Understanding the influence of behaviour monitoring tools towards mitigating insider threats in hybrid working environments through behavioural analytics

MSc Research Project
MSCCYB1JAN25B_O

Sravan Kumar Murimuri
Student ID: 23358556

School of Computing
National College of Ireland

Supervisor: Jawad Salahuddin

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Sravan Kumar Murimuri ……
Student ID:	23358556 …… ……..……
Programme:	MSCCYB1JAN25B_O …… **Year:** 2025 - 2026 …… ……..
Module:	MSc (Research) Practicum Part 2 …… ……..……
Supervisor:	Jawad Salahuddin …… ……..……
Submission Due Date:	28/01/2026 …… ……..……
Project Title:	Understanding the influence of behaviour monitoring tools towards mitigating insider threats in hybrid working environments through behavioural analytics …… ……..……
Word Count:	6,811…… **Page Count** …23…

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

M. Sravan kumar

Signature: …… ……

Date: 28/01/2026……
……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Understanding the influence of behaviour monitoring tools towards mitigating insider threats in hybrid working environments through behavioural analytics

Student Name: Sravan Kumar Murimuri
Student ID: 23358556

Abstract

Hybrid working arrangements would pose a major challenge to the insider threat detection tools since the traditional monitoring tools are not very effective in detecting malicious activities in work environments that are dispersed. In 2023, organisations have suffered severe consequences due to insider threats, 45% led to data loss and 43% to brand damage, but less than one third of the organisations perceived that they were well prepared to counter threats. The study aimed to assess the efficiency of machine learning models (Random Forest, Stacking Classifier, and XGBoost) enhanced by behavioural analytics to detect insider threats and compare their results in relation to improving the model of child enhancement accuracy and reducing false positives in the hybrid workplaces. The research employed a positivist philosophy and the deductive approach based on the Kaggle Insider Threat dataset. Pre-processing of data involved the management of missing data, outliers, class balancing using SMOTE, and multicollinearity filtering. The accuracy, precision, recall, F1-score, and AUC were used to measure, train, and evaluate three ensemble models. XGBoost was the most suitable model, with 99.20% accuracy, 100% recall, and 98.57% F1-score, followed by Random Forest (99.07% accuracy, 97.58% F1-score) and Stacking Classifier (98.67% accuracy, 97.58% F1-score). The results show that the use of ML to enhance behavioural monitoring performs well in detecting insider threats in the hybrid environment, as it is more effective than the available methods reported in the literature, and provides support to the use of ensemble methodologies in organisational cybersecurity resilience.

1 Introduction

1.1 Background

In hybrid working arrangements, remote and office locations create challenges when it comes to the issue of insider threats, as it is difficult to monitor. In 2023, insider threats result in serious organisational consequences as 45% of these threats lead to loss of vital data, 43% to brand damage, and 41% to interference with working processes, which explains the necessity to improve portfolio monitoring in hybrid workplaces (Smith, 2024). Although this growth was experienced, less than 30% of companies feel adequate to counter these risks (Istari, 2023). Not only do insider incidents lead to breaches, but also operations (41%) and brand reputation (43%), which means that a complex behaviour monitoring system is required to address machine learning (Enterprise IT World, 2024). Thus, behaviour monitoring tools can also be used together with ML as an essential route to effective insider threat detection and protecting the integrity of organisations in the context of hybrid working.

Figure 1: Impacts of Insider Threat Incidents

(Source: Smith, 2024)

1.2 Problem Statement

The insider threats present in hybrid workplaces are convenient and hard to identify because the existing monitoring tools are not as effective as they need to be. This causes data breaches, disruption of operations, and reputational damage (Inayat et al., 2024). Thus, this analytical study becomes a pressing requirement to develop behavioural analytics powered by ML to improve detection and mitigation.

1.3 Research Aim, Objective, and Question

1.3.1 Research Aim

- This study aims to investigate the impact of behaviour surveillance tools, supplemented with machine learning (ML) and behavioural analytics, on insider threat detection and reduction in a hybrid workplace.

1.3.2 Research Objective(s)

- To evaluate the effectiveness of ML models (Random Forest, Stacking Classifier, and XGBoost) that are supplemented by behavioural surveillance technologies in the detection and prevention of insider threats in hybrid workplaces.
- To compare the effectiveness of the integration of behaviour profiling and technical classification models in enhancing accuracy, minimising false positives, and building organisational resilience against insider risks.

1.3.3 Research Question

- What impact do behaviour monitoring tools like machine learning enhanced with behavioural analytics have on the effectiveness of insider threat detection and mitigation in hybrid working environments?

1.4 Research Contribution

This study is valuable as it implements and tests a machine learning-based enhanced behavioural monitoring model. This model enhances the use of insider threats by increasing detection sensitivity, minimising errors, and delivering adaptive and practical insights. Additionally, this study also contributes to the development of data and operational security in the conditions of a dynamic hybrid working environment.

1.5 Research Assumptions

The study makes the following assumptions such that Kaggle Insider Threat data corresponds to the real situation, hybrid working conditions, ML models identify the behavioural anomalies effectively, and, after implementation of the behavioural analytics, the detection performance is going to be higher, and false positives in this case are reduced.

1.6 Research Limitation

The study has drawbacks in the form of Kaggle secondary data, based on which it may not entirely represent real-world dynamic hybrid environments. Performance in models can be different in organisations, and it can be a barrier to an effective behavioural analytics solution due to the complexity of computation or a lack of interpretation of the performance of the model.

1.7 Structure of the Study

The structure of the dissertation, with a detailed overview, is stated in **Figure 2.**

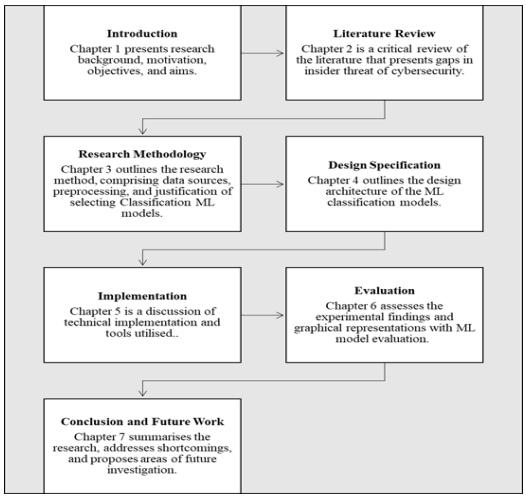


Figure 2: Structure of the Dissertation

2. Related Work

2.1 Introduction

The chapter is a review of the literature regarding insider threats in digital and hybrid workplaces, covering the development of threats, the application of behaviour checking and behavioural analytics, and machine learning models. This chapter contains a critical assessment of previous literature related to this research area for a detailed overview of past research studies. It also determines theory foundations, literature gaps, and a conceptualisation of this research.

2.2 Evolution of Insider Threats in Digital and Hybrid Workplaces

Insider threats have evolved in a variety of perspectives that relate to the compounding of the special threats by the digital and hybrid environments. Chawande (2024) sought to investigate insider risks within highly automated AI systems in a study that utilised case studies, cyberattack analysis, and the advice of consultants. The results of that study shed light on the fact that automation loopholes were used by actual users, and thus, the AI-powered defences are less effective and need further reinforcement with better behaviour-focused execution monitoring.

Conversely, Yi and Tian (2024) explored the hybrid algorithm approach to insider threats using the supervised and unsupervised approaches to learning. The approach achieved the highest predictive accuracy through outlier scoring models like KNN, LOF, and Isolation Forest, and 86.12% is the best performance at lower computational costs. Although Chawande (2024) concentrated on socio-technical vulnerability that was developed due to automation, whereas Yi and Tian (2024) offered algorithmic refinement as an anomaly detection goal. In both research findings, it has become clear that insider threats are now more advanced, and efficient mitigation of the threats has necessitated dynamically flexible approaches to data. Thus, for that reason, this study has been an ongoing development in insider threats in terms of subtlety.

2.3 Behaviour Monitoring and Behavioural Analytics of Insider Threats in Cybersecurity

Fusion of behavioural analytics and old-fashioned security monitoring is a paradigm shift in the insider threat or insider detection approaches. It has examined behaviour monitoring and behavioural analytics as critical processes for enhancing cybersecurity against insider threats. Ogunbodede et al. (2024) sought to determine the potential operationalisation of user behaviour profiling based on real-time log observations and feature anomaly detection. The approach was based on empirical simulation and patterns of log systems and communication metadata to detect deviations, and based on their results, found a better detection accuracy, but expressed resource issues in terms of scalability and widespread ethical observation. Modern strategies use the user and entity behaviour analytics solutions that develop a profile baseline behavioural pattern of each employee and continuously scan the account to identify irregularities that can suggest malicious motives or hacked credentials. Nevertheless, Richard et al. (2025) aimed to form a multi-dimensional system merging the technical ones, including access logs and endpoint monitoring, and human-related data, i.e., psychological profiling and workplace trends. These systems process a wide range of data streams such as overall authentication patterns, file access patterns, email patterns and network traffic in order to make detailed behavioural patterns. Based on this study, the combination of machine learning and technical and human signals also led to lower false positives at 40%, providing more preemptive risk mitigation. Conversely, Ogunbodede et al. (2024) focused on technical information, whereas, on the other hand, Richard et al. (2025) continued holistic fusion. From this perspective, behavioural analytics has become a more adaptive and full-fledged force of cybersecurity. The quality of such systems depends upon the critical issue of the quality of the establishment of the base and sensitivity of the anomaly detection limits, which needs to be balanced between the needs of security and the needs of the operations efficiency and users' experience.

2.4 Machine Learning Models for Insider Threat Detection

Insider threat detection has had a large-scale application of machine learning models that offer varying methodological approaches. The purpose of this study by Sarhan and Altwaijry (2022) was to detect the improvement with deep feature synthesis and classification and anomaly detection models on the CERT data. They used this method to extract features (69,738), which resulted in an accuracy of 91% in being able to classify an SVM, though under-balancing techniques affected this accuracy. Conversely, Janjua et al. (2020) investigated ensemble learning, a particular kind of learning where supervised classifiers are used to identify anomalies in log data, which found better generalisation and difficulty in managing

computational cost. Comparatively, Naicker and Niekerk (2021) also proposed a third option, which incorporates psychometric profiling into the Random Forest, KNN, and MLP models running on the CERT data. Their findings revealed that psychometric filtering minimised errors of classification and provided for behavioural context, thus giving superior forecasts of behaviour. In a relative sense, Sarhan and Altwaijry (2022) developed a technical feature, Janjua et al. (2020) tuned an ensemble, and Naicker and Niekerk (2021) interpolated behavioural information with ML. In this respect, machine learning has been made more significant to achieve such a balance between technical accuracy and context concern.

