National College of Ireland

CLOUD SECURITY AND COMPLIANCE: EVALUATING LONG TERM DATA COMPLIANCE

MSc Research Project
MSc in Cyber Security

PALLAVI KARUNAKAR SALIAN
Student ID: 23267381

School of Computing
National College of Ireland

Supervisor: Joel Aleburu

National College of Ireland

National College of Ireland

Project Submission Sheet

Student Name:	Pallavi Karunakar Salian		
Student ID:	23267381		
Programme:	MSc in Cyber Security	**Year:**	2024-2025
Module:	Practicum 2		
Lecturer:	Joel Aleburu		
Submission Due Date:	11 Aug 2025		
Project Title:	Cloud Security and Compliance: Evaluating Long Term Data Compliance		
Word Count:	8194		

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.
<u>ALL</u> internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:	Pallavi Karunakar Salian
Date:	11 Aug 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. Projects should be submitted to your Programme Coordinator.
3. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
4. You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. **Late submissions will incur penalties.**
5. All projects must be submitted and passed in order to successfully complete the year. **Any project/assignment not submitted will be marked as a fail.**

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

AI Acknowledgement Supplement

[Insert Module Name]

[Insert Title of your assignment]

Your Name/Student Number	Course	Date
Pallavi Karunakar Salian	Msc in Cyber Security	11 Aug 2025

This section is a supplement to the main assignment, to be used if AI was used in any capacity in the creation of your assignment; if you have queries about how to do this, please contact your lecturer. For an example of how to fill these sections out, please click here.

AI Acknowledgment

This section acknowledges the AI tools that were utilized in the process of completing this assignment.

Tool Name	Brief Description	Link to tool

Description of AI Usage

This section provides a more detailed description of how the AI tools were used in the assignment. It includes information about the prompts given to the AI tool, the responses received, and how these responses were utilized or modified in the assignment. **One table should be used for each tool used**.

[Insert Tool Name]	
[Insert Description of use]	
[Insert Sample prompt]	[Insert Sample response]

Evidence of AI Usage

This section includes evidence of significant prompts and responses used or generated through the AI tool. It should provide a clear understanding of the extent to which the AI tool was used in the assignment. Evidence may be attached via screenshots or text.

Additional Evidence:

[Place evidence here]

Additional Evidence:

[Place evidence here]

CLOUD SECURITY AND COMPLIANCE: EVALUATING LONG TERM DATA COMPLIANCE

ABSTRACT: The goal of the proposed exploration is to research cloud security and resilience to long-lasting data breaches payloads, especially ransomware attacks that exhibit cyclostationary characteristics and their interplay with machine learning (ML) models. This paper compares three ML classification classes, AdaBoost, SGDClassifier, and Random Forest, in their capacity to identify and prevent data breach. Throughout the research, the use and adoption of adaptive and scalable cloud security tools and services were needed since cyber threats are becoming more complex and numerous. In the analysis, however, it becomes apparent that Random Forest classifier is superior to other models in terms of accuracy, precision, recall, and F1 score thanks to which it can be considered the most credible choice of detecting and managing ransomware in a cloud environment. The paper also looks at some of the most important tactics that can be utilized to enhance the security of clouds such as hybrid detection mechanisms, retraining as it occurs, fallback mechanism, and congruence with compliances. Besides, it highlights the necessity of additional progress on questions of model interpretation and resource-saving systems in real-time implementation. The study is also beneficial in cloud security because it suggests effective modification, and dynamic frameworks that can counter and manage emerging cyber threats and bolster long-range resilience.
Keywords: Cloud Security, Data Breach Resilience, Machine Learning, Ransomware, Cyclostationary Behaviour

INTRODUCTION: BACKGROUND

The evaluation of long-term data breach resilience is one of the most significant steps for cloud security and compliance in recent times. The rapid growth of cloud computing, in which the global market is expected to hit about $912.77 billion in 2025 and the expected increase in its growth by more than $5,150.92 dollars by 2034, indicating that cloud security is a top organisational priority (Precedence Research, 2025). Worldwide, approximately 70% of cloud availability zones are operated by American providers, with all the remaining 30% controlled by Chinese firms *[Refer to Figure 1]* (Buchholz, 2025). Even with this increase, it is known that 45% of all data breaches are in the cloud, so it is deeply worrying that 80% of enterprises revealed one or more cloud security events in the last 12 months (Edge Delta, 2024). These alarming statistics highlight the necessity to develop robust methods of detection, which need to be able to quickly adapt to new and more sophisticated threats, such as ransomware and attacks that have never been seen before.

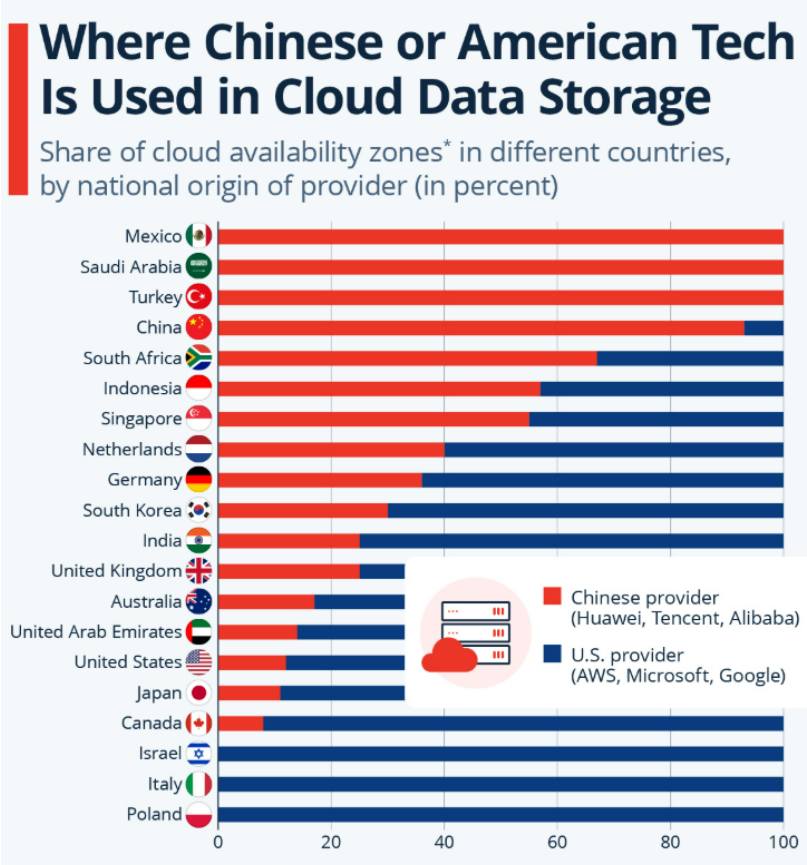

Figure 1: Usage of Cloud Data Storage in China and the American Tech Field

PROBLEM STATEMENT

Although cloud computing is widely used by organisations, there is a high possibility of failing to stop long-term data breaches. The long-term resilience is missing in the traditional forms of detection (encryption, firewalls), while they are not always very adaptable (Welsh and Benkhelifa, 2020). Traditional security methods (encryption, firewalls) rely on established detection techniques and lack the adaptability and long-term resilience which is required in cloud environments (Shukla et al., 2022). Thus, this study helps to improve the Machine Learning (ML) classification models with the ability to distinguish such threats and ensure a compliance-based, strong cloud security architecture, especially using complex threats, such as ransomware with cyclostationary behaviour.

