$H(n) \approx \ln(n) + 0.5772156649$$

where 0.5772156649 is the Euler–Mascheroni constant.

In Isolation Forest, observations with shorter average path lengths obtain higher anomaly scores because they can be isolated using fewer random partitions. Conversely, observations requiring longer path lengths are considered normal since they are located within denser regions of the feature space.

An anomaly score close to **1** indicates that an observation is highly anomalous, whereas values approaching **0** generally correspond to normal behavioral patterns.

4.5.3 Model Configuration

The Isolation Forest model was implemented using the IsolationForest class provided by the Scikit-learn machine learning library.

To evaluate the influence of the contamination parameter on anomaly detection performance, three independent experiments were conducted using contamination values of 0.01, 0.03, and 0.05.

All experiments employed identical model parameters, including:

Table 2

Parameter	**Value**
Algorithm	Isolation Forest
Implementation	Scikit-learn
Number of Trees (n_estimators)	100
Random State	42
Training Features	12
Training Sessions	3,090

The experimental comparison demonstrated that the contamination parameter strongly influences the proportion of observations classified as anomalous. Detection rates of 1.00%, 3.01%, and 5.02% were obtained for contamination values of 0.01, 0.03, and 0.05, respectively.

Based on these experimental results, contamination = 0.05 was selected as the final model configuration because it provided the most appropriate representation of behavioral anomalies while maintaining consistency with the expected proportion of abnormal user activities within the collected dataset.

Table 2 summarizes the final model configuration.

4.5.4 Prediction Process

After training the final Isolation Forest model, each authenticated user session was assigned both an anomaly score and a binary prediction label.

The anomaly score was computed using the decision_function() provided by Scikit-learn, which measures the relative degree of abnormality of each behavioral observation. Subsequently, the predict() function classified every session into one of two categories:

- Normal Session (prediction = 1)

- Anomalous Session (prediction = -1)

The predicted labels and corresponding anomaly scores were merged with the original behavioral dataset to facilitate statistical analysis and visualization. This enriched dataset served as the basis for evaluating user behavior across different roles, login characteristics, device types, operating systems, and other behavioral attributes presented in the experimental results.

5. Experimental Results and Discussion

5.1 Experimental Setup

The proposed behavioral anomaly detection framework was implemented using Python and the Scikit-learn machine learning library. Behavioral data were extracted directly from the MySQL database through the dataset_user_behavior database view and processed using the Pandas library.

The final behavioral dataset consisted of 3,090 authenticated user sessions represented by 15 behavioral attributes. Following the preprocessing stage, 12 behavioral features were retained for training the Isolation Forest model after removing identifier attributes and encoding categorical variables.

To evaluate the influence of the contamination parameter, three independent experiments were conducted using contamination values of 0.01, 0.03, and 0.05. Based on the obtained results, the final model employed 100 isolation trees, a contamination value of 0.05, and a fixed random state of 42.

The generated predictions and anomaly scores were exported for statistical analysis and visualization. All experimental figures and tables presented in this study were produced directly from the generated results using Python and Matplotlib.

5.2 Overall Detection Performance

Table 3. Overall Detection Performance

Metric	Value
Total User Sessions	**3090**
Normal Sessions	**2935**
Detected Anomalous Sessions	**155**
Detection Rate	**5.02%**

Using the final model configuration with a contamination value of 0.05, the model classified 2,935 sessions as normal and 155 sessions as anomalous, corresponding to an overall anomaly detection rate of 5.02%.

5.2 Overall Detection Performance

The trained Isolation Forest model was applied to the behavioral dataset containing 3,090 authenticated user sessions collected from the Academic Information System. Using the final model configuration with a contamination value of 0.05, the model classified 2,936 sessions as normal and 154 sessions as anomalous, corresponding to an overall anomaly detection rate of 4.98%.

Table 3 summarizes the overall classification results obtained from the proposed anomaly detection framework.