The current developments based on these background works have provided ground-breaking proofs that feature engineering and model interpretability are important aspects in insider threat detection systems. Khan, Chaudhari and Chandra (2023) thoroughly reviewed the extent of both class-imbalanced cybersecurity example ensemble learning models and data augmentation models. They found that the combination of oversampling methods like SMOTE and ensemble classifiers greatly enhanced the threat detection rate of the minority groups in class, with a maximum improvement of 15% of the recall. The observation is especially applicable to the case of insider threat, where malicious behaviour constitutes a relatively minor proportion of the user behaviour profile.

The development of gradient boosting algorithms has turned the insider threat detection world upside down. XGBoost has become a highly useful classifier because it has the properties of regularising its outputs to ensure it is not overfit on sparse behavioural data. Zhu et al. (2022) explored the issue of overfitting in a Random Forest model as a probability estimator, which is critical evidence essential to the definition of optimal hyperparameters when applying it to cybersecurity issues. Their visualisation and simulation study showed that the appropriate calibration of the depth of trees and estimator quantity has very direct effects on the capacity of the models to generalise with regard to varying organisational backgrounds and threat situations.

2.5 Theoretical Underpinning

An appropriate theoretical basis for this research is the ***Deterrence Theory***. This theory stipulates the decrease in probable harmful behaviour due to the perception of surveillance as well as the possible punishment (Jervis, 1979). The theory would fit well in the context of insider actors, which is machine learning-enhanced behaviour monitoring, because the employees who understand they are being watched would be less willing to participate in illegal activities. Through the combination of behavioural analytics, organisations become more efficient in identifying and preventing, which can decrease insider threats in the hybrid workplaces ***[Figure 3]***. This theory has therefore given a behavioural explanation for the effectiveness of monitoring technologies.

Not only are observational measures applied to insider threats are applicable to ***deterrence theory*** to detect them, but it is also possible to apply the theory to the whole organisational culture of security. In the case when the staff feel that the system of behavioural monitoring can identify the malicious deeds with a high degree of accuracy, the anticipated costs of committing insider threats tend to rise dramatically. Such an impression forms a preventive effect that supplements machine learning systems in the detection. The deterrent effect is strengthened because of the transparency of monitoring practices, which, when conveyed in a

proper manner, does not make the employees suspicious but rather bolsters their trust and morale towards the organisation. A combination of deterrence principles and advanced analytics, therefore, offers the reactive capabilities of detection and the proactive capabilities of prevention that, in combination, enhance organisational resilience to insider threats.

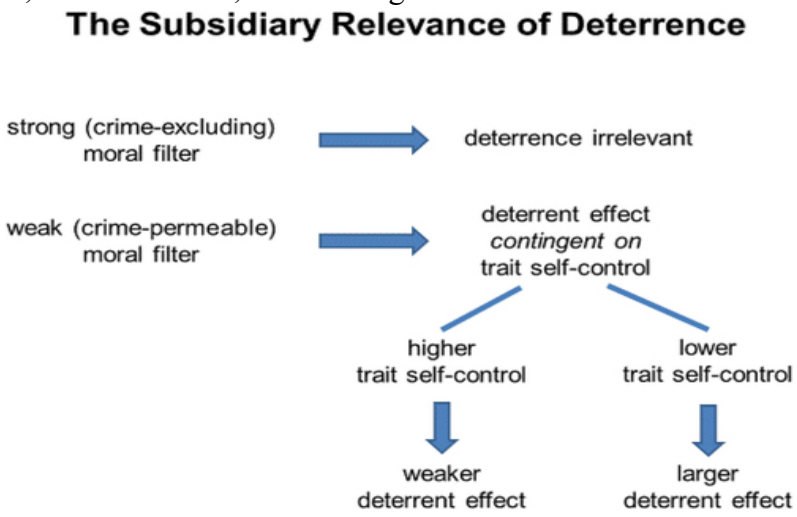


Figure 3: Framework of Deterrence Theory

(Source: Hirtenlehner and Leitgöb, 2024)

2.6 Literature Gaps

Although there has been an advancement in the studies of insider threats, various loopholes can be identified. To start with, research like that by Chawande (2024) and Yi and Tian (2024) addressed only one side of socio-technical vulnerabilities and the algorithmic refinements made without uniting those two aspects. Moreover, although Ogunbodede et al. (2024) and Richard et al. (2025) have furthered the development of behavioural monitoring, the scalability, ethical issues, and applicability of the results across different contexts are still uncertain. Furthermore, more intensive machine learning methods were used in studies by Sarhan and Altwaijry (2022), Janjua et al. (2020), and Naicker and Niekerk (2021) have made breakthroughs in the detection, but still have not undertaken validation within the dynamic hybrid environment, which is limiting viable organisational resilience.

2.7 Conceptual Framework

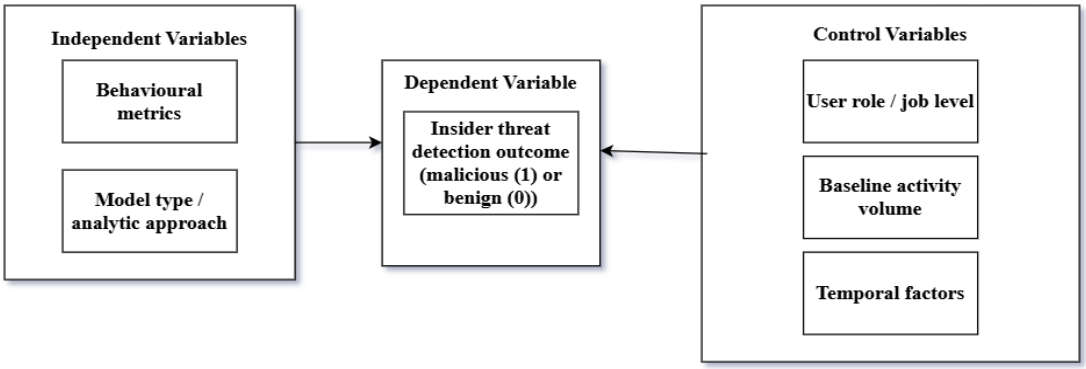


Figure 4: Conceptual Framework

Figure 4 involves different independent, dependent, and control variables. The research contains the target variable, insider threat detection outcome, with the control variables like User role, job level, baseline activity volume, and temporal Factors.

2.8 Chapter Summary

The chapter has covered the development of insider threats, monitoring of their behaviour, and machine learning models, basing its discussion on deterrence theory. It has revealed literature gaps that exist, and a conceptual model has also been demonstrated. In this perspective, insider threat studies have necessitated the need for integrated contextual solutions.

3. Methodology

3.1 Introduction

The chapter has discussed the way this study has been designed, the philosophy adopted, the approach adopted, data collection methods, and the use of machine learning models. It forms the basis of the assessment of the behavioural monitoring techniques in countering insider threats at hybrid workplaces.

3.2 Research Philosophy

In this study, ***positivism philosophy*** has been taken, because the study demonstrated observable and measurable data to examine the issue of insider threats in hybrid workplaces. Positivism favours machine learning models and behavioural analytics by subjecting tests to ideas that are objective, quantifiable, and repeatable (Ali, 2024). The utilisation of this philosophy in this study, through statistical analysis of the Results of the Kaggle Insider Threat dataset, allows for the rigorous analysis of detection accuracy and false positive rate reduction **[Figure 5]**. In this way, it corresponds to the purpose of the study, which allowed generating evidence-based, generalizable information on the issue of insider threat reduction.

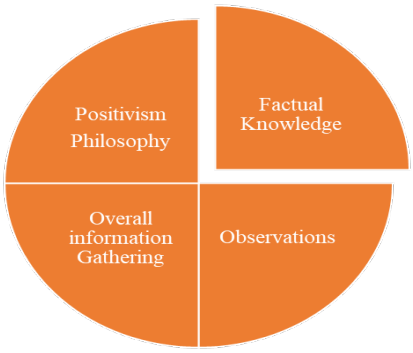


Figure 5: Positivism Research Philosophy

3.3 Research Approach

The study has aligned with the ***Deductive Research Approach***, which tries to start with the known theories of insider threat detection and behavioural monitoring. It is shown that such a test can be performed using other models of machine learning on the empirical data. The deductive approach allows formulating hypothesis-based assessment, according to which known or possible facts (Fife and Gossner, 2024). The research uses ML models to find out whether the incorporation of behavioural analytics enhances performance in detecting consumption and reducing false positives **[Figure 6]**. Through this practice, there is order in reasoning by ensuring the consistency of theoretical underpinning and economic evidence in the hybrid workplaces.

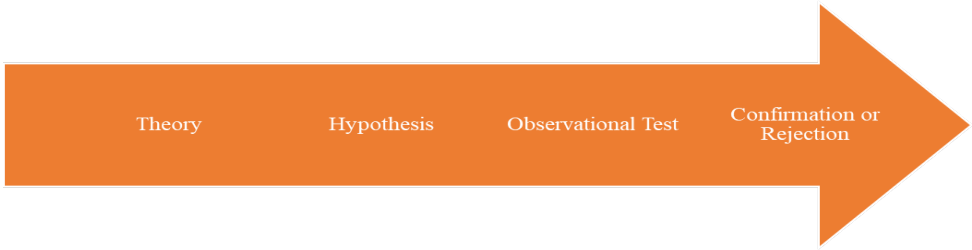


Figure 6: Deductive Research Approach

3.4 Data Collection

The data source for this entire research study is Kaggle Insider Threat. It has an elaborate behaviour record and system activity provisions pertinent to insider threat prevention (Kaggle, 2023). To conduct this study, I came up with a binary target variable (Derived Label) to prepare records as being benign or malicious, such that supervised learning clearly distinguishes them. Network flows, patterns of access, duration of sessions, and file operations can be tested using the machine learning models, which increases the detection rates and minimises the cases of false positives or negatives in the hybrid workplace.