RESEARCH AIM, OBJECTIVES, AND QUESTIONS

1.3.1 Research Aim

The research aims to assess the long-term resilience of the cloud security environment to data breaches through the analysis of cyclostationary attacks like ransomware exploits by using ML classification models.

1.3.2 Research Objectives

- To analyse previous literature related to cloud security and identify gaps to bridge in this analysis for evaluating Long-Term Data Breach Resilience using cloud computing.

- To perform and evaluate classification models (AdaBoost, SGDClassifier, and Random Forest) for the analysis of ransomware, particularly those exhibiting cyclostationary behaviour.
- To evaluate the best-performing classification model based on the evaluation metrics (Accuracy percentage, classification report, and confusion matrix visualisations) for observing data transfer patterns, and ransomware for solving long-term data breaches.
- To recommend diverse, significant strategies, including hybrid detection pipelines, continuous retraining, fallback protocols, and compliance alignment based on empirical insights.

1.3.3 Research Questions

- What are the gaps identified from the previous research studies to be bridged in this analysis for evaluating Long-Term Data Breach Resilience using cloud computing?
- How to perform and evaluate classification models (AdaBoost, SGDClassifier, and Random Forest) for the analysis of ransomware and zero-day cyberattacks, particularly those exhibiting cyclostationary behaviour?
- What is the best-fitted classifier to observe data transfer patterns, and ransomware for solving long-term data breaches?
- What are the recommended strategies based on empirical insights for compliance alignment in long-term data breaches?

1.3.4 Research Hypotheses

Alternate Hypothesis (H1): ML models (AdaBoost, SGDClassifier, LSTM, and RNN) provide a great boost in ransomware attack detection and classification in cloud-based systems since they are amazingly effective at analysing data patterns on long-term data breaches.

1.4 Research Motivation

The increasing complexity and rate of cyber threats, particularly ransomware on cloud systems, create high risks of adverse impacts on data integrity and regulatory compliance as well as organisational resilience. Security systems that are currently in place have difficulties adapting to changing patterns of threats, especially those in cyclostationary behaviour. The research is inspired by the necessity to study advanced machine learning and deep learning models that could help to improve threat detection, to properly classify the anomalies, and create long-term breach resilience by predicting ransomware. This research helps to accomplish the development of adaptive, compliance-aligned cloud security frameworks of remote digital enterprises by assessing the performance of models and planning strategic interventions for cloud security.

RELATED WORK

2.1 Introduction

This section critically analyses research on cloud breach detection and resilience, covering cyclostationary feature extraction, comparative model evaluations, metric-driven assessments, and strategic implementations. It examines aims, methodologies, findings, limitations, and debates, highlighting insights and gaps for future cloud security solutions.

2.2 Critical Analysis of Previous Research Studies

2.2.1 Cyclostationary Behaviour Analysis for Cloud-Based Breach Detection

Analysing cyclostationary behaviour enhances periodic feature extraction for resilient cloud breach detection against ransomware in cloud security environments. Nkongolo, Deventer and Kasongo (2022) model cyclostationary malware detection based on Boruta feature selection and on periodic signal features extraction via “PCA (Principal Component Analysis)”. They apply methods mechanisms to network traffic, wherein the performance of classifiers against the malware types is considered. Findings state a detection rate of more than 95% of Random Forest classifier, which is unaffected by various forms of threats such as ransomware signatures. Con among others is that it is dependent on synthetic datasets and that it also has computational overhead. Although the method drastically improves the relevancy of the features, against the method, one observes that the complexity of Boruta can interfere with real-time cloud implementation.

On the same note, Genetic algorithms are suggested to use in detecting the zero-day threats Nkongolo et al. (2022) present a cloud-based framework in which genetic algorithms to weight features and support ensemble learning are used. They are optimized with classifier parameters to find abnormalities and measure the performance in data sets. The results indicate enhanced recall and accuracy reporting on new threats such as ransomware. The limitations are overfitting, high computational cost, and bias on dataset. On the one hand, the ensembles enhance the precision, but counterarguments remind that genetic tuning might not be working with the real-time streams.

Rathna (2024), on the contrary, develops a cloud detection system centred around LSTM networks conditioned by time-series telemetry in detecting ransomware and zero-day attacks. The approach cyclostationary preprocesses the features to deep models sequentially. It shows detection accuracy and alerts of more than 90 %. Constrains include high requirements of labelled data, open to concept drift, and high requirements of resources. As much as the application of deep learning improves on dynamic detection, the counterarguments reveal shortcomings in interpretability as well as the possibility of scaling difficulties, in the event of cloud implementation.

2.2.2 Comparative Analysis of Classification Models for Breach Detection

Comparative analysis evaluates classification models' effectiveness and resilience metrics for breach detection against ransomware threats. Raghunath et al. (2022) develop an "XRC (XGBoost Regression Classifier)" model integrating XGBoost with Inception V4 to classify cyberattacks, including ransomware variants. Evaluated on BoT-IoT and NSL-KDD, the model achieves over 97% detection in nine of thirteen scenarios and at least 75% accuracy on challenging datasets on R2L and U2R attacks. Limitations include computational overhead and integration complexity, prompting debate on practicality versus performance.

However, Rani, Nag and Shahriyar (2024) propose a comprehensive data-driven breach classification framework using Naive Bayes, logistic regression, SVM and random forest on Privacy Rights Clearinghouse data. Random forest achieved 89.7% accuracy, 0.88 F1, SVM yielded 87.3% accuracy, 0.85 F1. Methods include LDA-based topic modeling to enhance features of Topic distributions. Limitations include dataset bias and interpretability concerns, prompting debate on complexity versus practicality. Whereas Yeboah et al. (2024) survey 81 breach detection studies and analyse 40 incidents, including ransomware cases, to use classifiers, initiative-taking strategies and early, intermediate, and late fusion. The taxonomy outlines ten feature extraction methods like "statistical moment analysis, time-domain signal features," four initiative-taking techniques like self-labelling, data augmentation, and three fusion types like early fusion, intermediate fusion, and late fusion. Limitations include reliance on secondary sources and absence of performance metrics, prompting debate on empirical rigour versus conceptual scope.

2.2.3 Metric-Driven Model Evaluation for Long-Term Breach Solutions

Through the utilisation of comprehensive and measurable security metrics, metric-based assessment frameworks offer a sound mechanism for gauging the long-term resilience against breaches. Tallam (2025) aims to develop security-first AI models that raise the standards of robustness and trustworthiness by integrating formal verification methods with adversarial training, and their paper addresses the attack that ransomware poses. Synthetic datasets and model interpretability measures supply the investigators with an analysis of system robustness and the premise that architecture decisions are directionally guided by security metrics. Their findings demonstrate better threat detection rates and indicate computational overhead as a weakness. In comparison, Ismail et al. (2024) evaluate a smart system of medication engagement, which focuses on the compliance and security indicators that would guarantee patient data and integrity. The researchers apply a mixed design where the machine learning classifiers are applied to identify anomaly-based data tampering through ransomware. According to their discovery, they found increased detection performance and compliance, but the system relies on a substantial scale of labelled datasets, which becomes a problem when used in constrained environments.

