

Insider Threat Detection Using Machine Learning

C. Vani Shri¹, Karunya V², Keerthana B³,
L. Neanthara⁴, Dr. R. Ahila⁵

*Student^{1,2,3,4}, Associate Professor⁵, Department of Computer Science and Engineering,
School of Engineering, Avinashilingam institute for Home Science
and Higher Education for Women*

Abstract:

Insider threats represent one of the most challenging cybersecurity risks for modern organizations because they originate from users who already possess authorized access to critical systems, applications, and sensitive information. These threats may occur due to malicious intent, negligence, or accidental misuse of resources, and they are often difficult to identify because the activities performed by insiders can closely resemble legitimate user behaviour. Traditional security mechanisms such as firewalls and signature-based intrusion detection systems mainly focus on external attacks and may fail to detect abnormal actions carried out by trusted users. Therefore, intelligent approaches that analyze user behaviour patterns are required to improve early detection and prevention of insider-related security incidents.

This research proposes a machine learning-based framework for insider threat detection using behavioural analysis of user activity logs. The proposed system analyzes various behavioural attributes, including login patterns, file access activities, system command execution, and network usage information, collected from a publicly available corporate activity dataset. A systematic preprocessing approach is applied to improve data quality by handling missing values, transforming categorical attributes, normalizing numerical features, and selecting the most significant behavioural indicators that contribute to threat identification.

The proposed behavioural analytics approach demonstrates that machine learning techniques can effectively identify unusual user activities and provide a proactive method for insider threat management. The framework offers a scalable and interpretable solution for organizations seeking to strengthen their cybersecurity infrastructure. Furthermore, the proposed system provides a foundation for future improvements, including real-time threat monitoring, deep learning-based behavioural modelling, and integration with advanced security information and event management platforms.

Key Words: Insider Threat Detection, Machine Learning, Behavioural Analytics, Anomaly Detection, Random Forest, User Activity Monitoring, Cybersecurity, Feature Engineering, Threat Intelligence, Predictive Security Analytics.

1. Introduction

Contemporary organizations rely heavily on digital infrastructure, cloud platforms, and internal networks to manage sensitive information, which makes information security a fundamental operational requirement rather than an optional safeguard. While much of the security effort within an enterprise is directed outward, against attackers who do not already hold access to the network, a substantial share of security incidents originate from within the organization itself.

An insider threat arises when an individual who has been granted legitimate access misuses that access, either deliberately or through negligence, resulting in data breaches, financial loss, or erosion of institutional trust. Because the actions involved are carried out through authorized accounts, they are considerably harder to distinguish from ordinary work activity than actions originating from an external, unauthenticated source.

Conventional security mechanisms, including firewalls, signature-based intrusion detection systems, and access-control lists, are designed primarily to keep unauthorized parties out of a network. These mechanisms offer little visibility into the behaviour of users who are already inside the perimeter, and they are largely ineffective at recognizing when a trusted account begins to act in a way that deviates from its normal pattern of use.

Behavioural analysis addresses this gap by continuously observing how users interact with systems, files, and networks. Attributes such as login timings, the volume and type of files accessed, the commands issued, and the pattern of network usage together form a behavioural profile

for each user, and a meaningful departure from this profile can be treated as a candidate indicator of risk.

Machine learning is well suited to this task because it can process large volumes of behavioural log data and learn the boundary between normal and anomalous activity directly from examples, rather than relying on a fixed set of hand-written rules. This project accordingly develops an intelligent detection framework that classifies user activity as either normal or suspicious, with the goal of supporting earlier identification of insider threats and strengthening the overall security posture of the organization.

2. Literature Survey

Rushudha V. [1] proposed an AI-based behavioural analytics approach for insider threat detection in which activity logs were preprocessed, behavioural features were engineered, and a Random Forest classifier was used to flag anomalous activity. The approach was reported to provide accurate and interpretable detection with a comparatively low rate of false alarms, although its effectiveness remains dependent on the availability of representative historical behavioural patterns and on the responsiveness of the system under real-time conditions.