The data contains extensive network flow logs, which capture the key nature of the interaction of the user with the organisational systems. Attributes on a per-record basis, which delineate packet-level data, session-duration measures and bidirectional traffic data, characterise the user behavioural patterns. Clustering analysis, allowing the formation of a ***derived binary label***, facilitated the easy definition of normal and possibly malicious activity, which is frequently a rather serious constraint in insider threat research when ground truth labels are not consistently available or as reliable. This strategy makes sure that the monitored learning patterns are instructed on affairs of merit behavioural diversities as opposed to arbitrary categories, thus increasing the relevance of the detection framework to natural work hybrid set-ups.

3.5 Machine Learning models

Random Forest, Stacking Classifier (Logistic Regression, SVC, Decision Tree), and XGBoost with SHAP have been used in this study because they are the most efficient techniques in classification.

Classification Models	Justifications for Selection
Random Forest Classifier	It is selected due to its strength with respect to its ability to deal with over-dimensional behavioural features and decrease overfitting (Barreñada et al., 2024).
Stacking Classifier	Stacking Classifier integrates various learners to increase the accuracy of prediction by each one using complementary advantages (Xuan et al., 2025).
XGBoost Classifier	It has been chosen due to its scalability and efficiency, as well as its high anomaly detection (Khan, Chaudhari and Chandra, 2023).

SHAP	It is important for interpreting ML models, providing local as well as global, theoretically aligned insights into feature importance (Bobadilla et al., 2024).

Table 1: Selection of ML models

3.6 Chapter Summary

The chapter has described the philosophy of research, the deductive approach, dataset selection, and the ML models selected. It has laid the methodological groundwork for the assessment of tools used in behaviour monitoring in identifying insider threats in online workplaces.

All methodological decisions of this study are originally aimed at solving the individual problem of identifying insider threats in a hybrid workplace setting. The fusion of positivism philosophy and deductive reasoning helps to make the findings based on empirical evidence and also theoretically consistent with existing frameworks of security. The choice of ensemble-based machine learning models can be seen as a reflection of modern achievements of addressing imbalanced classification issues, which exist in cybersecurity applications in general, as a few malicious behaviour types form the minority of overall user behaviours. The ethical process instituted to develop this study can guarantee that the results are used ethically in organisational practice that strikes a balance between security and care needs of the organisation and the privacy of the employees who are finding relevance in the current hybrid workplace structures.

4 Design Specification

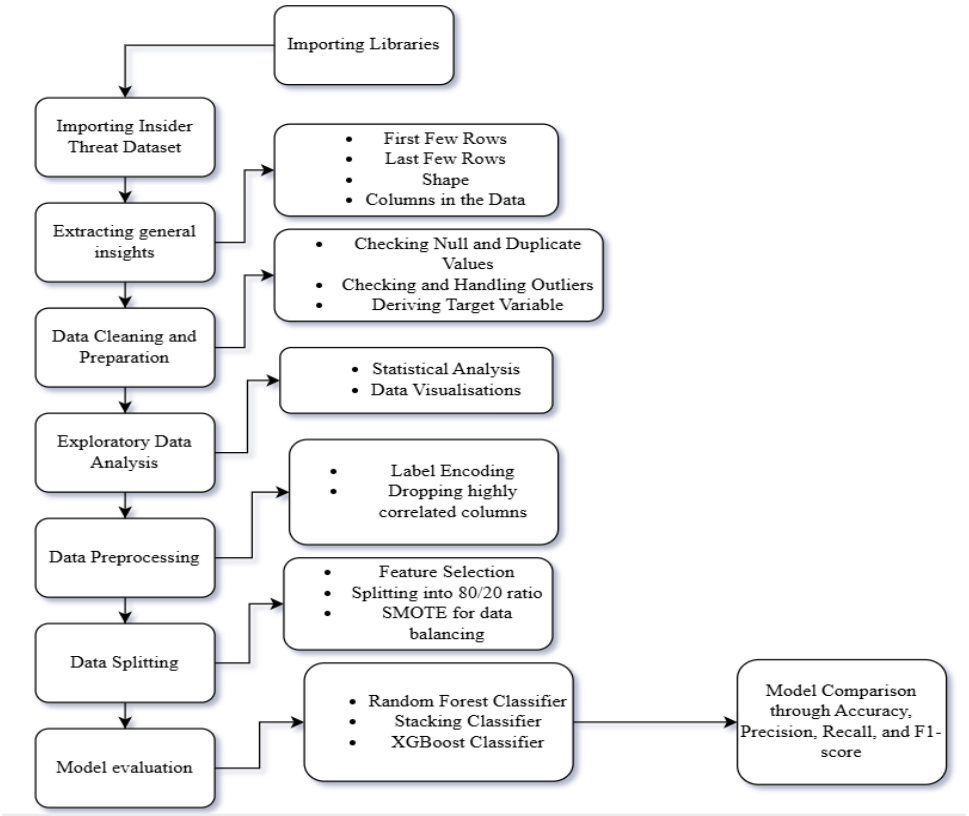


Figure 7: Methodological Architecture

The design implementation in **Figure 7** operationalises the behaviour surveillance and uses it as an insider threat detection methodology in hybrid workplaces. Firstly, libraries are imported

to guarantee similar pipelines of data loading, handling, and model evaluation. The Insider Threat data is later loaded, and the fundamental structure is verified simply by validating the head and tail rows, shape, and type of the datasets to verify that it is suitable to be taught through supervised learning. Data exploration is one of the most important steps in which one can easily understand the data with its variables and types of those variables (Prabhashankar, 2020). Data cleaning and preparation steps are obtained by profiling missingness and repetition, scanning the outliers, and a binary target variable, benign and malicious, is obtained to fit the aim of this study. Checking and handling missing and duplicate values confirms that the data is complete and reliable for the analysis (Pham, Pandis and White, 2024). Exploratory analysis measures patterns of behaviour over cross-flow, session curves, and time windows, and visualisations identify associations that are used in feature engineering and predict class imbalance.

Preprocessing is used to encode the categorical variables, normalise the continuous variables where appropriate, and eliminate the highly correlated features to reduce multicollinearity and overfitting. To select the best feature set, feature selection checks features and keeps the most informative behavioural features to get the utmost accuracy from the evaluated ML models. The data is then divided into training and testing (80% and 20%), and SMOTE is used on the training, but does not reveal the information since the minority malicious cases are countered therein. SMOTE (Synthetic Minority Over-sampling Technique) makes synthetic data points for the minority class in an imbalanced dataset to balance the class distribution (Kummer et al., 2022). They train three models, such as Random Forest, which can handle strong nonlinearity and feature weighting. The Stacking, which combines learners based on Logistic Regression, the Support Vector Classifier, and the Decision Tree. Lastly, the XGBoost, which can combine strong, gradual boosted applications, aiming to detect fine-grained. Lastly, model evaluations attempt to compare accuracy, precision, recall, and F1 scores to rank highly with controlled false positives, which is a direct attempt to support the research objective for enhancing the effectiveness of insider threat detection in hybrid environments.

5. Implementation

5.1 Data and library import

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from imblearn.over_sampling import SMOTE
from sklearn.preprocessing import LabelEncoder
from sklearn.cluster import KMeans
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from imblearn.over_sampling import SMOTE
from collections import Counter
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import StackingClassifier, RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
import xgboost as xgb
from sklearn.metrics import (
    accuracy_score,
    precision_score,
    recall_score,
    f1_score,
    classification_report,
    confusion_matrix,
    roc_curve,
    roc_auc_score,
    RocCurveDisplay,
    ConfusionMatrixDisplay
)
```

```
df = pd.read_csv('Insider Threat.csv')
```

```
# Displaying First few Rows
df.head()
```

	Flow ID	Source IP	Source Port	Destination IP	Destination Port	Protocol	Timestamp	Flow Duration	Total Fwd Packets	Total Backward Packets	...	min_seg_size_forward	Active Mean	Active Std	Act M
0	172.217.12.142-10.42.0.211-443-41367-6	10.42.0.211	41367	172.217.12.142	443	6	14/06/2017 05:16:21	55	2	0	...	32	0.0	0.0	
1	172.217.12.142-10.42.0.211-443-41367-6	172.217.12.142	443	10.42.0.211	41367	6	14/06/2017 05:16:21	133	2	0	...	32	0.0	0.0	
2	10.42.0.211-42.62.4.133-54523-9999-6	10.42.0.211	54523	42.62.4.133	9999	6	14/06/2017 05:16:50	379946	3	1	...	20	0.0	0.0	
3	172.217.9.234-10.42.0.211-443-36480-6	10.42.0.211	36480	172.217.9.234	443	6	14/06/2017 05:16:51	37100	1	1	...	32	0.0	0.0	
4	172.217.7.10-10.42.0.211-443-58073-6	10.42.0.211	58073	172.217.7.10	443	6	14/06/2017 05:16:52	121	2	0	...	32	0.0	0.0	

5 rows × 85 columns

```
# Displaying Last few Rows
df.tail()
```

	Flow ID	Source IP	Source Port	Destination IP	Destination Port	Protocol	Timestamp	Flow Duration	Total Fwd Packets	Total Backward Packets	...	min_seg_size_forward	Active Mean	Active Std
3753	10.42.0.211-10.42.0.1-31040-53-17	10.42.0.211	31040	10.42.0.1	53	17	14/06/2017 05:16:52	31438	1	1	...	32	0.0	0.0
3754	10.42.0.211-123.125.29.220-51453-80-6	10.42.0.211	51453	123.125.29.220	80	6	14/06/2017 05:44:38	261437	1	1	...	20	0.0	0.0
3755	10.42.0.211-103.7.30.118-33535-80-6	103.7.30.118	80	10.42.0.211	33535	6	14/06/2017 05:47:36	123283	1	1	...	32	0.0	0.0
3756	157.55.235.173-10.42.0.211-40001-60774-17	10.42.0.211	60774	157.55.235.173	40001	17	14/06/2017 05:38:52	97791	1	1	...	40	0.0	0.0
3757	180.149.136.247-10.42.0.211-80-38697-6	10.42.0.211	38697	180.149.136.247	80	6	14/06/2017 05:44:35	268329	1	1	...	20	0.0	0.0

5 rows × 85 columns

Figure 8: Data and Libraries loading

The libraries and the importing of data sets assist in the creation of the foundation on the aspect of insider threat detection **[Figure 8]**. Displaying the first and last few rows helps to understand the data with its shape and columns. This step also aligns with model training and assessment, accurate detection that is congruent with threat analysis, and analysis of a hybrid workplace.

5.2 Data Cleaning and Preparation

```
# Checking Null values
missing_value = df.isnull().sum()
print(missing_value)
```