Similarly, Stray et al. (2023) have collected the methods that integrate human values into recommender systems and achieve the alignment between algorithmic goals and measures and requirements. Systematic literature review, together with the use of workshops, reveals to them that metric-based evaluation is fundamental to ethical conformance and to determining the soundness of a system. They additionally emphasise combating adversarial manipulation, comparing the corresponding attacks to ransomware intruding on the purchase recommendations channels. Their results indicated enhanced trust, but a conflict between values and performance expressions. Weaknesses comprise the methodological differences in approaches and the impossibility of generalising between different system architectures.

2.2.4 Strategic Implementation for Enhancing Cloud Breach Resilience

The application of strategy in the implementation enables the resilience of cloud breach through automated monitoring, integrated console, and reaction to ransomware. Akinade et al. (2024) seek to survey the problems and remedies related to cloud security by reviewing the best practices that counter attacks, such as ransomware. Using a literature review of 80 papers, the authors single out a layered defence approach, encryption systems, and ordered checks on compliance with ISO 27001 and SOC 2 standards. The results indicate that combination authentication, intrusion detection and incident response strategies enhance the durability of long-term breaches. Weaknesses are that they rely on secondary data, may not be empirically or experimentally justified or may be rendered outdated by changes in ransomware methods. Likewise, Ahmadi (2024) stated that a systematic review of the security threats to cloud computing and security mitigation strategies, with reference to incident detection of such attacks as ransomware. With the help of PRISMA-guided analysis, this study evaluates papers that classify threat vectors and defence mechanisms. Its limitations are the possible selection bias, the narrow period of threat detection, and the coverage of the new ransomware variants. In contrast, Abdullayeva (2023) addresses the cyber resilience issues and security in cloud computing, creating a model and evaluating model-based resilience metrics, such as availability, reliability, and integrity, to measure the reactions to ransomware-like incidents. The experiment introduces control-theoretic models like PID controllers and state-space feedback control, and resiliency curves of the cloud computing infrastructure. The results indicate that adaptive redundancy and dynamic resource allocation reduce virtual machine and service downtimes, which leads to enhanced availability and faster recovery of cloud operations. However, it has drawbacks such as high computational cost and assumptions about workloads.

2.3 Theoretical Underpinning

The suitable framework for this entire research study is the ***Technology-Organisation-Environment (TOE) Framework***. TOE theory reveals how the environment in the use of technological innovations is enhanced by three major contexts, namely, the technology readiness, organisational capabilities, and environmental pressures (Eveland and Tornatzky, 1990; Gama and Magistretti, 2023). The TOE framework aids in this assessment of how organisations are embracing and using the implementation of ML classification models (technological context), meeting compliance requirements and data governance (organisational context), and tackling external cybersecurity threats (malware, phishing attacks) and regulatory requirements (environmental context) ***[Refer to Figure 2]***. The theory helps to consider systemic enablers and constraints that affect cloud security resilience using adaptive threat detection systems by machine learning methods.

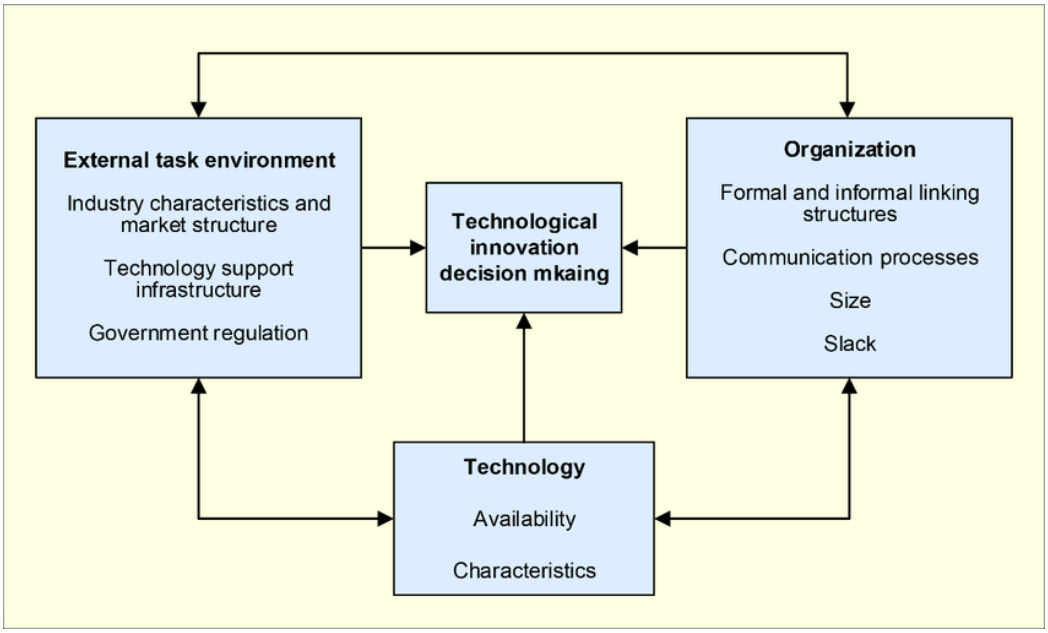

Figure 2: Framework of Technology-Organisation-Environment (TOE) Theory
(Hadwer et al., 2021)

2.4 Literature Gap

1. Existing research advances specialised detection techniques but often lacks an integrated framework unifying cyclostationary analysis, classification modelling, metric-driven evaluation, and strategic implementation. Empirical validation under realistic cloud conditions and evolving ransomware attacks remains scarce. Trade-offs between detection accuracy, computational overhead, and model interpretability are underexplored. Adaptive updating mechanisms and standardised benchmarks are missing, hindering cross-study comparison. There is a need for a comprehensive, scalable, and context-aware framework that balances performance, resource efficiency, and robustness to ensure sustained long-term cloud resilience.