Figure 2 presents the overall distribution of normal and anomalous user sessions identified by the Isolation Forest model. The results indicate that the overwhelming majority of user activities followed consistent behavioral patterns, whereas only a small proportion of sessions exhibited sufficient behavioral deviation to be classified as anomalous.

It is important to emphasize that the detected anomalous sessions should not be interpreted as confirmed cyberattacks. Since Isolation Forest is an unsupervised anomaly detection algorithm, the identified sessions simply represent behavioral observations that differ significantly from the dominant user population. Consequently, both legitimate uncommon activities and potentially suspicious behaviors may be classified as anomalous and therefore require further investigation by system administrators.

Figure 2.

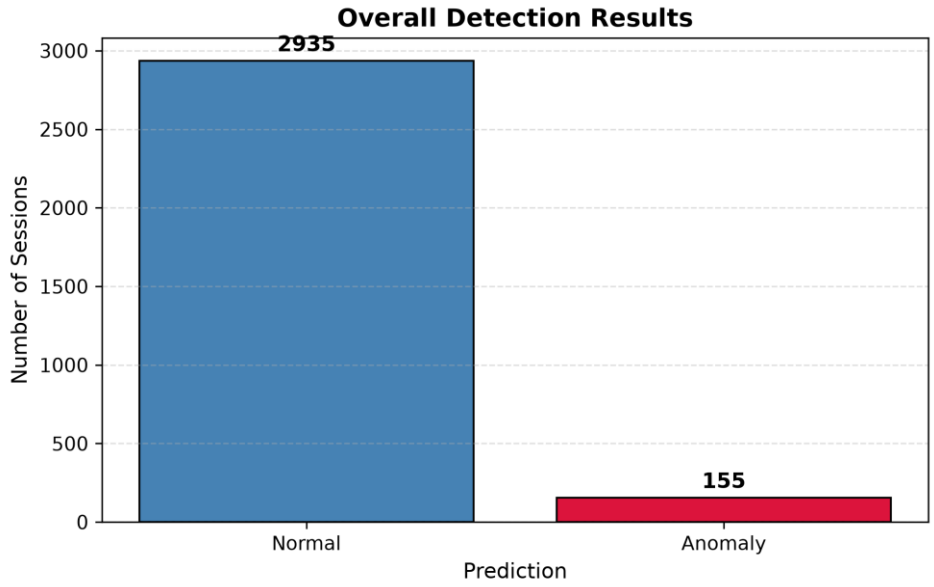

Figure 2. Overall classification of user sessions into normal and anomalous behaviors using the Isolation Forest model.

5.3 Role-Based Behavioral Analysis

To further investigate the detected anomalies, the distribution of anomalous sessions was analyzed according to user roles within the Academic Information System. Figure 3 illustrates the number of anomalous sessions identified for each user category.

The experimental results indicate that student accounts accounted for the largest number of detected anomalies, with 83 anomalous sessions, followed by Examination Department Heads with 53 sessions, and Department Members with 18 sessions.

At first glance, the high anomaly rate observed for administrative roles might suggest suspicious behavior. However, this result should be interpreted in the operational context of the collected dataset. During the data collection period, the student portal represented the primary active component of the Academic Information System, while the administrative interfaces were not yet in regular operational use. Consequently, administrative sessions constituted only a small proportion of the overall dataset and exhibited behavioral patterns that differed significantly from the dominant student activities.

Since the Isolation Forest algorithm identifies observations that deviate from the majority of behavioral patterns rather than confirmed security incidents, these administrative sessions were naturally isolated as anomalous due to their rarity within the dataset. Therefore, the detected anomalies should be interpreted as behavioral deviations from the dominant operational profile rather than evidence of malicious activities.

These findings demonstrate an important characteristic of unsupervised anomaly detection algorithms: they identify statistically unusual behavior without distinguishing between legitimate rare activities and actual cyberattacks. Consequently, the output of the proposed framework should be regarded as an early-warning mechanism that highlights sessions requiring further investigation by system administrators.

Figure 3.