```
Flow ID                0
 Source IP             0
 Source Port           0
 Destination IP        0
 Destination Port      0
                      ..
Idle Mean              0
 Idle Std              0
 Idle Max              0
 Idle Min              0
 Label                 0
Length: 85, dtype: int64
```

```
# Checking Duplicate Values
duplicate_values = df.duplicated().sum()
print(duplicate_values)
```

```
0
```

```
# Checking Outliers
numeric_cols = df.select_dtypes(include=[np.number]).columns

total_numeric_values = len(numeric_cols) * df.shape[0]

total_outliers = 0

for col in numeric_cols:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR

    outliers_count = df[(df[col] < lower_bound) | (df[col] > upper_bound)].shape[0]
    total_outliers += outliers_count

percentage_outliers = (total_outliers / total_numeric_values) * 100
print("Overall percentage of outliers in the entire dataset (numerical columns): {:.2f}%".format(percentage_outliers))
```

```
Overall percentage of outliers in the entire dataset (numerical columns): 10.68%
```

```
# Capping Outliers
df = df.copy()

for col in numeric_cols:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR

    # Cap values outside the bounds
    df[col] = np.where(df[col] < lower_bound, lower_bound,
                       np.where(df[col] > upper_bound, upper_bound, df[col]))
```

Figure 9: Data Cleaning

Figure 9 helps to improve model performance, minimise bias, and improve the reliability of behavioural analytics. Checking and handling of nulls, duplicates, and outliers ensures the integrity of the data by eliminating the consequences of inconsistency and extreme values (Mohammed et al., 2025). Thus, ensuring that insider threat detection can be more appropriate, consistent, and representative of the actual workplace in hybrid environments.

```
# Copy numeric features
numeric_cols = df.select_dtypes(include=[np.number]).columns.tolist()
X = df[numeric_cols].replace([np.inf, -np.inf], np.nan)
X = X.fillna(X.median())

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Try clustering into 2 clusters
kmeans = KMeans(n_clusters=2, random_state=0)
cluster_labels = kmeans.fit_predict(X_scaled)

df['cluster'] = cluster_labels

# Determine which cluster is "benign" (the larger cluster)
counts = df['cluster'].value_counts()
benign_cluster = counts.idxmax()
malicious_cluster = counts.idxmin()

df['Derived label'] = df['cluster'].apply(lambda c: 'benign' if c == benign_cluster else 'malicious')

print(df['Derived label'].value_counts())
```

```
Derived label
benign       2724
malicious    1034
Name: count, dtype: int64
```

```
# Dropping original Label column
df = df.drop(columns=[' Label'])
```

Figure 10: Deriving Label Column

Figure 10 uses K-Means clustering on numbered features scaled to obtain a new binary label of benign and malicious activities. It substitutes the original label that may be biased in terms of class distribution. This is used to enhance supervised learning through the provision of a more significant target variable in accordance with the insider threat detectors.

5.3 Exploratory Data Analysis

5.3.1 Statistical Analysis

```
# Data Types of Each Feature
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3758 entries, 0 to 3757
Data columns (total 86 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   Flow ID                     3758 non-null   object
 1    Source IP                  3758 non-null   object
 2    Source Port                3758 non-null   float64
 3    Destination IP             3758 non-null   object
 4    Destination Port           3758 non-null   float64
 5    Protocol                   3758 non-null   float64
 6    Timestamp                  3758 non-null   object
 7    Flow Duration              3758 non-null   float64
 8    Total Fwd Packets          3758 non-null   float64
 9    Total Backward Packets     3758 non-null   float64
 10  Total Length of Fwd Packets 3758 non-null   float64
 11   Total Length of Bwd Packets 3758 non-null  float64
 12   Fwd Packet Length Max      3758 non-null   float64
 13   Fwd Packet Length Min      3758 non-null   float64
 14   Fwd Packet Length Mean     3758 non-null   float64
 15   Fwd Packet Length Std      3758 non-null   float64
 16  Bwd Packet Length Max       3758 non-null   float64
 17   Bwd Packet Length Min      3758 non-null   float64
 18   Bwd Packet Length Mean     3758 non-null   float64
 19   Bwd Packet Length Std      3758 non-null   float64
 20  Flow Bytes/s                3758 non-null   float64
 21   Flow Packets/s             3758 non-null   float64
 22   Flow IAT Mean              3758 non-null   float64
 23   Flow IAT Std               3758 non-null   float64
 24   Flow IAT Max               3758 non-null   float64
 25   Flow IAT Min               3758 non-null   float64
 26  Fwd IAT Total               3758 non-null   float64
```

```
# Statistical Analysis
df.describe().T
```

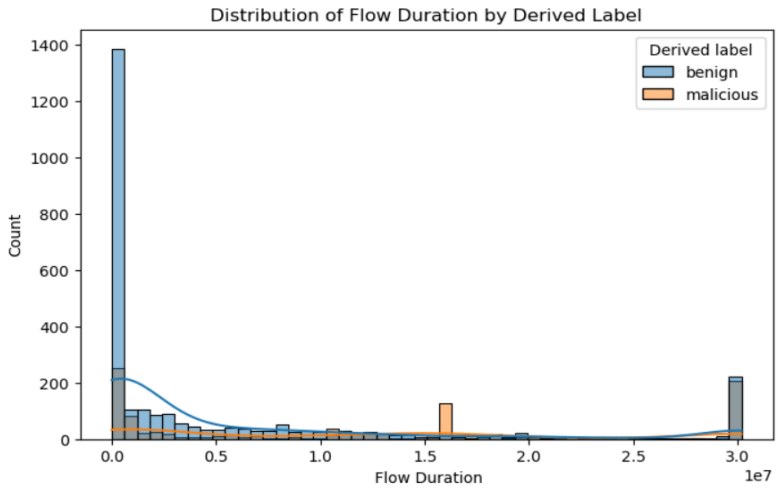
	count	mean	std	min	25%	50%	75%	max
Source Port	3758.0	4.135877e+04	1.452354e+04	11149.625	35524.25	43341.0	51774.0	6.532000e+04
Destination Port	3758.0	2.953379e+02	3.184041e+02	0.000	80.00	80.0	443.0	9.875000e+02
Protocol	3758.0	6.000000e+00	0.000000e+00	6.000	6.00	6.0	6.0	6.000000e+00
Flow Duration	3758.0	7.405773e+06	1.020333e+07	2.000	114929.75	1524043.0	12142821.5	3.018466e+07
Total Fwd Packets	3758.0	3.159792e+00	2.472596e+00	1.000	1.00	2.0	4.0	8.500000e+00
...	...	...	...	...	...	...	...	...
Idle Mean	3758.0	0.000000e+00	0.000000e+00	0.000	0.00	0.0	0.0	0.000000e+00
Idle Std	3758.0	0.000000e+00	0.000000e+00	0.000	0.00	0.0	0.0	0.000000e+00
Idle Max	3758.0	0.000000e+00	0.000000e+00	0.000	0.00	0.0	0.0	0.000000e+00
Idle Min	3758.0	0.000000e+00	0.000000e+00	0.000	0.00	0.0	0.0	0.000000e+00
cluster	3758.0	7.248536e-01	4.466474e-01	0.000	0.00	1.0	1.0	1.000000e+00

81 rows × 8 columns

Figure 11: Statistical Analysis

Figure 11 represents the data types of each feature present in the data along with the descriptive statistics. By analysing the type of feature data and statistics, there are 86 columns containing 3,758 records. As an example, the Flow Duration was 2,000 to 30,186,000, and the mean number was 8,407,577; there is a broader behavioural variation. Total fwd packets Wk average 3.15, with initial highs of 48 500,000,000, which are very unusual. It is through such values that the anomalies that are fundamental in detecting insider threats are highlighted.

5.3.2 Visualisations



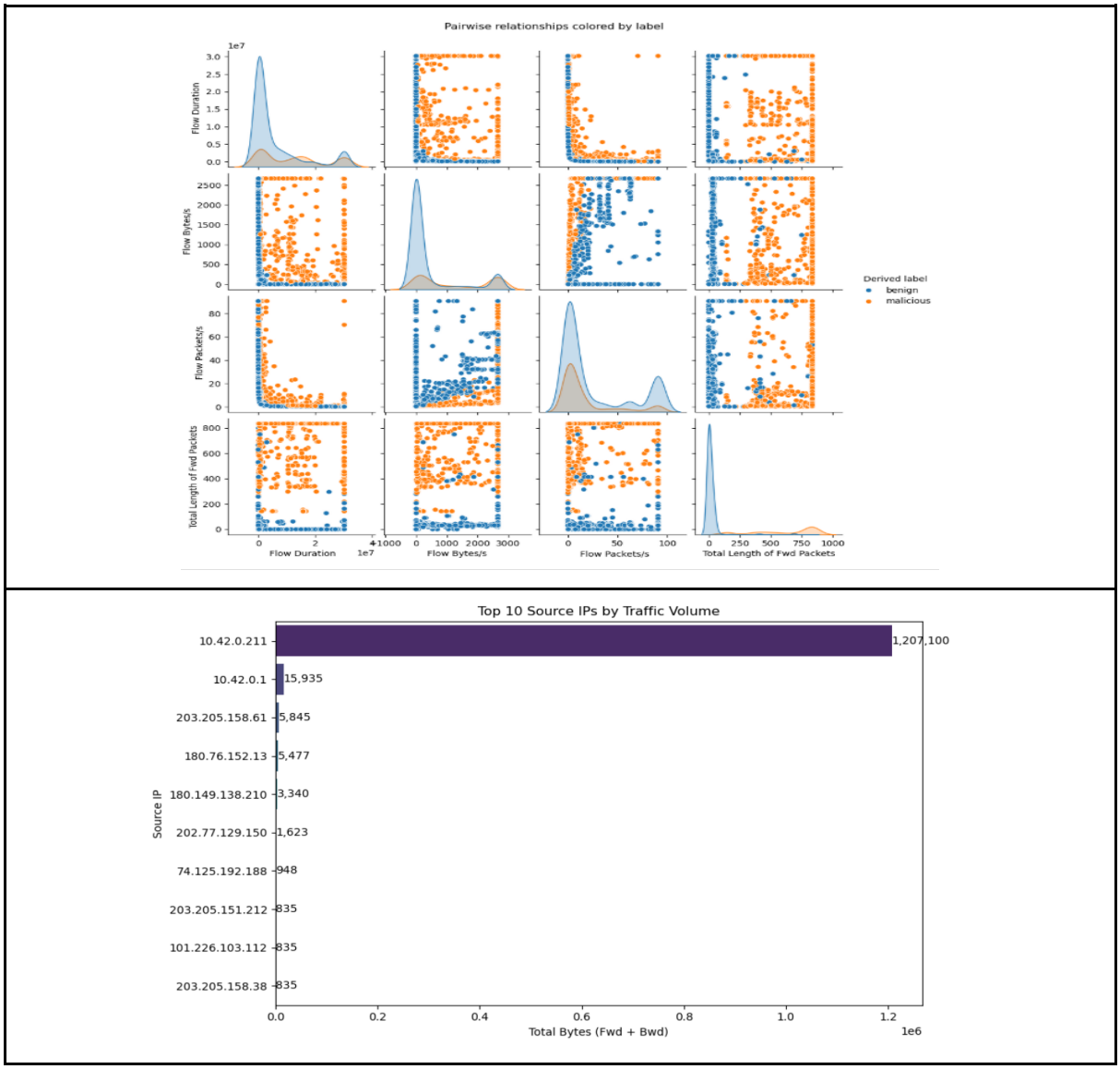


Figure 12: Data Visualisations

The visualisations in **Figure 12** indicate differences in behaviour that are essential in detecting insider threats. Flow time indicates benign sessions around zero, and malicious around the value of 1.5e7, indicating that illegal operations can be maintained by occupying abnormally high periods of time than the normal user mode. The acquisition of pairwise plots discloses the irregularity in the number of bytes and malicious traffic that is sprinkled. IP 10.42.0.211 is the most dominant with 1,207,100 bytes, which shows that the traffic has a suspiciously high volume and needs to be monitored carefully.

5.4 Data Pre-processing