2.5 Conceptual Framework

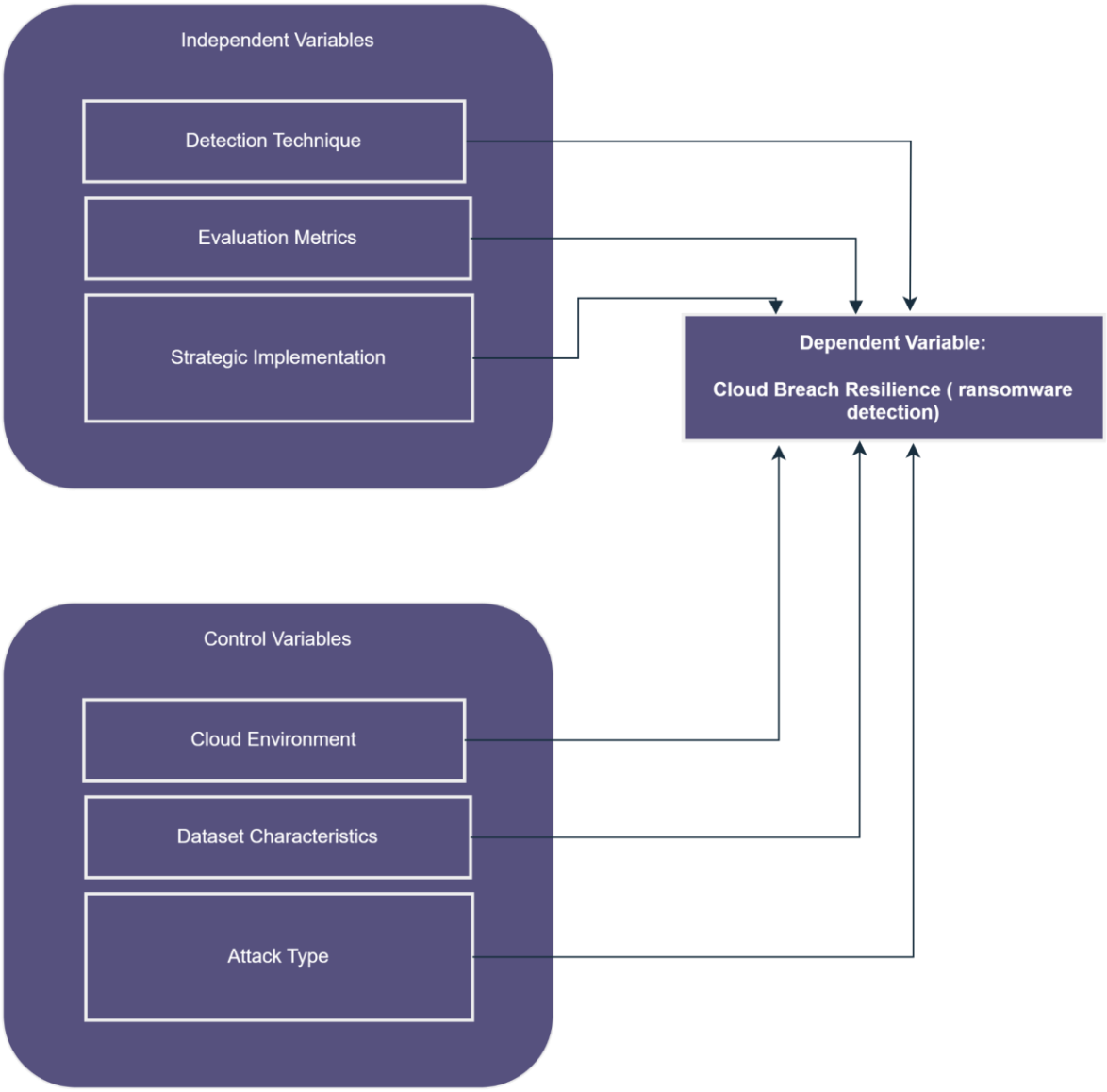

Figure 3: Conceptual Framework

2.6 Chapter Summary

This chapter systematically critiques cloud breach detection research, encompassing cyclostationary feature analysis, model comparisons, metric-based evaluations, and strategic resilience implementations. It highlights methodological strengths, limitations, and evolving challenges, guiding robust, adaptive frameworks against ransomware and zero-day threats in cloud environments.

RESEARCH METHODOLOGY

3.1 Introduction

This chapter helps in outlining the entire research framework to evaluate the long-term resilience of compliance and cloud security against data breaches like ransomware. This also reflects the research approach, design, technical framework for assessing the efficiency of cloud security strategies. In addition, the reasoning for choosing the dataset and machine learning (ML) models is explained, highlighting their applicability in forecasting potential security risks and improving breach resilience. The chapter is concluded with a synopsis of the methodological approach, paving the way for the following analysis.

3.2 Research Approach

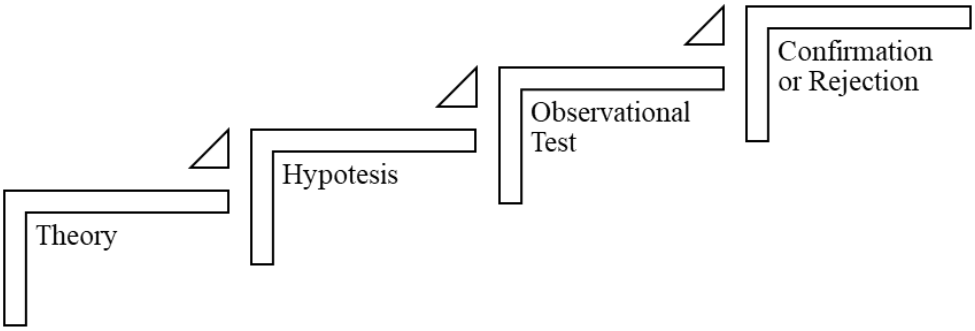

Figure 4: Deductive Research Approach

The ***Deductive research approach*** has been employed in this study for quantitative data to assess the long-run resilience of cloud security against data breaches like ransomware. Through examination of quantitative information from secondary sources like studies, databases, and reports, this deductive approach enables the determination of patterns, correlations, and factors linked to data (Naeem et al., 2023). A deductive research approach is used to validate prevailing theories of cloud security and compliance, basing them on existing data and known frameworks. The approach also proves useful for concluding previous knowledge so that the findings are more generalizable and applicable to a greater context of data breaches like ransomware ***[Refer to Figure 4]***. It also enables the study to concentrate on assessing the efficacy of existing cloud security controls in real environments over time.

3.3 Research Design

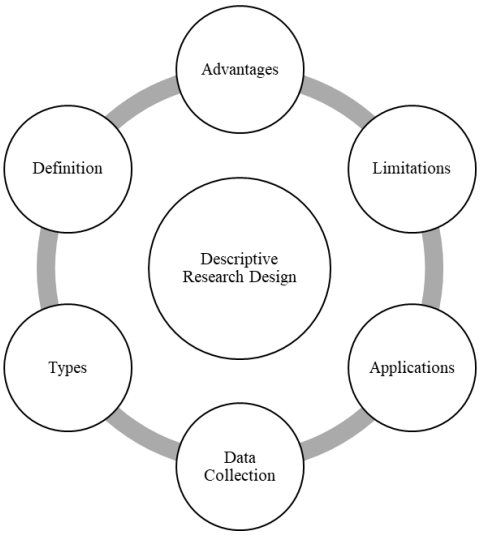

Figure 5: Descriptive Research Design

The research design that has been used in this study is ***Descriptive***, which enables an in-depth exploration of cloud security and compliance procedures for long-term resilience against data breaches. Descriptive research design includes collecting and analysing systematically secondary quantitative data from a range of sources, including industry reports and secondary data (Siedlecki, 2020). Descriptive research is perfect for establishing and recording the attributes of cloud security controls

and their efficacy over time. Through the research design of an extensive overview of cloud security, compliance frameworks, and past data breach trends, this method assists in clarifying the extent and magnitude of current security actions ***[Refer to Figure 5]***. It also aids in determining areas where cloud resilience may be enhanced. Thus, this descriptive research design enables the identification of existing conditions and trends without even manipulating features, providing insights into how security strategies have been employed in the real world with their influence on reducing data breaches.