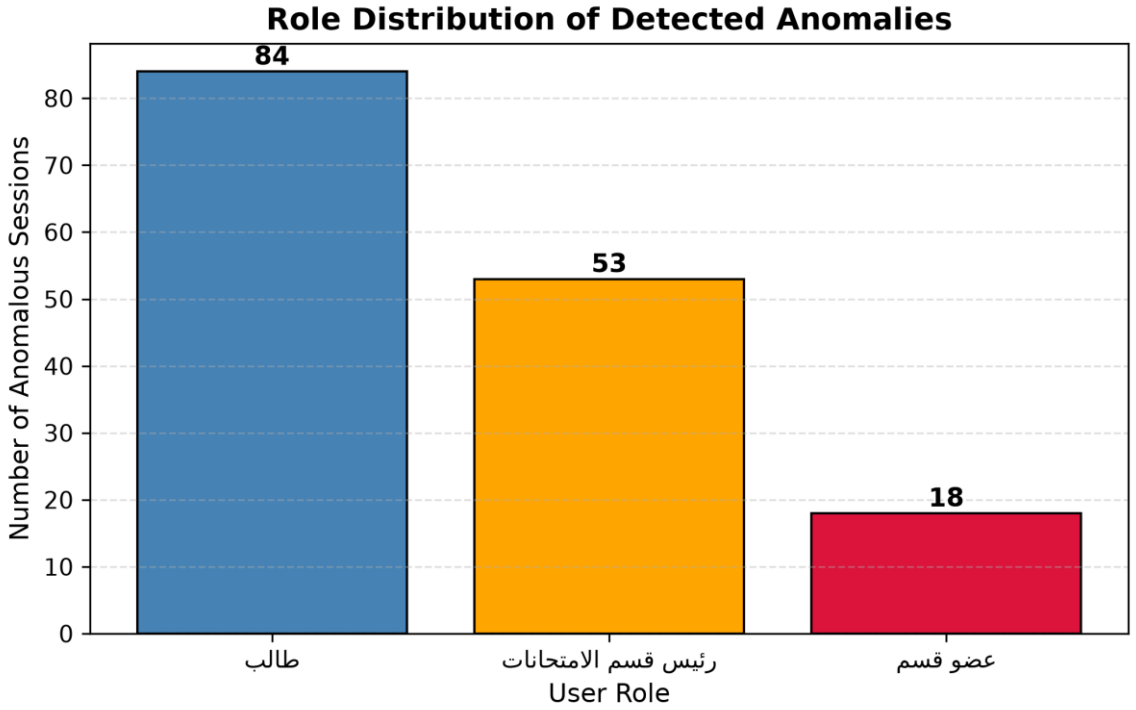

Figure 3. Distribution of detected anomalous sessions according to user roles. Student sessions account for the largest number of detected anomalies because they represent the dominant operational activity during the data collection period, while administrative roles appear as anomalous due to their relatively limited operational usage.

5.4 Behavioral Characteristics of Detected Anomalies

To gain deeper insight into the behavioral differences between normal and anomalous user sessions, a comparative statistical analysis was performed using the behavioral attributes extracted from the Academic Information System. The analysis focused on temporal characteristics, session properties, and user activity metrics that were used as input features for the Isolation Forest model.

Table 4 presents the average values of the selected behavioral features for both normal and anomalous user sessions.

Table 4. Comparison of Behavioral Features

Feature	Normal Mean	Anomaly Mean	Difference
Login Hour	11.739	10.344	-1.395
Session Duration	0.008	0.617	+0.609
Action Count	3.048	3.383	+0.335
Insert Count	3.048	2.708	-0.340
Update Count	0.000	0.675	+0.675

The comparison reveals measurable behavioral differences between normal and anomalous sessions across multiple features. Although anomalous sessions occurred, on average, approximately 1.4 hours earlier than normal sessions, the observed difference is relatively small, suggesting that login time alone was not the primary factor influencing anomaly detection.

A more pronounced difference was observed in session duration, where anomalous sessions exhibited noticeably longer durations than normal sessions. This indicates that prolonged user interactions contributed to the behavioral deviations identified by the Isolation Forest model.