Muzaffar and Mazher [2] examined an AI-powered behavioural analysis technique for insider threat detection within enterprise networks, relying on anomaly-based profiling of user actions. The study highlighted practical obstacles associated with this line of work, including inconsistent data quality, privacy considerations surrounding the logging of user behaviour, a persistent tendency toward false positives, and the difficulty of

integrating such a system into existing enterprise infrastructure.

Falana [3] investigated a hybrid anomaly-detection approach for financial fraud that combined Graph Neural Networks with time-series analysis and feature fusion. Although the study was oriented toward financial fraud rather than insider threats specifically, it demonstrated that fusing structural and temporal behavioural signals can improve robustness on imbalanced and dynamic datasets, a consideration that is directly relevant to behaviour-driven insider threat detection.

Qawasmeh and AlQahtani [4] proposed a machine learning framework for real-time insider threat identification and user profiling that moved beyond static, firewall-centred defence. Their work emphasized continuous behavioural profiling as a means of catching threats that perimeter defences overlook, while noting that sustaining real-time performance at scale remains a significant engineering challenge.

Ye et al. [5] explored insider threat detection through personalized federated learning combined with behaviour log analysis, allowing models to be trained across distributed organizational data without centralizing sensitive logs. This preserved data privacy across departments but introduced additional complexity in coordinating model updates and maintaining detection consistency across heterogeneous client behaviour.

Pennada et al. [6] compared behavioural analysis carried out through classical machine learning against deep learning techniques for insider threat detection, concluding that ensemble machine learning methods offered a favourable balance of accuracy and interpretability for

moderately sized behavioural datasets, whereas deep learning approaches required considerably larger volumes of labelled data to achieve comparable reliability.

Yumlembam et al. [7] proposed a graph-based behavioural modelling approach combining Graph Convolutional Networks with Bi-LSTM layers to capture both relational and sequential aspects of user behaviour. The model captured richer behavioural context than feature-based classifiers but at the cost of higher computational overhead and reduced transparency in how individual predictions were reached.

Taken together, the reviewed literature indicates that rule-based and signature-based systems are inherently limited in adapting to evolving user behaviour, that ensemble machine learning models such as Random Forest offer a strong trade-off between detection accuracy and interpretability for structured activity-log data, and that more complex graph- or deep-learning-based methods, while promising, demand larger datasets and greater computational resources than a mini-project setting typically affords. These observations motivate the present work, which adopts a machine learning-based behavioural analysis framework built around classical classifiers and structured feature engineering.

Building upon these findings, the proposed framework focuses on identifying behavioural anomalies through the analysis of structured user activity data using supervised machine learning techniques. By emphasizing effective feature engineering and the use of computationally efficient classifiers, the system aims to distinguish normal user behaviour from potentially malicious activities with improved accuracy. This approach provides a practical balance between detection performance, model interpretability, and

implementation complexity, making it well suited for academic research and small-scale organizational environments.

Table 1. Comparison of Existing and Proposed System

Feature	Existing System	Proposed System
Detection approach	Rule-based / signature-based monitoring	Machine learning-based behavioural analysis
Adaptability	Limited to predefined, known patterns	Learns and adapts to evolving behaviour
False positive rate	Comparatively high	Reduced through learned feature-based classification
Real-time monitoring	Limited	Supported through a web-based dashboard
Feature utilization	Manual rules, minimal feature use	Automated feature engineering and selection
Scalability	Constrained by manual configuration	Improved via an automated ML pipeline
User interface	Minimal, log-based output	Interactive dashboard with visual alerts

3. Proposed System and Methodology

The proposed system is designed to detect insider threats by analysing behavioural attributes

drawn from user activity logs rather than relying on fixed rules or known attack signatures. User activity data, including login times, file access records, system commands, and network usage, is first collected and consolidated. This raw data is then cleaned and transformed through a preprocessing stage before being reduced to a compact set of behavioural features that capture how each user typically interacts with organizational systems.