3.4 Technical framework

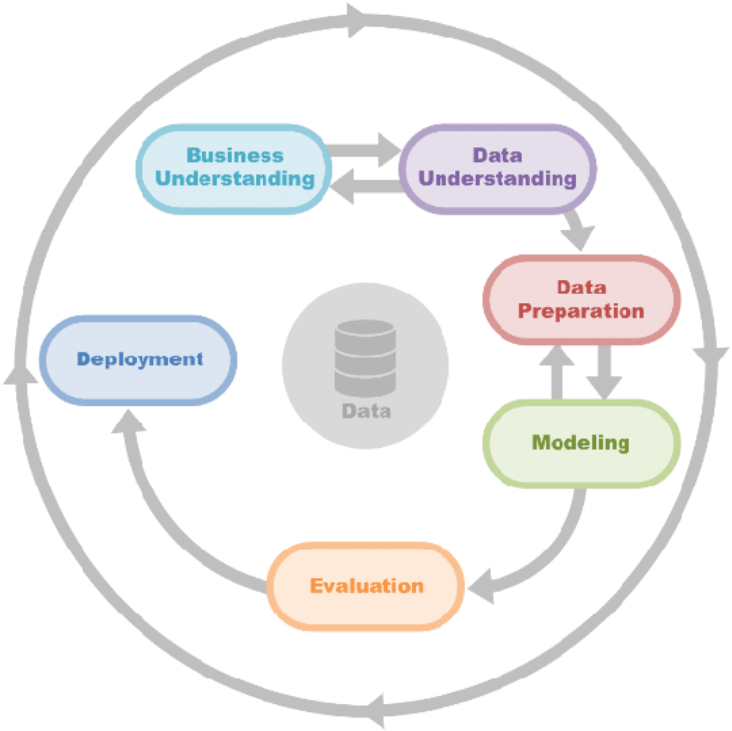

Figure 6: CRISP-DM Framework
(Source: Tounsi, Anoun and Hassouni, 2020)

The **CRISP-DM (Cross-Industry Standard Process for Data Mining)** model has been used in this research to systematically examine cloud security information and determine long-term data breach resilience in case of attacks like ransomware happen. The CRISP-DM framework ensures a holistic and iterative analysis and aligns with the goals (Shimaoka, Ferreira, and Goldman, 2024). CRISP-DM is made up of six essential steps, as shown in **Figure 6**:

- ❖ **Business Understanding:** Establishes the goals of analysing cloud security resilience to data breaches for ransomware, in line with the research objective of determining the adequacy of current security controls.
- ❖ **Data Understanding:** This entails gathering and investigating secondary data, including breach reports and compliance information, to determine pertinent patterns and security vulnerabilities.
- ❖ **Data Preparation:** Data cleansing, conversion, and pre-processing is conducted to make the data consistent, complete, and ready for analysis.
- ❖ **Modelling:** Machine learning algorithms, including classification, are used to forecast breaches using historical information.
- ❖ **Evaluation:** Outputs from the models are evaluated to quantify cloud security practice effectiveness and resistance to breaches like ransomware.
- ❖ **Deployment:** Analysis findings are used to suggest enhancements to cloud security frameworks.

3.5 Data Collection

The ***UGRansome dataset***, collected from Kaggle, is the key data source for ***ransomware and zero-day cyberattack analysis***, especially those showing cyclostationary characteristics. It contains 207,533 samples from 14 different families of ransomware, labelled as Normal Behaviour, Abnormal

Behaviour, and Cyclostationary Patterns (Kaggle, 2023). The dataset contains "Prediction" (SS, A, and S), which is the target variable, needed for network intrusion detection system (ML models) training and testing.

This data set supports the purpose of the study by allowing the analysis of past research concerning cloud security and the detection of gaps to fill in assessing long-term data breach resiliency with the use of cloud computing. It makes it possible to perform and evaluate classification algorithms (AdaBoost, SGDClassifier, and Random Forest) for ransomware analysis, particularly those that show cyclostationary characteristics. The structure of the dataset facilitates the comparison of the performance of the top-performing classification model with measures like percentage accuracy, classification report, and confusion matrix plots for monitoring data transfer patterns and ransomware. It also helps in suggesting various and impactful strategies such as hybrid detection pipelines, ongoing retraining, fallback procedures, and compliance alignment based on empirical findings.

3.6 Justification of Machine Learning Models

Classification Models	Strength	Justification
AdaBoost Classifier	• Combines multiple weak classifiers to create a strong classifier. • Highly effective in handling imbalanced datasets. • It boosts performance in cloud security enhancement and achieves 99.5% accuracy (Attou et al., 2025).	AdaBoost has been efficiently applied in anomaly detection and ransomware detection in earlier research, with high classification accuracy. It is highly resistant to overfitting and enhances performance at every iteration.
Random Forest Classifier	• Handles large datasets effectively. • Provides feature importance insights. • It is used in predicting data security breaches in AI-powered cloud applications and achieved 98.5% accuracy (S. Hassan Abdul Cader and Dr. K. Nirmala, 2024)	Random Forest has been widely employed in cybersecurity because it can efficiently process complex, high-dimensional data. It enhances the accuracy of predictions by averaging over several decision trees.
Stochastic Gradient Descent (SGD) Classifier	• Efficient for large-scale datasets. • Works well with sparse data and high-dimensional features. • It has been used as a secure encryption algorithm for cloud data storage and achieved 98.45% accuracy (Suganya and Sasipraba, 2023).	SGD Classifier has been utilised in previous research for effective classification of security events. It is chosen based on its speed and capacity to support large data sets with high-dimensional data attributes, which is an issue of relevance in this study.

Table 1: Justification of Selecting ML models.

3.7 Chapter Summary

This chapter provides the research design for assessing cloud security resilience to data breaches. Applying a deductive strategy and secondary quantitative data, descriptive design, and CRISP-DM methodology, the research examines ransomware trends via classification models (AdaBoost, Random Forest, SGD Classifier) to furnish executable insights into security enhancement.