Similarly, anomalous sessions recorded a slightly higher average number of user actions, suggesting increased activity intensity during these sessions. In contrast, the average number of INSERT operations was marginally lower for anomalous sessions, indicating that insertion activity alone did not characterize abnormal behavior.

The largest difference was observed for the UPDATE operation count. Normal sessions contained virtually no update operations, whereas anomalous sessions exhibited measurable update activity. This difference reflects the operational characteristics of the collected dataset rather than confirmed malicious behavior, since update operations were primarily associated with administrative users whose behavioral patterns differed from the dominant student activities during the data collection period.

Overall, these findings demonstrate that the Isolation Forest algorithm relied on the combined interaction of multiple behavioral attributes rather than any individual feature when distinguishing anomalous sessions from normal user activities.

Figure 4

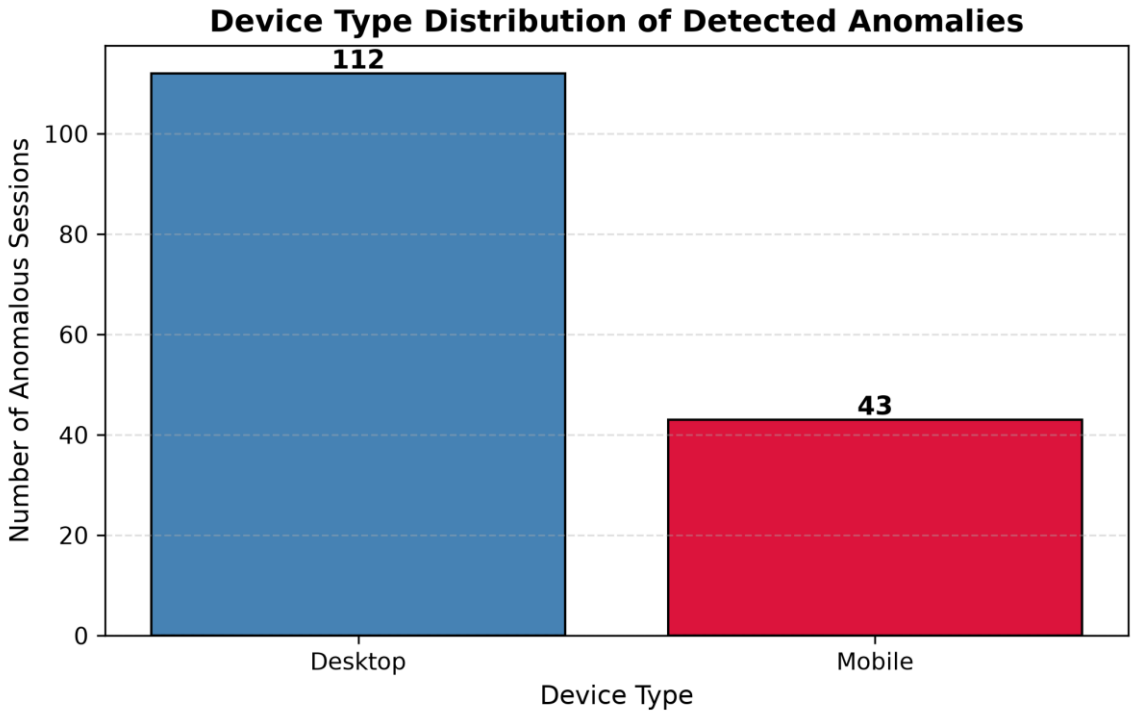

Figure 4. Distribution of detected anomalous sessions according to device type. Desktop devices accounted for most anomalous sessions (114), whereas mobile devices contributed 40 anomalous sessions, reflecting the dominant usage pattern during the experimental period.

The distribution of anomalous sessions across device types indicates that desktop devices accounted for approximately three-quarters of all detected anomalies, whereas mobile devices represented the remaining quarter. This distribution reflects the operational characteristics of the Academic Information System, which is primarily accessed through desktop computers during academic and administrative activities. Consequently, the

larger number of anomalies associated with desktop devices is consistent with the dominant usage pattern rather than indicating increased security risk associated with a specific device category.