A supervised machine learning model is trained on the processed features to classify each activity record as either normal or anomalous. When an anomaly is identified, the system generates an alert, and the results are surfaced through a web-based dashboard that allows an administrator to review flagged activity, examine the overall distribution of normal versus suspicious behaviour, and export the analysed data for further investigation. The overall workflow, from raw log ingestion through to alert generation, is summarized in Fig. 1.

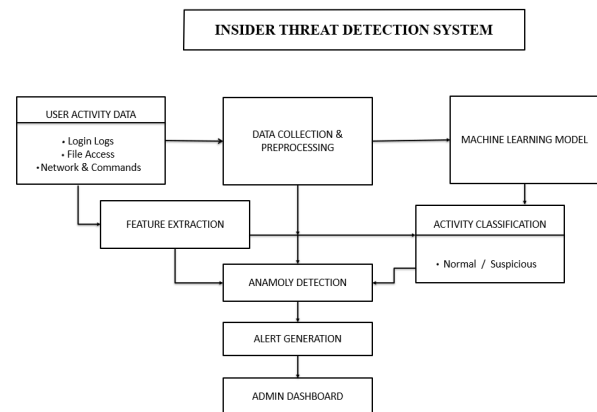


Fig. 1. Block diagram of the proposed insider threat detection system

4. System Implementation

The proposed system is organized into five cooperating modules: data collection and preprocessing, feature engineering, model training,

classification and anomaly detection, and the web-based dashboard. Each module is described in the subsections that follow.

4.1 Data Collection and Preprocessing

The dataset used for this project is drawn from a publicly available Kaggle repository of insider threat activity and represents structured user activity logs from a simulated corporate environment [8]. Each record includes attributes such as user identifier, login time, file access behaviour, system commands issued, and network activity, together with an indication of whether the associated behaviour is normal or malicious.

The raw dataset is first inspected for inconsistencies, missing values, and duplicate or irrelevant records. Missing entries are addressed through removal or imputation, depending on the extent and nature of the gaps, and duplicate or irrelevant rows are discarded to keep the input data reliable. Categorical attributes, such as user roles and activity types, are then converted into a numerical representation using label encoding or one-hot encoding, since the downstream classification models require numerical input. Link: <https://www.kaggle.com/code/alabiobin/insider-threat-detection>

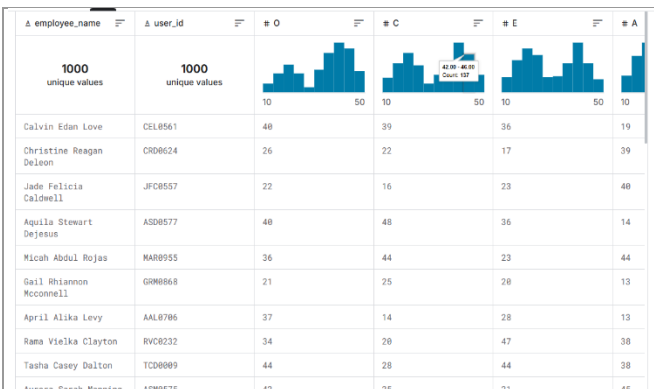


Fig. 2. Sample of the insider threat dataset obtained from Kaggle

Numerical attributes are scaled through normalization to place all features on a comparable range and to aid model convergence during training. Time-based fields, such as login timestamps, are transformed into derived attributes including session duration and access frequency, which are more directly informative of behavioural patterns than raw timestamps. The processed dataset is then split into training and testing subsets to support unbiased evaluation, and the balance between normal and malicious records is examined so that any class imbalance can be accounted for during training. Together, these steps ensure that the data supplied to the learning algorithms is clean, consistent, and representative of genuine behavioural variation.

4.2 Feature Engineering

Following preprocessing, relevant behavioural attributes are extracted from the cleaned log data to represent each user's activity pattern in a compact form. Features such as login frequency, the number of files accessed, and the volume of data downloaded are computed, and time-based attributes are converted into higher-level indicators such as session duration and activity trend. Features that are redundant or contribute little discriminative value are removed to reduce dimensionality and improve computational efficiency, leaving a focused feature set that is expected to strengthen the accuracy of the subsequent classification stage.