DESIGN SPECIFICATION

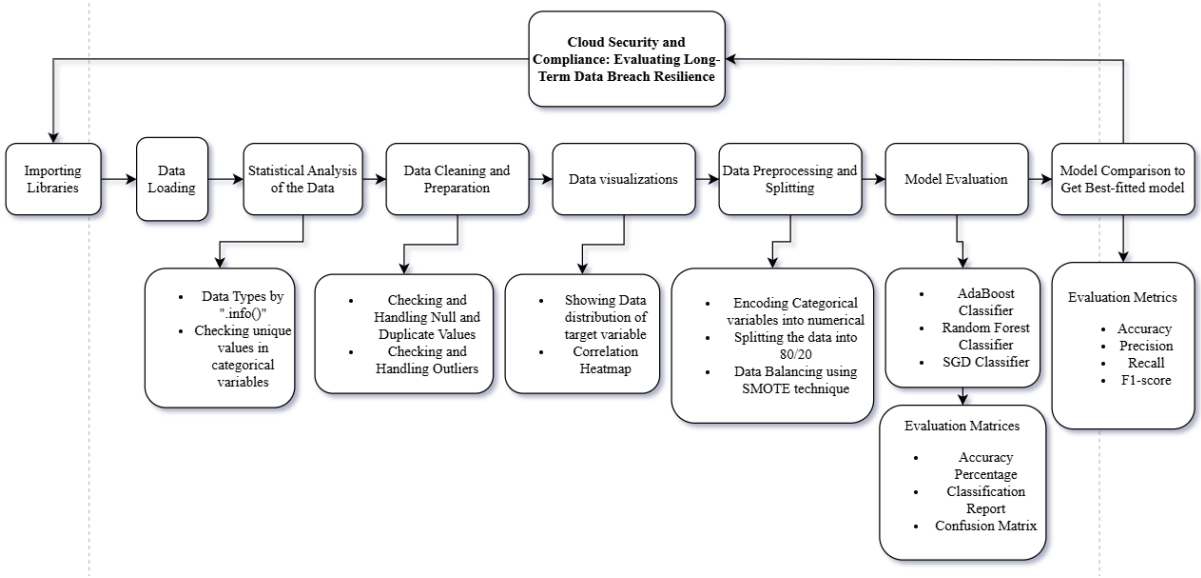

Figure 7: Methodological Architecture

The flowchart in **Figure 7** summarises the most important steps in the methodology for assessing cloud security and compliance regarding long-term data breach robustness, by employing machine learning.

- **Importing Libraries:** This is the initial step where required Python libraries (such as Pandas, Scikit-learn, and Matplotlib) are imported to conduct data manipulation, machine learning, and visualisation activities. This ensures an efficient data analysis process.
- **Data Loading:** Firstly, the data is loaded into the system to be analysed, and it is important since it guarantees that the proper data is present to process and train the models.
- **Statistical Analysis of the Data:** The step consists of verifying the data types, detecting missing values, and checking unique values for categorical variables. Statistical analysis also assists in detecting outliers and anomalies that may impact model performance. Checking unique values is crucial for comprehending the structure of the dataset and preparing clean, actionable data for subsequent steps (Aldoseri, Khalifa, and Hamouda, 2023).
- **Data Cleaning and Preparation:** Managing missing values, duplicates, and outliers helps to provide high-quality data, which is critical for precise model predictions (AlSalehy and Bailey, 2025).
- **Data Visualisations:** The visualisations of the distribution of the data and correlation heatmap assist with pattern and relationship detection among variables, which informs the model development process.
- **Data Preprocessing and Splitting:** Here, the categorical features are converted into numerical form, and the dataset is divided into training and test sets based on an 80/20 ratio. SMOTE is used to balance the dataset, resolving any imbalances in classes that could influence model performance (Matharaarachchi, Domaratzki and Muthukumarana, 2024).
- **Model Evaluation and Comparison:** Three models of classification, such as AdaBoost, Random Forest, and SGD Classifier, are evaluated and compared in terms of accuracy, precision, recall, F1-score, classification report, and confusion matrices. The comparison is necessary to identify the best model to use in detecting and preventing long-term cloud security data breaches in cases where organisations are facing a data breach due to a ransomware attack.
- Breach due to a ransomware attack.

IMPLEMENTATION

```
# Importing necessary libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.ensemble import AdaBoostClassifier, RandomForestClassifier
from sklearn.linear_model import SGDClassifier
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.metrics import f1_score, accuracy_score, precision_score, recall_score
from imblearn.over_sampling import SMOTE
from tensorflow.keras.models import Sequential
from tensorflow.keras.optimizers import Adam


import warnings
warnings.filterwarnings("ignore")
```

```
# Load dataset
data = pd.read_csv('final(2).csv')
data.head()
```

	Time	Protcol	Flag	Family	Clusters	SeddAddress	ExpAddress	BTC	USD	Netflow_Bytes	IPaddress	Threats	Port	Prediction
0	50	TCP	A	WannaCry	1	1DA11mPS	1BonuSr7	1	500	5	A	Bonet	5061	SS
1	40	TCP	A	WannaCry	1	1DA11mPS	1BonuSr7	1	504	8	A	Bonet	5061	SS
2	30	TCP	A	WannaCry	1	1DA11mPS	1BonuSr7	1	508	7	A	Bonet	5061	SS
3	20	TCP	A	WannaCry	1	1DA11mPS	1BonuSr7	1	512	15	A	Bonet	5061	SS
4	57	TCP	A	WannaCry	1	1DA11mPS	1BonuSr7	1	516	9	A	Bonet	5061	SS

Table 1: Import Libraries and Load the dataset.

The image shows the necessary Python libraries being imported for the analysis, including common libraries like pandas, numpy, seaborn, and matplotlib for data manipulation and visualisation. It also imports machine learning models from sklearn like Ada Boost Classifier, RandomForestClassifier and SGDClassifier, along with tools for data preprocessing (StandardScaler, LabelEncoder) and handling class imbalance (SMOTE). The warnings.filterwarnings("ignore") line suppresses unnecessary warnings.

The dataset being loaded and the first few rows of the dataset (data.head()). It shows that the dataset includes columns like Time, Protocol, Flag, Family, Clusters, BTC, USD, and Prediction. Aligning with the topic of network traffic analysis, the dataset contains features related to network behaviour, such as protocols, flags, and addresses, to classify traffic as either benign or malicious. The Prediction column is the target variable, which would indicate whether the network traffic is malicious. Both figures demonstrate essential steps for data analysis, starting from loading libraries and data to preparing the dataset for machine learning tasks.

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 149043 entries, 0 to 149042
Data columns (total 14 columns):
 #   Column          Non-Null Count   Dtype
---  ------          --------------   -----
 0   Time            149043 non-null  int64
 1   Protcol         149043 non-null  object
 2   Flag            149043 non-null  object
 3   Family          149043 non-null  object
 4   Clusters        149043 non-null  int64
 5   SeddAddress     149043 non-null  object
 6   ExpAddress      149043 non-null  object
 7   BTC             149043 non-null  int64
 8   USD             149043 non-null  int64
 9   Netflow_Bytes   149043 non-null  int64
 10  IPaddress       149043 non-null  object
 11  Threats         149043 non-null  object
 12  Port            149043 non-null  int64
 13  Prediction      149043 non-null  object
dtypes: int64(6), object(8)
memory usage: 15.9+ MB
```

```
data.isnull().sum()
```

```
Time             0
Protcol          0
Flag             0
Family           0
Clusters         0
SeddAddress      0
ExpAddress       0
BTC              0
USD              0
Netflow_Bytes    0
IPaddress        0
Threats          0
Port             0
Prediction       0
dtype: int64
```

```
data.duplicated().sum()
```

```
0
```

Table 2: Checking data info, null values, and outliers.

The figure presents the output of data.info(): as it appears, the dataset will have 149,043 rows and 14 columns. There are no missing values in any of the columns (the number of non-null values is equal to the number of entries). Columns in the dataset are of different data types, including int64 data type in numbers of features (Time, BTC, USD and so on) and object data type in categorical features

(Protocol, Flag, Family and so on). This means that the data structure is clean to the extent of null values, but mixed data types exist, which require close preprocessing, particularly for the problem of machine learning.
The second picture represents the output of data.isnull().sum(), which proves the fact that there is no missing data in any columns. It is critical to machine learning because missing data may affect the performance of the model. The lack of data may negatively influence the quality of the model, its accuracy and add errors, which is why data preprocessing is essential to manage such issues. The third picture shows the results of the duplicate check (data.duplicated().sum()), which proves that there are no similar rows. This implies that there is no duplicate data, and each point of data is unique. This is best practice since duplicate rows distort the results, especially where they are used in classification problems, resulting in overfitting.