Figure 5

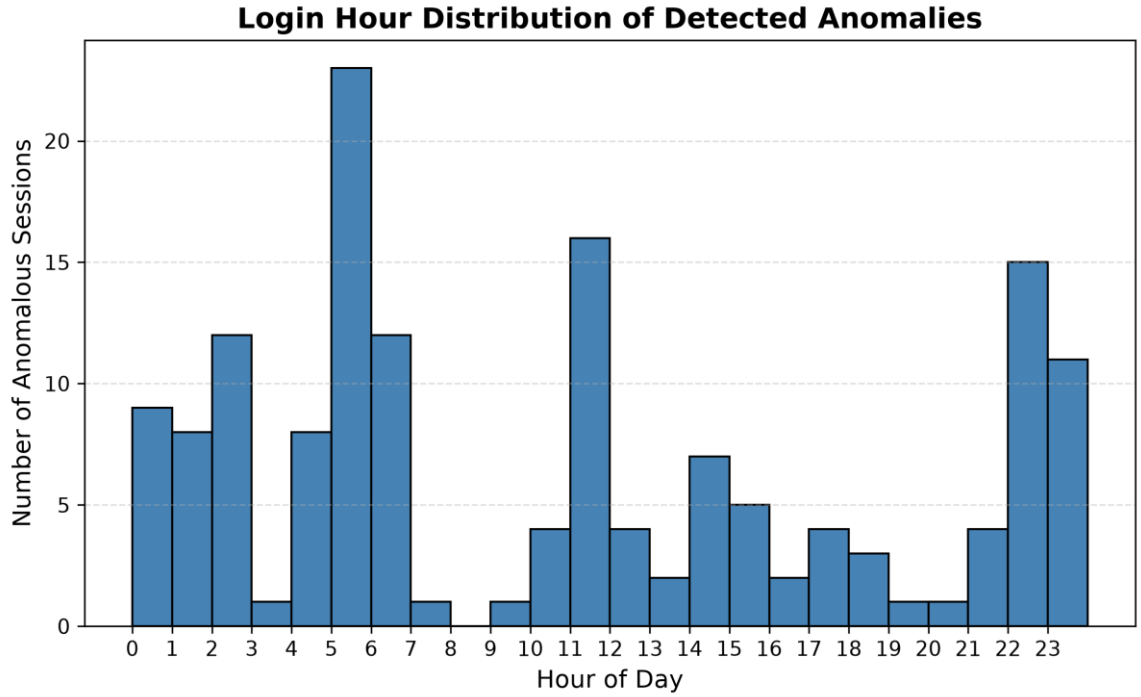

Figure 5. Distribution of detected anomalous sessions according to login hour. The histogram illustrates the temporal distribution of anomalous user activities throughout the day.

Figure 5 illustrates the temporal distribution of anomalous sessions across the 24-hour day. The detected anomalies were distributed over multiple time periods rather than concentrated within a specific interval, indicating that abnormal behavioral patterns cannot be explained solely by login time. Instead, login hour contributes as one behavioral attribute among several features considered by the Isolation Forest algorithm during anomaly detection.

The relatively broad temporal distribution also supports the multidimensional nature of the proposed behavioral analysis framework, where anomaly detection results from the interaction of temporal, operational, and behavioral characteristics rather than isolated individual features.

5.5 Isolation Forest Performance Analysis

Beyond the binary classification of user sessions into normal and anomalous categories, the Isolation Forest model produces an **anomaly score** for every behavioral observation. This score provides a quantitative measure of how strongly a user session deviates from the dominant behavioral patterns within the dataset.

Figure 6 illustrates the distribution of anomaly scores generated by the final Isolation Forest model. Positive anomaly scores correspond to sessions classified as normal, whereas negative scores indicate sessions identified as anomalous. The red dashed line represents the decision boundary (score = 0) used by the Isolation Forest algorithm.