4.3 Model Training

The engineered feature set is used to train supervised classification models that learn the relationship between behavioural attributes and the corresponding activity label. Three classical algorithms, namely Random Forest, Logistic

Regression, and Decision Tree, are employed for this purpose, reflecting a balance between predictive performance and interpretability that is appropriate for behavioural log data of this scale. Model hyperparameters are tuned to improve prediction accuracy while limiting the risk of overfitting, and performance is subsequently assessed on the held-out testing data to confirm that the learned patterns generalize beyond the training set.

4.4 Classification and Anomaly Detection

During inference, the trained model receives the behavioural features associated with a given activity record and classifies it as either Normal or Suspicious. Activities that depart noticeably from established behavioural norms, such as unusually high file access counts or unusually large data downloads, are flagged as anomalies. Key behavioural attributes, including login time, number of files accessed, and volume of data downloaded, serve as the primary input features for this classification step, and any record identified as suspicious is passed on to the alert-generation stage for administrator review.

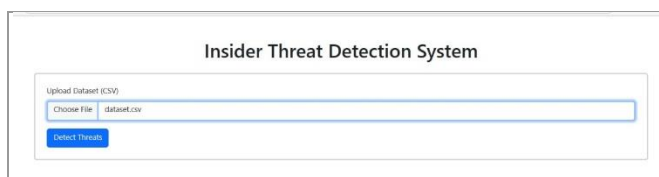


Fig. 3. Insider threat detection dashboard showing classified activity

4.5 Web Application (Dashboard)

The trained model is embedded within an interactive web application built using Flask/Streamlit, allowing a user to upload an activity dataset and receive analysis results without needing direct access to the underlying code. Once

a file is uploaded, it is processed dynamically, enabling near real-time interaction between the administrator and the detection system. The application generates a graphical summary of the distribution between normal and suspicious activity, which supports rapid visual interpretation of the overall threat level, and it presents detailed per-record results that can be downloaded for further investigation and reporting. A working prototype of the application has been deployed and made accessible online, demonstrating the practical feasibility of the proposed approach beyond a purely offline experimental setting.

In addition to providing prediction results, the application is designed with usability and scalability in mind. The intuitive interface enables administrators to monitor user activity efficiently, review detection outcomes, and analyze suspicious behaviour without requiring specialized technical expertise. The modular architecture also allows future enhancements, such as integrating advanced machine learning algorithms, real-time log monitoring, automated alert notifications, and database connectivity, making the system adaptable to evolving organizational security requirements.