```
categorical_columns = data.select_dtypes(include=['object']).columns

# Check unique values for each categorical column
for column in categorical_columns:
    print(f"Unique values in '{column}':")
    print(data[column].unique())
    print("\n")
```

```
Unique values in 'Protcol':
['TCP' 'UDP' 'ICMP']


Unique values in 'Flag':
['A' 'APSF' 'APS' 'ARF' 'AP' 'R' 'AF' 'APRSF' 'ASF']


Unique values in 'Family':
['WannaCry' 'Locky' 'TowerWeb' 'CryptXXX' 'Cryptohitman' 'CryptoLocker'
 'NoobCrypt' 'DMALocker' 'Globev3' 'CryptoLocker2015' 'Flyper' 'SamSam'
 'Globe' 'EDA2' 'JigSaw' 'APT' 'Razy']


Unique values in 'SeddAddress':
['1DA11mPS' '1NKi9AK5' '1AEoiHYZ' '1KZKcvx4' '17dcMo4V' '1GZkujBR']


Unique values in 'ExpAddress':
['1BonuSr7' '1SYSTEMQ' '1CLag5cd' '1Lc7xTpP' '1DiCeTjB' '18e372GN' '1']


Unique values in 'IPaddress':
['A' 'B' 'C' 'D']


Unique values in 'Threats':
['Bonet' 'DoS' 'Spam' 'Blacklist' 'SSH' 'UDP Scan' 'Scan' 'NerisBonet'
 'Port Scanning']


Unique values in 'Prediction':
['SS' 'A' 'S']
```

```
# Define the numerical columns in the dataset
numerical_columns = data.select_dtypes(include=[np.number]).columns

# Calculate the IQR for numerical columns
Q1 = data[numerical_columns].quantile(0.25)
Q3 = data[numerical_columns].quantile(0.75)
IQR = Q3 - Q1

# Define outlier condition: values outside the 1.5 * IQR range are outliers
outliers_condition = ((data[numerical_columns] < (Q1 - 1.5 * IQR)) | (data[numerical_columns] > (Q3 + 1.5 * IQR)))

# Drop the rows where any column has outliers
data_cleaned = data[~outliers_condition.any(axis=1)]

# Verify if the rows with outliers have been removed
print(f"Original dataset size: {data.shape}")
print(f"Dataset size after removing outliers: {data_cleaned.shape}")
```

```
Original dataset size: (149043, 14)
Dataset size after removing outliers: (110289, 14)
```

Table 3: Check unique values and Outliers.

The image shows that it checks the unique values for each categorical column in the dataset. The code prints out the unique entries for columns like “Protocol, Flag, Family, SeddAddress, ExpAddress, IPaddress, Threats, and Prediction.” The output highlights a diverse range of values in these categorical columns, such as multiple “protocol types (TCP, UDP, ICMP),” “threat categories (Dos, Spam, SSH),” and various address formats. For example, the Family column includes multiple types of malware families, indicating a variety of attack types. The Prediction column has 3 “unique values (anomaly (A), signature (S), and synthetic signature (SS))”, representing different classifications of network traffic. This diversity suggests that the dataset involves complex categorical data, which will need careful encoding for machine learning models.

The figure provides the steps for outlier detection and removal, and using the IQR method, outliers are identified based on the +/- 1.5 * IQR range and removed. The dataset shrinks from 149,043 rows to 110,289 rows, indicating the removal of outlier data points. This is crucial, as outliers can disproportionately affect model performance, particularly with sensitive models.

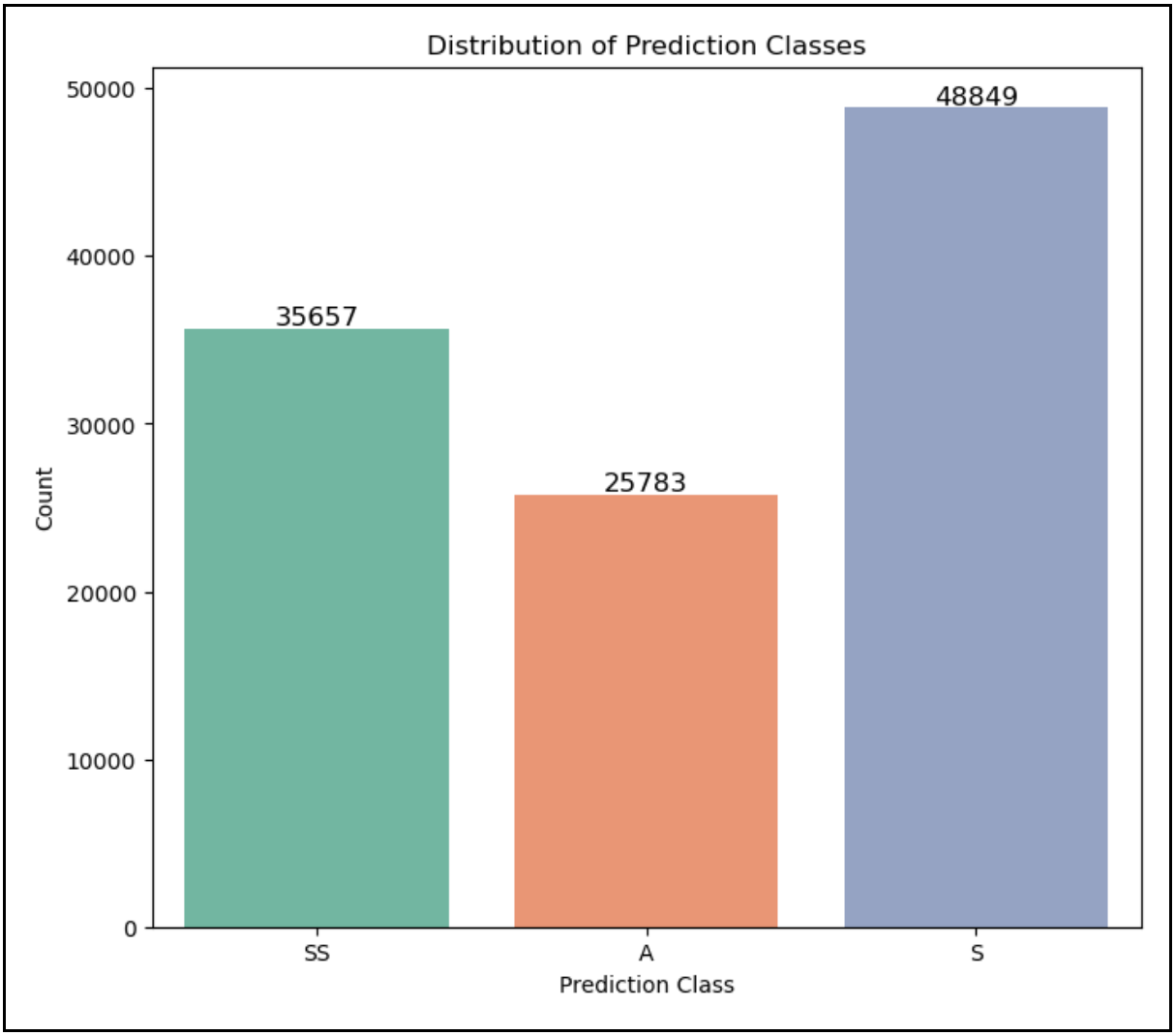

Table 4: Distribution of Prediction Classes

The image displays the distribution of the Prediction classes in the dataset, using a bar plot. It shows that class S (with 48,849 counts) is the most frequent, followed by class SS (35,657 counts), and class A (25,783 counts) are the least frequent. This distribution suggests that the dataset is imbalanced, with class S being dominant. This imbalance could potentially affect the performance of machine learning models, as they might be biased towards predicting the majority class (S). To address this, techniques like SMOTE have been used to balance the dataset and improve model accuracy for minority classes.