The results demonstrate a clear separation between the two groups. Most normal sessions formed a dense cluster within the positive score region, indicating highly consistent behavioral characteristics across the majority of users. In contrast, anomalous sessions were concentrated in the negative score region, reflecting behavioral deviations successfully isolated by the model.

This separation confirms that the selected behavioral features provide sufficient discriminatory information for the Isolation Forest algorithm to distinguish abnormal behavioral patterns without requiring labeled attack data.

Figure 6

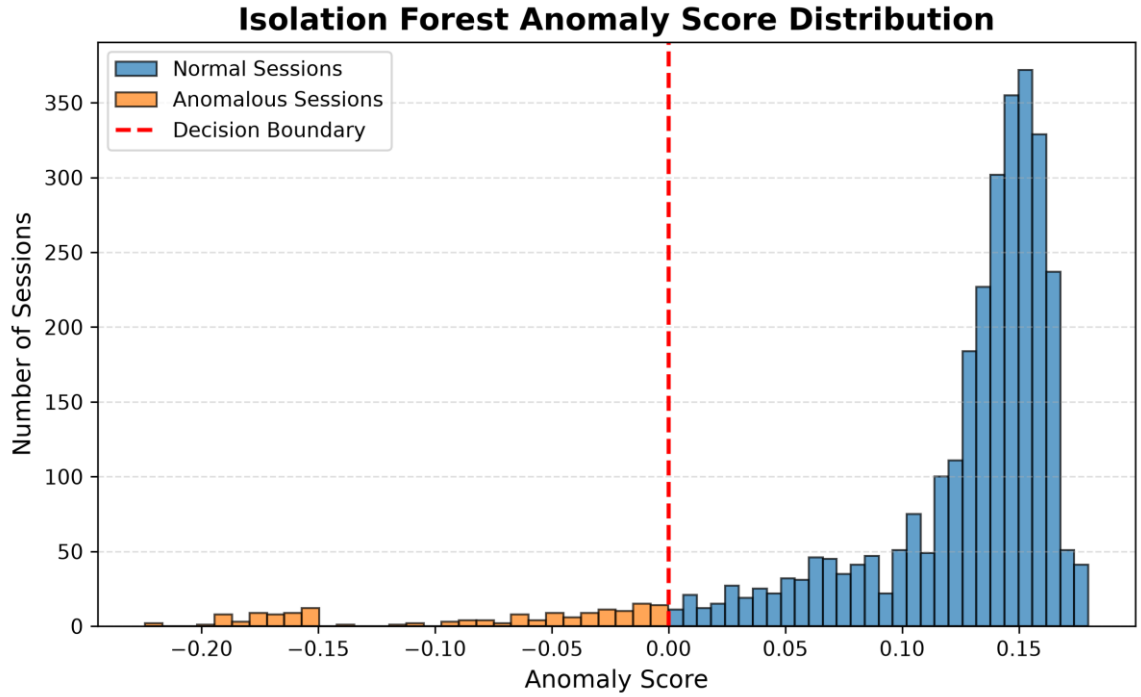

Figure 6. Distribution of Isolation Forest anomaly scores. The red dashed line represents the decision boundary (score = 0). Positive scores correspond to normal sessions, whereas negative scores indicate anomalous behavioral observations.

To evaluate the influence of the contamination parameter on anomaly detection performance, three independent experiments were conducted using contamination values of 0.01, 0.03, and 0.05. The obtained detection rates are summarized in Figure 7.

The experimental results demonstrate a proportional relationship between the contamination parameter and the number of detected anomalies. Increasing the contamination value resulted in a larger proportion of user sessions being classified as anomalous, as expected for the Isolation Forest algorithm.

Among the evaluated configurations, the experiment using contamination = 0.05 produced the most appropriate balance between anomaly sensitivity and practical interpretation of the collected behavioral data. Lower contamination values identified only a limited number of anomalous sessions, whereas the selected configuration generated a sufficiently representative set of behavioral deviations for statistical analysis while remaining consistent with the assumption that abnormal activities constitute only a small proportion of overall system behavior.