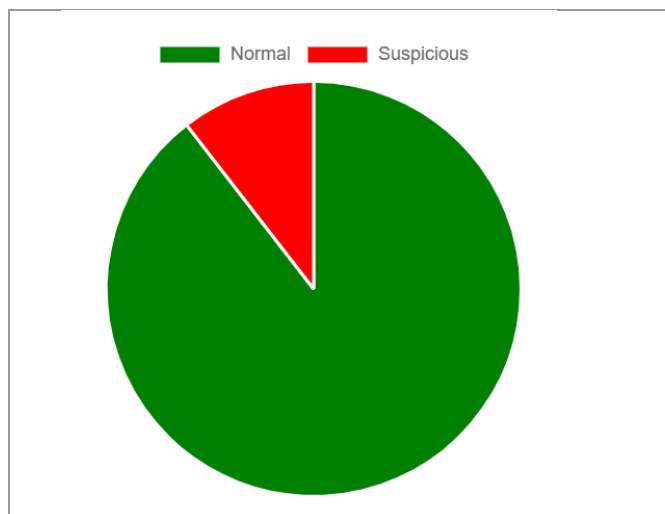


Fig. 4. Insider threat analysis output generated by the dashboard

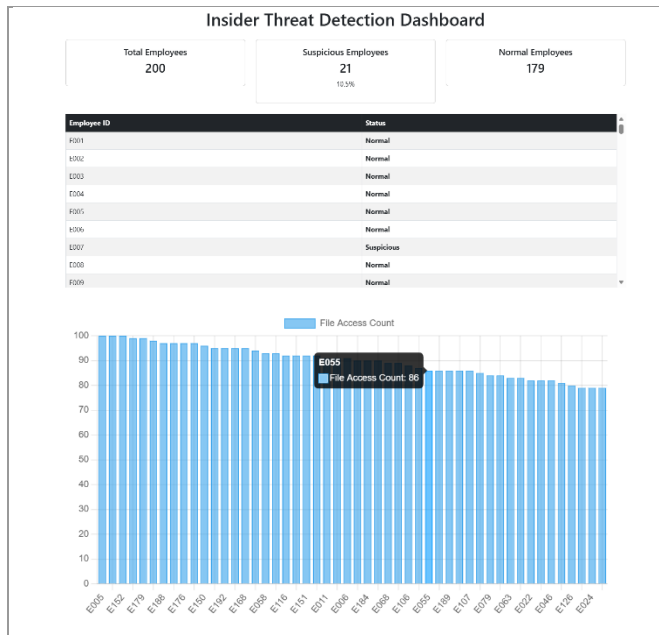


Fig. 5. Deployed prototype of the insider threat detection dashboard

5. Role of Machine Learning in the Project

Machine learning occupies a central role in the proposed system because insider threat detection ultimately depends on distinguishing subtle, evolving behavioural deviations from the normal, everyday variation that is inherent to genuine user activity. Rule-based approaches struggle in this setting because they can only encode behaviours that have already been anticipated and written into a rule, whereas a trained classifier can generalize from the patterns present in historical data to activity it has not explicitly seen before.

By training on engineered behavioural features rather than raw log entries, the Random Forest, Logistic Regression, and Decision Tree models used in this project are able to weigh multiple behavioural indicators simultaneously, such as login timing, access volume, and command usage, when deciding whether an activity record is

normal or suspicious. The ensemble nature of the Random Forest classifier in particular allows it to combine the judgments of many individual decision trees, which tends to make its predictions more stable and less sensitive to noise in any single feature than a single decision tree would be. This capacity to learn from data, rather than from a fixed set of rules, is what allows the system to adapt as user behaviour and organizational usage patterns change over time.

6. Results and Discussion

6.1 Performance Analysis

The trained classification models were evaluated on the held-out portion of the Kaggle insider threat dataset using standard classification metrics, including accuracy, precision, recall, and F1-score. Among the three algorithms considered, the Random Forest classifier exhibited more consistent and stable classification behaviour than the Logistic Regression and Decision Tree baselines, which is broadly consistent with its ensemble structure and with its established suitability for behavioural datasets that contain some degree of class imbalance between normal and malicious activity.

A degree of misclassification was observed at the boundary between borderline-normal and mildly suspicious activity, which is expected given that certain legitimate but unusual user actions can resemble the early indicators of malicious behaviour. This observation suggests that feature engineering, and in particular the derivation of session-level and frequency-based attributes, plays an important role in helping the classifiers separate these closely related cases.

6.2 Output Visualization

The deployed web application successfully processed uploaded activity datasets and returned classification results through its dashboard interface. The dashboard presents the predicted distribution of normal and suspicious records as a graphical summary, alongside a detailed, downloadable table of per-record predictions, making the output straightforward for an administrator to interpret and act upon. Overall, the results indicate that a behavioural, machine learning-based approach can provide a practical and interpretable alternative to purely rule-based insider threat monitoring.

7. Future Scope

While the current system demonstrates the practicality of machine learning-based insider threat detection, several extensions could further strengthen its capability. Integration with real-time monitoring pipelines would allow continuous streams of user activity to be analysed as they occur, rather than in batch uploads, enabling more immediate detection and response.

Model performance could be enhanced by incorporating deep learning techniques capable of capturing more complex, non-linear behavioural patterns than the classical classifiers used in this work. Similarly, incorporating multi-source data, such as email metadata, system logs, and network traffic, alongside the current activity attributes could provide a more comprehensive behavioural profile for each user.