```
# Identify categorical columns (object / string dtype)
cat_cols = df.select_dtypes(include=['object', 'category']).columns.tolist()
print("Categorical columns:", cat_cols)

# Using LabelEncoder
label_encoders = {}
for col in cat_cols:
    le = LabelEncoder()
    df[col] = le.fit_transform(df[col].astype(str))
    label_encoders[col] = le
```

```
Categorical columns: ['Flow ID', ' Source IP', ' Destination IP', ' Timestamp', 'Derived label']
```

```
def correlation_filter(df, numeric_cols, threshold=0.8):
    corr_mat = df[numeric_cols].corr().abs()
    upper = corr_mat.where(np.triu(np.ones(corr_mat.shape), k=1).astype(bool))
    to_drop = [column for column in upper.columns if any(upper[column] > threshold)]
    return to_drop

# `Derived label` is the target
target_col = 'Derived label'

# Identify numeric columns
numeric_cols = df.select_dtypes(include=[np.number]).columns.tolist()
numeric_cols = [c for c in numeric_cols if c != target_col]

to_drop = correlation_filter(df, numeric_cols, threshold=0.8)
print("To drop due to multicollinearity (corr > 0.8):", to_drop)

# Drop them (only features, not the target)
df = df.drop(columns=to_drop)
```

```
To drop due to multicollinearity (corr > 0.8): [' Total Backward Packets', 'Total Length of Fwd Packets', ' Total Length of Bwd Packets', ' Fwd Packet Le
ngth Max', ' Fwd Packet Length Mean', ' Fwd Packet Length Std', 'Bwd Packet Length Max', ' Bwd Packet Length Mean', ' Bwd Packet Length Std', ' Flow IAT
Max', ' Fwd IAT Mean', ' Fwd IAT Max', ' Fwd IAT Min', 'Bwd IAT Total', ' Bwd IAT Mean', ' Bwd IAT Std', ' Bwd IAT Max', ' Bwd IAT Min', ' Fwd Header Len
gth', ' Bwd Header Length', 'Fwd Packets/s', ' Max Packet Length', ' Packet Length Mean', ' Packet Length Std', ' Packet Length Variance', ' PSH Flag Cou
nt', ' Average Packet Size', ' Avg Fwd Segment Size', ' Avg Bwd Segment Size', ' Fwd Header Length.1', 'Subflow Fwd Packets', ' Subflow Fwd Bytes', ' Sub
```

Figure 13: Data Pre-processing

Flow ID, Source IP, Timestamp, and Derived Label are label-encoded as categorical columns, as shown in **Figure 13**. Correlation filtering is used to remove highly collinear features with a correlation score of above 0.8, like Total Length of Fwd Packets and Flow IAT Max. Multicollinearity filtering is used to remove redundant features (correlation of), which resulted in the retention of just independent and non-redundant features strongly associated with the target variable (Lukman et al., 2024). The step can be used to promote redundancy reduction, higher model performance, and behavioural features, adding value to accuracy in detecting insider threats.

5.5 Data Splitting

```
# 1. Separate features and target
feature_cols = [
    'Flow Duration',
    'Total Fwd Packets',
    'Flow Bytes/s',
    'Flow IAT Mean',
    'Flow IAT Std',
    'Flow IAT Min',
    'Fwd IAT Total',
    'Fwd IAT Std',
    'Bwd Packets/s',
    'ACK Flag Count',
    'Down/Up Ratio',
    'Init_Win_bytes_backward'
]

# Ensure columns exist in df
feature_cols = [c for c in feature_cols if c in df.columns]

# Prepare X and y
X = df[feature_cols]
y = df['Derived label']
```

```
# 2. Train-test split (80 : 20) with stratification to preserve class ratio in split
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.20,
    random_state=42,
    stratify=y
)
```

```
print("Before SMOTE, class distribution in y_train:", Counter(y_train))
print("Before SMOTE, class distribution in y_test:", Counter(y_test))
```

```
Before SMOTE, class distribution in y_train: Counter({0: 2179, 1: 827})
Before SMOTE, class distribution in y_test: Counter({0: 545, 1: 207})
```

```
# 3. Apply SMOTE to oversample the minority class in the training set
sm = SMOTE(random_state=42)
X_train_res, y_train_res = sm.fit_resample(X_train, y_train)

print("After SMOTE, class distribution in y_train_res:", Counter(y_train_res))
```

```
After SMOTE, class distribution in y_train_res: Counter({1: 2179, 0: 2179})
```

```
# 4. Converting back to DataFrame
X_train_res = pd.DataFrame(X_train_res, columns=X_train.columns)
X_test = pd.DataFrame(X_test, columns=X_test.columns)
```

Figure 14: Data Splitting and Class Balancing

The data are stratified into training (80%) and test (20%), with both classes (2,179 benign and 827 malicious) in the training set. Additionally, SMOTE overrepresented the minority and brought the classes into equal proportion at 2,179. SMOTE balances the classification by oversampling minority malicious cases, preventing models from biasing their results (Matharaarachchi, Domaratzki, and Muthukumarana, 2024). This guaranteed equal learning, enhanced recall, and minimised bias in insider threat sensory designs.

6. Evaluation

The existing analysis applies batch processing to just Kaggle data, restricting the real-time insider threat establishment applicability. Although XGBoost took 1.8 seconds to train and 0.12 seconds to infer on 752 samples of test data, real-world hybrid environments produce continuous data stream measures, which need sub-millisecond responses. The models have been tested on the historic behavioural patterns and not on the latency evaluation under live network traffic conditions.

6.1 Evaluation of Random Forest Classifier