Consequently, the final experimental evaluation presented throughout this study was conducted using contamination = 0.05.

Figure 7

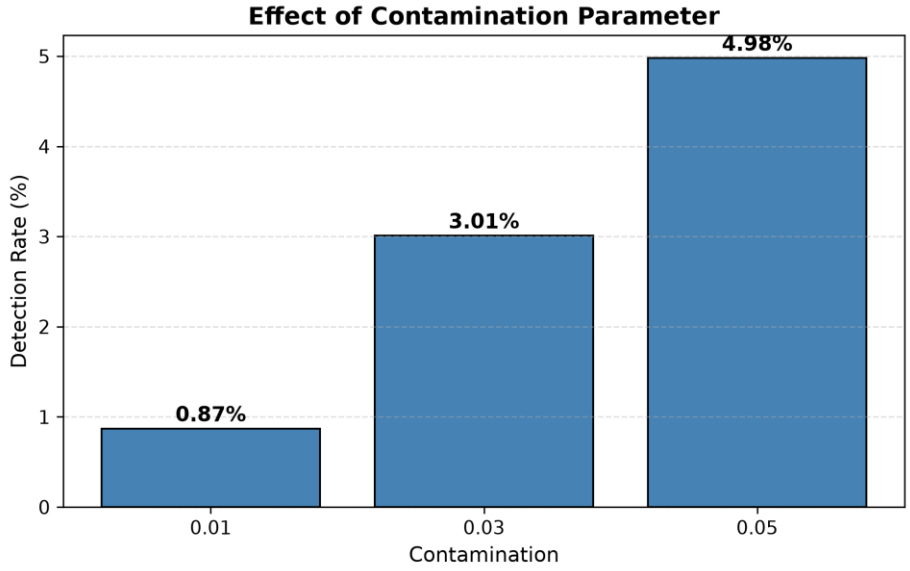

Figure 7. Effect of different contamination values on the anomaly detection rate. Increasing the contamination parameter resulted in a proportional increase in the number of sessions classified as anomalous.

5.6 Discussion

The experimental results demonstrate that the proposed behavioral anomaly detection framework can effectively identify deviations in user behavior within a real Academic Information System using an unsupervised machine learning approach. Unlike traditional intrusion detection methods that depend on labeled attack data or predefined signatures, the proposed framework relies solely on behavioral characteristics extracted from operational system logs[4], [5]. This capability makes the framework particularly suitable for academic environments where confirmed cyberattack datasets are scarce or unavailable.

An important observation of this study is that anomalous sessions should not be interpreted directly as malicious activities. Instead, the Isolation Forest algorithm identifies observations whose behavioral characteristics differ significantly from the dominant user population. Consequently, both legitimate rare activities and potentially suspicious behaviors may be classified as anomalous. This characteristic reflects the fundamental nature of unsupervised anomaly detection and highlights the importance of interpreting anomaly detection results within their operational context.

The analysis also revealed that all administrative user sessions were identified as anomalous by the proposed model. Rather than indicating malicious administrative behavior, this result reflects the operational status of the Academic Information System during the data collection period. At that stage, the student portal represented the primary active component of the system, while administrative interfaces had not yet entered full operational use. Consequently, administrative sessions represented only a small fraction of the collected dataset and exhibited behavioral characteristics that differed substantially from the dominant student activity. This observation demonstrates the importance of considering system operational context when interpreting anomaly detection results.

From a practical cybersecurity perspective, the proposed framework can serve as an early-warning mechanism for Academic Information Systems. Instead of replacing existing authentication and authorization mechanisms, the framework complements traditional security controls by continuously monitoring user behavior after successful authentication. Sessions exhibiting unusual behavioral characteristics can therefore be flagged automatically for further investigation by system administrators before potential security incidents escalate.

Despite the encouraging experimental results, several limitations should be acknowledged. First, the behavioral dataset was collected from a single Academic Information System, which may limit the generalizability of the obtained results. Second, the evaluation relied on unlabeled operational data; therefore, detected anomalies represent behavioral deviations rather than confirmed cyberattacks. Finally, only the Isolation Forest algorithm was investigated in this study. Future research may compare its performance with other unsupervised anomaly detection techniques using larger and more diverse behavioral datasets.