Adaptive learning models that update as user behaviour evolves over time could help reduce false positives that arise from legitimate but non-

stationary changes in work patterns, and deployment in a cloud-based environment would support scalability for larger organizations. Finally, integration with enterprise security infrastructure, such as SIEM platforms, would allow the detection framework to feed into centralized monitoring and automated incident-response workflows rather than operating as a standalone tool.

8. Conclusion

This project has presented a machine learning-based framework for insider threat detection that analyses user activity logs to distinguish normal behaviour from potentially malicious activity. By combining structured data preprocessing, behavioural feature engineering, and classical supervised classifiers such as Random Forest, Logistic Regression, and Decision Tree, the system is able to identify deviations from established behavioural norms with a level of adaptability that static, rule-based monitoring cannot easily match.

The accompanying web-based dashboard makes the system practically usable, allowing an administrator to upload activity data, review classification results, and export flagged records for further investigation, all without requiring specialized technical expertise. In doing so, the proposed approach reduces reliance on manual monitoring, supports more timely identification of suspicious activity, and offers organizations a proactive tool for strengthening internal security.

At the same time, the system's reliance on a single, Kaggle-sourced dataset means that its behaviour on real-world organizational logs, which may differ considerably in scale, diversity, and labelling quality, has not yet been established, and

its use of classical machine learning models, while interpretable, may ultimately be limited in capturing more subtle or evolving behavioural patterns than deep learning-based alternatives could. Addressing these limitations, along with the extensions outlined in the future scope, represents a natural next step for continued development of the system.

References

- [1] R. V, "Insider Threat Detection Using AI-Based Behaviour Analytics," *International Journal of Innovative Research in Technology (IJIRT)*, 2025.
- [2] J. Muzaffar and N. Mazher, "AI-Powered Behavioral Analysis for Insider Threat Detection in Enterprise Networks," *BJMR Journal*, 2023.
- [3] A. Falana, "AI-Driven Anomaly Detection for Financial Fraud: A Hybrid Approach Using Graph Neural Networks and Time-Series Analysis," *JOSTER Journal*, 2024.
- [4] S. A. Qawasmeh and A. A. S. AlQahtani, "Beyond Firewall: Leveraging Machine Learning for Real-Time Insider Threat Identification and User Profiling," *Future Internet*, vol. 17, no. 2, 2025.
- [5] X. Ye et al., "Research on Insider Threat Detection Based on Personalized Federated Learning and Behavior Log Analysis," *Scientific Reports*, vol. 15, 2025.
- [6] S. S. P. Pennada, S. K. Nayak, and V. K. M., "Insider Threat Detection Using Behavioural Analysis through Machine Learning and Deep Learning Techniques," *International Research Journal of Multidisciplinary Technovation*, vol. 7, no. 2, 2025.
- [7] R. Yumlembam et al., "Insider Threat Detection Using GCN and Bi-LSTM with Graph-Based Behavioral Modeling," *arXiv preprint*, 2025.
- [8] A. Uzaki, "Insider Threat Dataset for Corporate Environments," *Kaggle*, 2023. [Online]. Available: <https://www.kaggle.com/datasets/ahmeduzaki/insider-threat-dataset-for-corporate-environments>
- [9] M. Bishop and C. Gates, "Defining the Insider Threat," *Proceedings of the 4th Annual Workshop on Cyber Security and Information Intelligence Research*, Oak Ridge, TN, USA, 2008.
- [10] E. M. Hutchins, M. J. Cloppert, and R. M. Amin, "Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains," *Leading Issues in Information Warfare & Security Research*, vol. 1, no. 1, pp. 80–106, 2011.
- [11] R. Mitchell and I. R. Chen, "Behavior Rule Specification-Based Intrusion Detection for Safety Critical Medical Cyber Physical Systems," *IEEE Transactions on Dependable and Secure Computing*, vol. 12, no. 1, pp. 16–30, Jan. 2015.
- [12] A. Tuor, S. Kaplan, B. Hutchinson, N. Nichols, and S. Robinson, "Deep Learning for Unsupervised Insider Threat Detection in Structured Cybersecurity Data Streams," *AAAI Workshops*, 2017.