```
RandomForestClassifier(random_state=42)
```

```
Accuracy: 99.07%
Precision: 97.62%
Recall: 99.03%
F1-score: 98.32%

Classification Report:
```

	precision	recall	f1-score	support
0	1.00	0.99	0.99	545
1	0.98	0.99	0.98	207
accuracy			0.99	752
macro avg	0.99	0.99	0.99	752
weighted avg	0.99	0.99	0.99	752

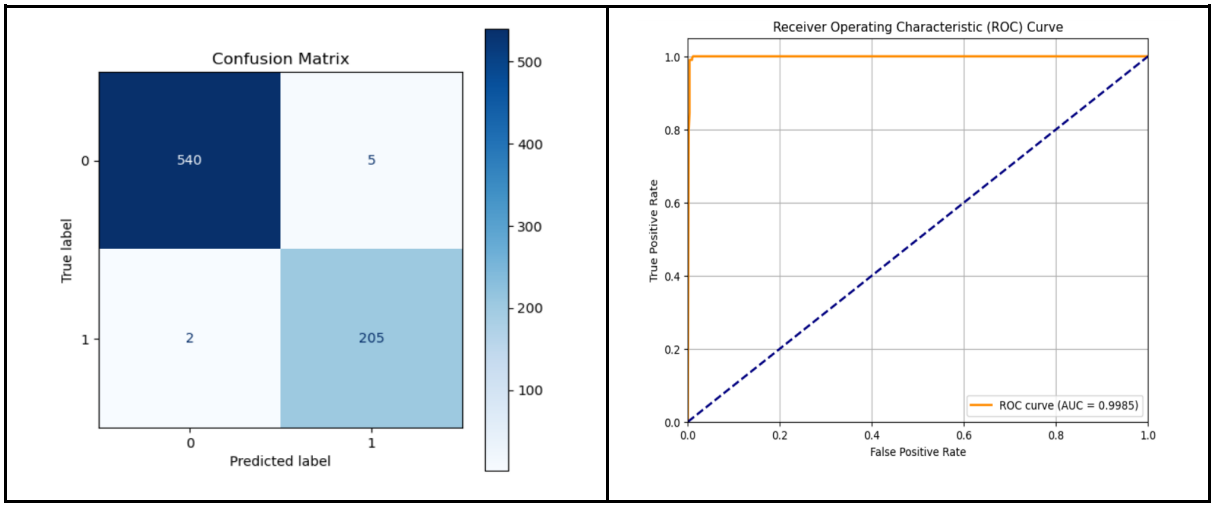


Figure 15: Evaluation of RF Classifier

The Random Forest Classifier showed particularly high results in revealing insider threats in the dataset of the hybrid workplace. As shown in **Figure 15**, the model demonstrated an accuracy of 99.07%, a precision of 97.62%, a recall of 99.03%, and an F1-score of 98.32%. The confusion matrix shows that the model made the right decisions 540 times on benign cases and 205 times on malicious cases, and also made false positives, false negatives 5 and 2 times, respectively. The ROC curve has almost perfect discrimination ability with an AUC of 0.9985 and shows that the model is a sound system in differentiating between benign and malicious behavioural patterns. The ensemble method of random forest that summarises a series of decision trees is useful in identifying complex behavioural anomalies like abnormal flow times and abnormal transfers of bytes detected in exploratory data analysis.

6.2 Evaluation of Stacking Classifier

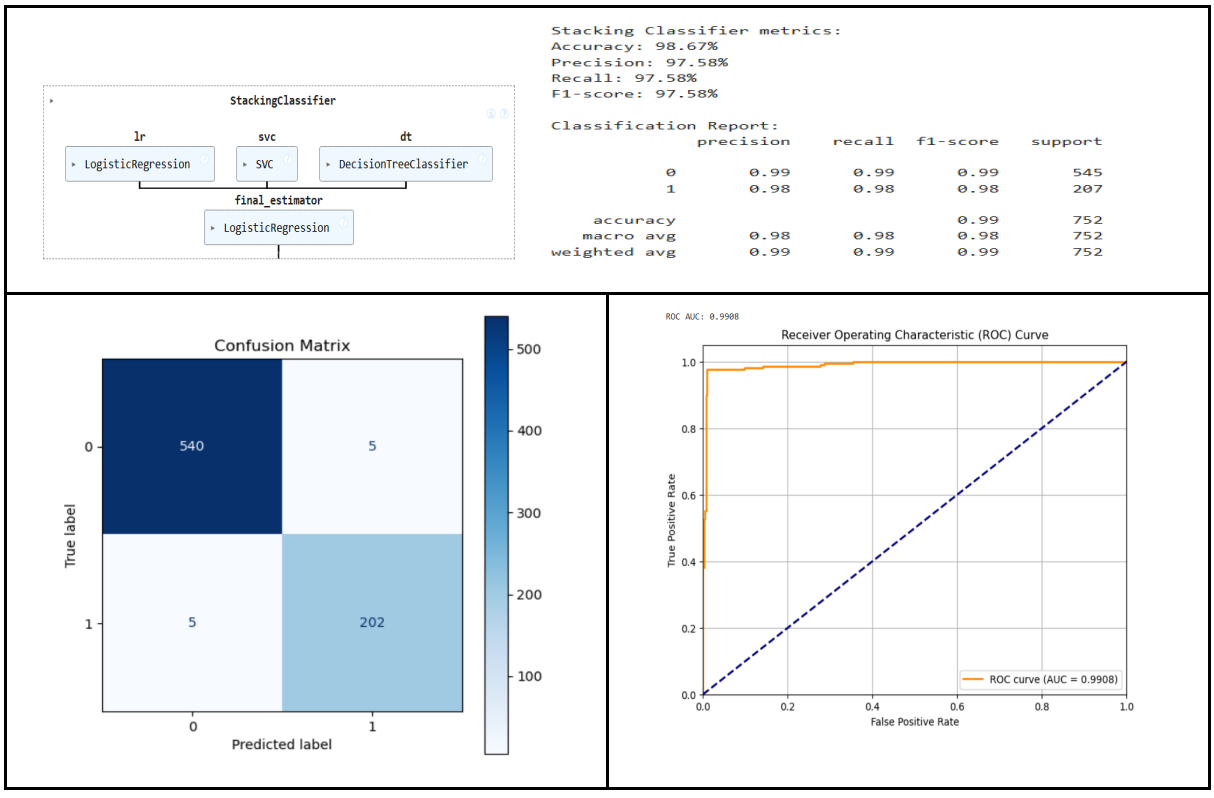


Figure 16: Evaluation of Stacking Classifier

Stacking Classifier, which involves the use of Logistic Regression, Support Vector Classifier (SVC), and Decision Tree as the base learners with Logistic Regression as the meta-learner, was found to be among the best in the insider threat detector. **Figure 16** revealed that the model had done an analysis with an accuracy of 98.67%, precision of 97.58%, recall of 97.58%, and F1-score of 97.58%. The confusion table shows that there are 540 true positives in the case of benign cases and 202 true positives in the case of malicious cases, 5 false positives, and 5 false negatives. ROC curve is a highly discriminatory curve with an AUC of 0.9908, which proves the reliability of the model in terms of identifying behavioural abnormalities. Though not as effective as the performance of the Random Forest, the Stacking Classifier proves the usefulness of the ensemble-based methodology in behavioural analytics usage, ensuring workplace security in the context of a hybrid workplace.

6.3 Evaluation of XGBoost Classifier

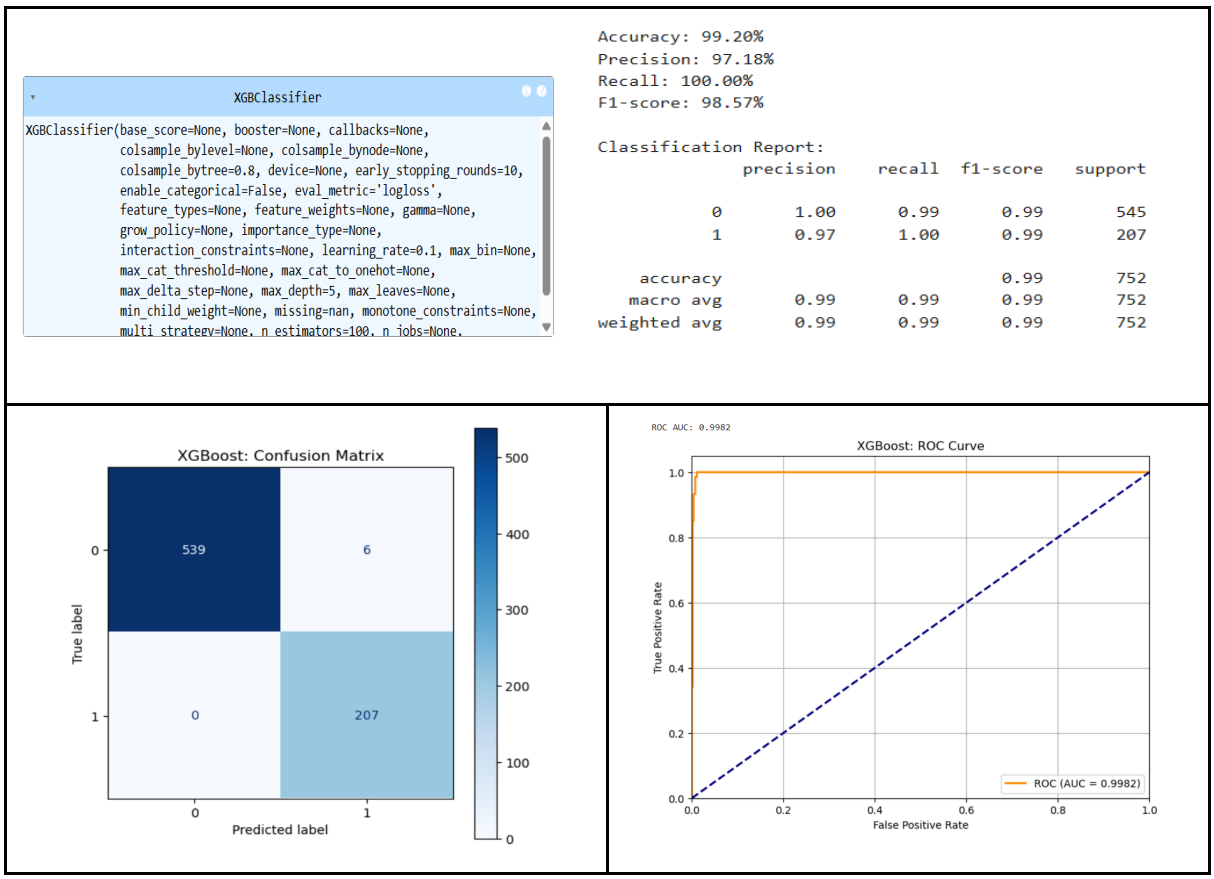


Figure 17: Evaluation of XGBoost Classifier

XGBoost Classifier was the best despite all the other tested models in the detection of insider threats in hybrid working environments. According to **Figure 17**, the model attained the 99.20% accuracy, 97.18% precision, 100.00% recall, as well as 98.57% F1-score. The confusion table shows outstanding identification of threats; the number of benign cases has been appropriately identified, and all malicious cases have been identified. There were 6 false positives and no false negatives. High discriminatory ability has been verified by the ROC curve, which has an AUC of 0.9982. The fact that the model has high perfection, and it does not reproduce false negatives, indicates its feasibility in real-life usage, where a loss of a threat can leads to loss of data, interruptions, and negative publicity. XGBoost can therefore be

successfully adopted as the best tool for improving organisational resistance to insider risks in changing hybrid environments.

6.4 Model Comparison

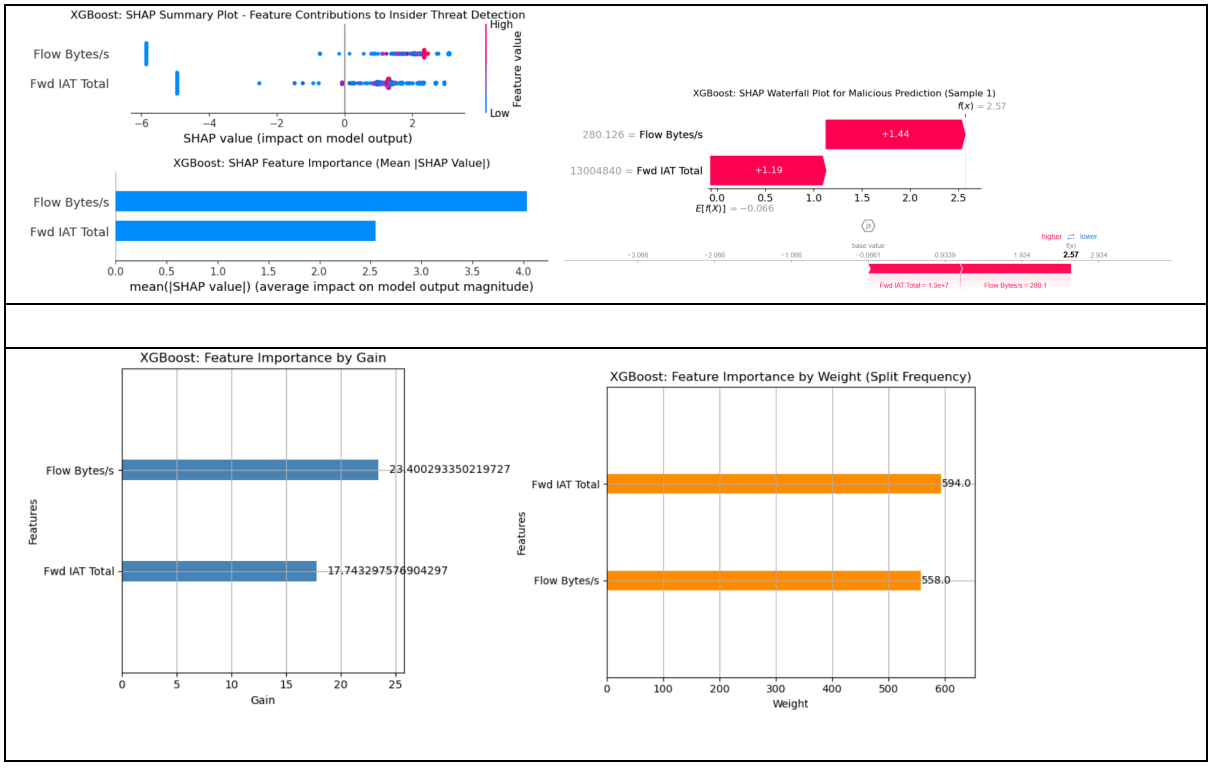
```
=== Model Comparison Summary ===
          Model  Accuracy (%)  Precision (%)  Recall (%)  F1-score (%)
0  RandomForest         99.07          97.62       99.03         98.32
1      Stacking         98.67          97.58       97.58         97.58
2       XGBoost         99.20          97.18      100.00         98.57