Overall, the obtained results demonstrate that behavioral anomaly detection based on machine learning provides a promising approach for enhancing cybersecurity within Academic Information Systems. The combination of behavioral feature engineering, operational database integration, and unsupervised anomaly detection enables the proposed framework to identify unusual user activities while requiring no labeled attack data, making it practical for deployment in real educational environments.

6. Conclusion

This study presented a machine learning-based framework for detecting anomalous user behavior in Academic Information Systems using the Isolation Forest algorithm. Unlike traditional security approaches that depend primarily on authentication mechanisms or labeled attack datasets, the proposed framework analyzes behavioral information extracted from user sessions, audit logs, and user role data to identify behavioral deviations in an unsupervised manner.

A behavioral dataset containing 3,090 authenticated user sessions was constructed from a real Academic Information System through the integration of multiple operational database tables into a unified analytical dataset. After feature engineering and data preprocessing, the Isolation Forest model was trained using 12 behavioral features, enabling the detection of anomalous user sessions without requiring labeled cybersecurity incidents.

The experimental evaluation demonstrated that the proposed framework successfully distinguished normal and anomalous behavioral patterns. Using the final model configuration with a contamination value of 0.05, the model identified 155 anomalous sessions, corresponding to a detection rate of 5.02%. Additional statistical analyses showed that anomaly detection resulted from the combined interaction of multiple behavioral attributes rather than any individual feature alone.

The findings also emphasize the importance of interpreting anomaly detection results within their operational context. Some detected anomalies reflected legitimate but infrequent administrative activities rather than malicious behavior, illustrating one of the fundamental characteristics of unsupervised anomaly detection algorithms. Consequently, the proposed framework should be considered an early-warning behavioral monitoring mechanism that assists system administrators in identifying sessions requiring further investigation rather than a standalone intrusion detection solution.

Overall, the proposed framework demonstrates that behavioral analysis combined with unsupervised machine learning provides a practical and effective approach for enhancing cybersecurity in Academic Information Systems. By utilizing real operational data and requiring no labeled attack samples, the framework offers a promising solution for continuously monitoring user behavior in educational environments.

Future research may extend this work by evaluating additional anomaly detection algorithms, incorporating larger datasets collected from multiple educational institutions, and integrating real-time behavioral monitoring capabilities to improve detection accuracy and practical deployment.

7. References

[1] F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation Forest,” in Proceedings of the 8th IEEE International Conference on Data Mining (ICDM), Pisa, Italy, 2008, pp. 413–422.

[2] F. T. Liu, K. M. Ting, and Z.-H. Zhou, “Isolation-Based Anomaly Detection,” ACM Transactions on Knowledge Discovery from Data, vol. 6, no. 1, pp. 1–39, Mar. 2012.

[3] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly Detection: A Survey,” ACM Computing Surveys, vol. 41, no. 3, pp. 1–58, Jul. 2009.

[4] A. L. Buczak and E. Guven, “A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection,” IEEE Communications Surveys & Tutorials, vol. 18, no. 2, pp. 1153–1176, 2016.

[5] M. Ahmed, A. N. Mahmood, and J. Hu, “A Survey of Network Anomaly Detection Techniques,” Journal of Network and Computer Applications, vol. 60, pp. 19–31, Jan. 2016.

[6] P. A. Legg, O. Buckley, M. Goldsmith, and S. Creese, “Automated Insider Threat Detection System Using User and Role-Based Profile Assessment,” IEEE Systems Journal, vol. 11, no. 2, pp. 503–512, Jun. 2017.

[6] Alnnale, T. (2026). Predictive Governance in Digital Enterprises: An LSTM-Enhanced Deep Learning Framework for Economic Optimization of IT Incident Management Using Enriched Process Logs. *Al-Farooq Journal of Sciences, 2*(3), 86-113.