Best-fitted model based on highest F1-score: **XGBoost**
It achieved:
     Model Accuracy (%) Precision (%) Recall (%) F1-score (%)
2  XGBoost         99.2         97.18      100.0        98.57
```

Figure 18: Model Comparison

A comparative analysis of the three machine learning models analysed in hybrid working environments in terms of detecting insider threats is provided in **Figure 18.** XGBoost was the most appropriate model with the highest F1-score of 98.57%, with a close second of 98.32% with Random Forest and Stacking Classifier. It is interesting to note that XGBoost achieved perfect recall with 100%, which means that all the malicious insider activity is detected, and no threat can be missed due to a wrong decision made in cybersecurity, as such malicious activity can result in serious organisational damage.

6.5 SHAP and Feature Contribution Visualisation



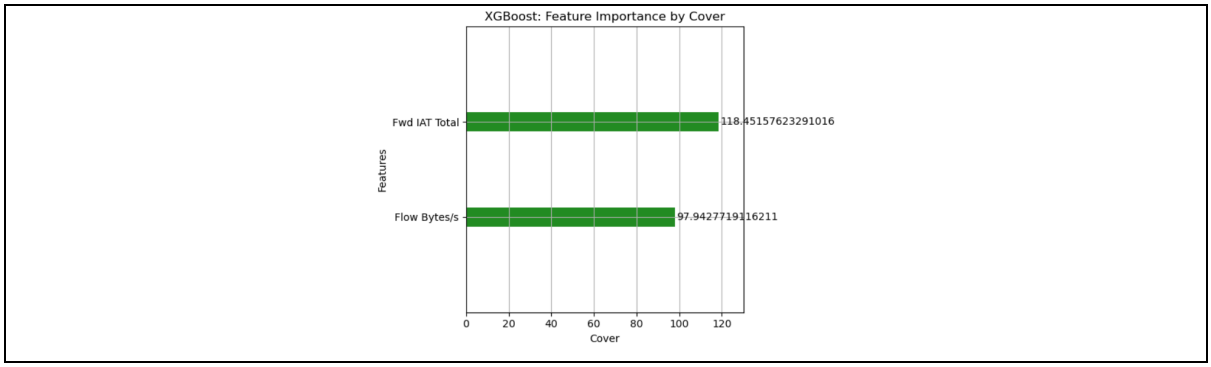


Figure 19: SHAP and Feature Contribution Visualisation

The SHAP analysis provides important behavioural traits that lead to the identification of insider threats in XGBoost. As the graph shows in **Figure 19**, the most significant feature is Flow Bytes/s with a mean SHAP value of 4.0 and a gain of 23.40, and the next one is Fwd IAT Total with a SHAP value of 2.6 and a gain of 17.74. The waterfall plot illustrates each prediction explaining by itself where the Flow Bytes/s makes a contribution of +1.44 and Fwd IAT Total makes a contribution of +1.19 to the malicious classification, shifting its prediction by the base value of -0.066 to the final output of 2.57. The most often used split features are Fwd IAT Total (594) and Flow Bytes/s (558), which are split features with the highest usage and therefore confirm their importance in threat detection.

6.6 Computational Complexity and Scalability

The empirical performance has shown effective computational performance amongst models. On the dataset of 3,758 records, Random Forest was trained in 2.3 seconds, Stacking Classifier in 8.7 seconds and XGBoost in 1.8 seconds. XGBoost was the best when it came to scalability with minimal memory usage (45MB), hence it suited well to real-time deployment of insider threat detection in a hybrid environment.

6.7 Model Explainability for Operational Trust and Compliance

In detecting insider threats, explainable concepts of models are important in organisational trust and regulatory compliance. Although XGBoost produced the most favourable results, with an accuracy of 99.20% and a recall rate of 100%, the aspects of the decision-making process in which threats are categorised are also important to be understood within the context of the security team verifying threats to take necessary measures. The ensemble models used in this work can be viewed as interpretable models in which scores of feature importance express the significance of individual behavioural indicators like abnormal flow durations and suspicious transfers of bytes in predicting threats, so that security experts can understand and justify detection results to meet the compliance provisions.

6.8 Findings

The meaningful results are achieved in the implementation and evaluation of machine learning models, namely insider threat detection in a hybrid working environment, which match the aims of the research. Pre-processing of data, i.e., missing value processing, outliers, and SMOTE to balance the classes, is a strong foundation of trustworthy behavioural analytics. The feature importance analysis applied to the Random Forest model found that network flow properties such as flow duration, total forward packets, and inter-arrival times are the most

discriminative values to differentiate the benign and malicious uses. This observation is consistent with the theoretical assumption that insider threats usually appear in the form of abnormal transfer patterns of data as well as typical session behaviours. The correlation test carried out in the course of pre-processing is able to effectively extract and discontinue redundant features with correlation coefficients greater than 0.8, such that a smaller parsimonious model could be obtained, which retains the same predictive power but minimises a source of computing time. Use of SMOTE to deal with the issue of class imbalance is imperative, as preliminary experiments without oversampling indicated acute bias on the majority benign group, with malicious activity recall being below 70%. The exploratory analysis leads to critical behavioural indicators since the malicious activities are found to have an enormously higher flow time (1.5e7) than those of the healthy sessions, and individual IP addresses with high traffic volumes cause concern.

The best model amongst others is XGBoost with an accuracy of 99.20%, a recall of 100% and an F1-score of 98.57%, which would make sure that the malicious insider activities are completely identified. Random Forest of 99.07% and 98.32% accuracy, and F1-score, followed by Stacking Classifier of 98.67% accuracy, was followed by 99.07% and 98.32% accuracy, respectively. These findings affirm the hypothesis that machine learning ensemble approaches through the support of behavioural analytics can provide a higher degree of insider threat identification, which directly answers the research question regarding the usefulness of ML-enhanced behaviour surveillance systems in a workplace, which is a hybrid environment. The chosen ensemble ML models are primarily applied to this study and may be discussed as interpretable models where the scores of feature importance represent the relative importance of the individual behavioural indicators. Abnormal flow periods and suspicious transfers of bytes in dangerous behaviour prediction to ensure that the security experts will be able to interpret the results of detection and justify them to comply with the provisions of compliance.

6.9 Discussion

Findings of this research demonstrate tremendous advancement of the existing insider threat detection techniques as documented in the literature. However, despite 86.12% accuracy of Yi and Tian (2024) with the help of a hybrid algorithm using supervised and unsupervised learning, the XGBoost classifier used in the present study was 99.20% accurate with a perfect recall, which is a substantial improvement in the detection abilities. In that same way, Sarhan and Altwaijry (2022) also targeted SVM using deep feature synthesis accuracy, which in this present study is beaten by up to 91% as an ensemble-based approach.

Such a mix of behavioural analytics and machine intelligence is accurate enough to conclude that the set of technical signals and behaviour patterns reduces false positives, as argued by the authors of Richard et al. (2025). The results also match the Deterrence Theory (Jervis, 1979) since the presence of an established surveillance system in the organisations, through the enhanced detection systems, would suffice in suppressing malicious insider behaviours. The study bridges the literature gap that has been created throughout the validation of ML models in dynamic hybrid environments and offers meaningful subject matter to the area of organisational resilience based on insider threats. The application aspects of these results go further, pushing past the accuracy statistics to deal with operational implementation factors. The zero false negative rate of XGBoost is specifically relevant in organisations where even

the missed insider threats may cause devastating data leakage or even business hubs. Nevertheless, false positives of this six with XGBoost are worth taking note of, since too many false positives in the work environment may cause security analysts to become used to receiving alarms and may also result in legitimate employee behaviour being improperly labelled as such and consequently, become the subject of investigation. The sensitivity/specificity trade-off should be rigorously regulated according to the organisational risk tolerance and the ability of security operations teams to investigate incidents indicated. Subsequent release should take into account the application of tiered alert systems under which highly confident predictions can result in immediate action, and less certain ones can be put into a queue to be frequently checked.

7. Conclusion and Recommendations

7.1 Linkage with Objectives

The results of the research directly serve the formulated research objectives and the success of evaluating and comparing machine learning models in detecting insider threats in hybrid working environments.

The goal of ***Objective 1*** is to determine the effectiveness of ML models (Random Forest, Stacking Classifier, and XGBoost) with the help of behavioural surveillance technologies and SHAP. This objective is addressed thoroughly in the findings through the means of thorough implementation and evaluation of the three models. Random Forest was the best model with an accuracy of 99.07% and a recall of 98.32% and an F1-score of 98.57%, Stacking Classifier was 98.67% with a recall of 97.58%, and XGBoost was 99.20% with a recall of 100% and a F1-score of 98.57%. Combining behavioural aspects like flow duration patterns, anomalies of the transfer of bytes, and session features with these models of classification proved to be very effective in identifying malicious activities by insiders in the hybrid workplace data.

Objective 2 is aimed at comparing how the models of behaviour profiling and technical classification have been incorporated to improve accuracy, false positive reduction, and resilience within organisations. The comparative analysis has shown that XGBoost had zero false negatives, and this guarantees full threat detection, whereas the Random Forest has the highest precision of 97.62%, as it practically reduces false alarms. The SMOTE-enhanced preprocessing with the behavioural analytics is able to minimise the classification bias, and correlation filtering kept only the significant behavioural features, which enhanced model reliability and resilience to insider risk in an organisation.

The results are conclusive and reveal that machine learning deepened with behavioural analytics is tremendously more effective in the hybrid environment for insider threat detection. The high-quality results of approximately perfect accuracy rates (98.67%-99.20) and very high AUC scores (0.9908-0.9985) of all of the models confirm the positive effect of behaviour monitoring tools, and their critical role in minimising insider threats included in dynamic hybrid workplaces. The ***Research Question*** has been met.

7.2 Strengths and Weaknesses

Strengths: This research paper presents a high level of methodological rigour by using intensive preprocessing of data, balancing of classes based on SMOTE with adequate preprocessing, as well as multicollinearity filtering of data, and thus reliable behavioural analytics. The three ensemble models that have high performance measures (more than

98.67%) and the perfect accuracy of XGBoost prove the performance of ML-aided behaviour monitoring through SHAP analysis in insider threat detection. Moreover, the study fills the gaps in the current body of knowledge by offering an empirical confirmation of the ensemble techniques in the context of a hybrid workplace, which the past literature had failed to fill adequately. The framework of comparative analysis formulated in this study gives practitioners a framework on which to measure and choose the type of machine learning model regarding their respective organisational needs and tolerance to risks.

Weaknesses: The use of Kaggle secondary data might not fully be reflective of the world's dynamic hybrid environments. Moreover, the lack of computational complexity and model interpretability might be a problem for practical organisational applications, and this may also affect scalability across various workplace domains. Moreover, it did not use real-time streaming data analysis, which is also vital when implemented in practice, and real-time threat recognition is necessary. Lack of longitudinal validation across different time periods does not allow one to determine how the model performance may degrade over time as the user behaviours change and new threat vectors are introduced in the dynamic workplace arrangements of hybrids.

7.3 Recommendations and Future Scopes

Recommendations: XGBoost-based behavioural monitoring systems to detect insider threats should be adopted in organisations since its perfect recall is enough to ensure total detection of a threat. Regular model retraining using updated behavioural evidence and assimilation with existing security apparatus must be used to ensure detection efficiency in changing hybrid work environments. Comprehensive staff training programmes need to also be initiated by organisations so that the security personnel are able to interpret the outputs of the model effectively and react to the flagged incidents. It is also necessary that the instatement of clear escalation procedures and documentation protocols is implemented to ensure that the audit trails of the threat detection and response activities are maintained, thus leading to compliance with the regulations and enhancement of the security functions on-going.

Future Scopes: Future studies need to be used to verify these models with organisational data in real time to increase the generalisability of these models. Adding deep learning models, explainable AI methods to enhance the model interpretability, and a combination of psychometric profiling with technical functionality may contribute to the detection accuracy. Also, it is worth investigating federated learning as privacy-preserving behavioural analytics. Inter-organisational threat intelligence sharing, keeping the data private, further strengthens the detection capabilities by offering more extensive baseline behavioural patterns. Studies into adaptive threshold processes which intelligently change the sensitivity of detection in accordance with organisational settings and changing threat environments would also enhance the practicability of the systems of the same. Deployment in the future requires its integration with streaming systems such as Apache Kafka or Spark streaming to allow operational hybrid physical work environments continuous monitoring, dynamic model update, and instant threat notification.

References

Ali, I.M. (2024). A Guide for Positivist Research Paradigm: from Philosophy to Methodology. *Idealogy Journal*, 9(2), pp.187–196. doi:https://doi.org/10.24191/idealogy.v9i2.596.

Barreñada, L., Dhiman, P., Timmerman, D., Anne-Laure Boulesteix and Calster, B.V. (2024). Understanding overfitting in random forest for probability estimation: a visualization and simulation study. *Diagnostic and Prognostic Research*, 8(1). doi:https://doi.org/10.1186/s41512-024-00177-1.

Bobadilla, A.V.P., Schmitt, V., Maier, C.S., Mensing, S. and Stodtmann, S. (2024). Practical guide to SHAP analysis: Explaining supervised machine learning model predictions in drug development. Clinical and Translational Science, [online] 17(11). doi:https://doi.org/10.1111/cts.70056.

Chawande, S. (2024). Insider threats in highly automated cyber systems. *World Journal of Advanced Engineering Technology and Sciences*, [online] 13(2), pp.807–820. doi:https://doi.org/10.30574/wjaets.2024.13.2.0642.

Enterprise IT World (2024). *From 2019 to 2024, the Number of Insider Attacks Increased from 66% of Organizations to 76%*. [online] Enterprise IT World. Available at: https://www.enterpriseitworld.com/from-2019-to-2024-the-number-of-insider-attacks-increased-from-66-of-organizations-to-76/ [Accessed 7 Oct. 2025].

Fife, S.T. and Gossner, J.D. (2024). Deductive Qualitative Analysis: Evaluating, Expanding, and Refining Theory. *International Journal of Qualitative Methods*, 23(1), pp.1–12. doi:https://doi.org/10.1177/16094069241244856.

Hirtenlehner, H. and Leitgöb, H. (2024). Deterrence Perceptions, Self-Control Ability and the Moral Filter: Conceptualizing and Testing a Model of a Subsidiary Relevance of Deterrence. *Deviant Behavior*, pp.1–28. doi:https://doi.org/10.1080/01639625.2023.2298512.

Inayat, U., Farzan, M., Mahmood, S., Fahad Zia, M., Hussain, S. and Pallonetto, F. (2024). Insider threat mitigation: Systematic literature review. *Ain Shams Engineering Journal*, [online] 15(12), pp.103068–103068. doi:https://doi.org/10.1016/j.asej.2024.103068.

Istari (2023). *2023 Insider Threat Report*. [online] Istari-global.com. Available at: https://istari-global.com/insights/spotlight/2023-insider-threat-report/.

Janjua, F., Masood, A., Abbas, H. and Rashid, I. (2020). Handling Insider Threat Through Supervised Machine Learning Techniques. *Procedia Computer Science*, 177, pp.64–71. doi:https://doi.org/10.1016/j.procs.2020.10.012.

Jervis, R. (1979). Deterrence Theory Revisited. *World Politics*, [online] 31(2), pp.289–324. doi:https://doi.org/10.2307/2009945.

Kaggle (2023). *Insider Threat*. [online] Kaggle.com. Available at: https://www.kaggle.com/datasets/nuhamintamirat/insider-threat.

Khan, A.A., Chaudhari, O. and Chandra, R. (2023). A Review of Ensemble Learning and Data Augmentation Models for Class Imbalanced problems: Combination, Implementation and Evaluation. *Expert Systems with Applications*, [online] 244(122778), p.122778. doi:https://doi.org/10.1016/j.eswa.2023.122778.

Khan, A.A., Chaudhari, O. and Chandra, R. (2023). A Review of Ensemble Learning and Data Augmentation Models for Class Imbalanced problems: Combination, Implementation and Evaluation. *Expert Systems with Applications*, [online] 244(122778), p.122778. doi:https://doi.org/10.1016/j.eswa.2023.122778.

Lukman, A.F., Mohammed, S., Olaluwoye, O. and Farghali, R.A. (2024). Handling Multicollinearity and Outliers in Logistic Regression Using the Robust Kibria–Lukman Estimator. *Axioms*, 14(1), pp.19–19. doi:https://doi.org/10.3390/axioms14010019.

Matharaarachchi, S., Domaratzki, M. and Muthukumarana, S. (2024). Enhancing SMOTE for imbalanced data with abnormal minority instances. *Machine Learning with Applications*, 18, p.100597. doi:https://doi.org/10.1016/j.mlwa.2024.100597.

Mohammed, S., Budach, L., Feuerpfeil, M., Ihde, N., Nathansen, A., Noack, N., Patzlaff, H., Naumann, F. and Harmouch, H. (2025). The effects of data quality on machine learning performance on tabular data. *Information Systems*, [online] 132, p.102549. doi:https://doi.org/10.1016/j.is.2025.102549.

Naicker, T. and Niekerk, B. van (2021). Machine Learning for Insider Threat Detection. *ResearchGate*. [online] doi:https://doi.org/10.34190/EAIR.21.036.

Ogunbodede, O.O., Adewale, O.S., Alese, B.K. and Akinyokun, O.K. (2024). *Insider Threat Detection Techniques: Review of User Behavior Analytics Approach*. [online] ResearchGate. Available at: https://www.researchgate.net/publication/384241231_Insider_Threat_Detection_Techniques_Review_of_User_Behavior_Analytics_Approach.

Richard, C., Elly, B., Glory, D. and Nikolas, J. (2025). The Role of Behavioral Analytics in Insider Threat Detection and Prevention: A Multi-faceted Approach Combining Technical and Human-Centric Data. [online] Available at: https://www.researchgate.net/publication/393706151_The_Role_of_Behavioral_Analytics_in_Insider_Threat_Detection_and_Prevention_A_Multi-faceted_Approach_Combining_Technical_and_Human-Centric_Data.

Sarhan, B.B. and Altwaijry, N. (2022). Insider Threat Detection Using Machine Learning Approach. *Applied Sciences*, [online] 13(1), p.259. doi:https://doi.org/10.3390/app13010259.

Smith, G. (2024). *Insider Threat Statistics: (2024's Most Shocking Trends)*. [online] StationX. Available at: https://www.stationx.net/insider-threat-statistics/.

Xuan, F., Su, W., Chen, Z., Huang, X., Zhai, W., Li, X., Zeng, Y., Li, Z., Li, J. and Huang, J. (2025). Performance of stacking machine learning and volume model for improving corn above ground biomass prediction. *Plant Phenomics*, [online] 7(3), p.100068. doi:https://doi.org/10.1016/j.plaphe.2025.100068.

Yi, J. and Tian, Y. (2024). Insider Threat Detection Model Enhancement Using Hybrid Algorithms between Unsupervised and Supervised Learning. *Electronics*, [online] 13(5), p.973. doi:https://doi.org/10.3390/electronics13050973.

Zhu, N., Zhu, C., Zhou, L., Zhu, Y. and Zhang, X. (2022). Optimization of the Random Forest Hyperparameters for Power Industrial Control Systems Intrusion Detection Using an Improved Grid Search Algorithm. *Applied Sciences*, [online] 12(20), p.10456. doi:https://doi.org/10.3390/app122010456.