```
# Encoding categorical columns into numeric values
label_encoder = LabelEncoder()

# List of columns that need encoding
categorical_columns = ['Protcol', 'Flag', 'Family', 'SeddAddress', 'ExpAddress', 'Threats', 'IPaddress', 'Port', 'Prediction']

for col in categorical_columns:
    data_cleaned[col] = label_encoder.fit_transform(data_cleaned[col])
```

```
# Selecting features and target
X = data_cleaned.drop(columns=['Prediction'])
y = data_cleaned['Prediction']
```

Table 5: Encoding categorical columns and selecting the target.

Label Encoding is performed on the "categorical columns (Protocol, Flag, Family, etc.)" using the LabelEncoder from sklearn. This step is crucial because machine learning models can only interpret numeric values, so categorical data needs to be converted into numeric formats for model compatibility. Encoding ensures the machine learning algorithm can process these features correctly. In the image, features (X) and the target variable (y) are selected. The features are stored in X after dropping the target column (Prediction), which is stored in y. This separation is important for training and testing models, ensuring the model learns from the features and predicts the target variable. Together, these steps are foundational for preparing data for machine learning, ensuring that the model is trained on clean, well-preprocessed data.

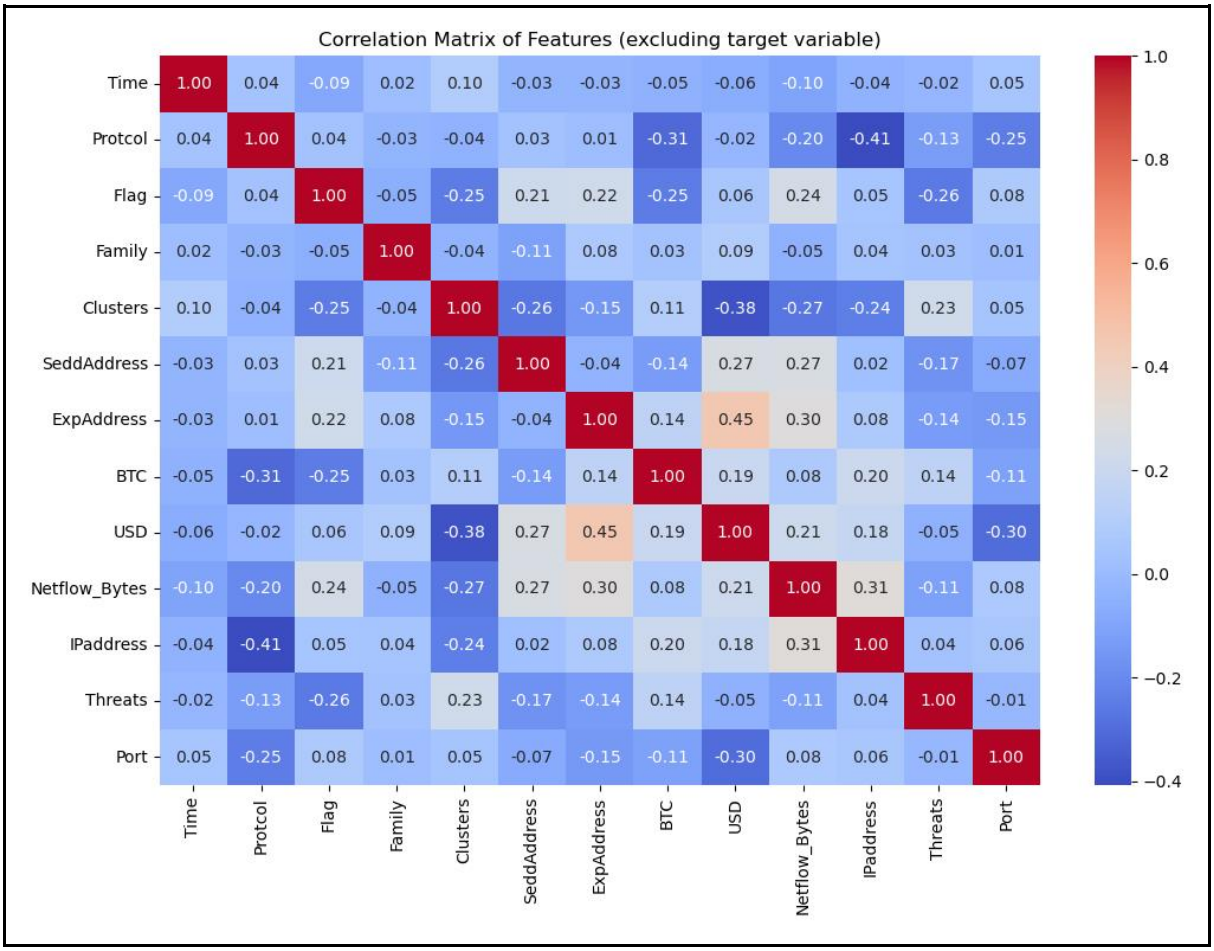

Figure 9: Correlation Heatmap

The figure below shows the correlation among all the characteristics of the dataset. The correlations of the features such as USD and Netflow_Bytes are positive with 0.45, and between the BTC and USD are 0.38. Such correlation values do not show that there is much multicollinearity. Multicollinearity happens when several features have a high correlation (normally above 0.7 or below -0.7) and may result in corresponding data redundancy. Multicollinearity may cause gross adjustments of model coefficients, so one might have difficulties comparing the use of individual features.

```
# Split data into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
# SMOTE: Oversampling the minority class in the training set
smote = SMOTE(random_state=42)
X_train_res, y_train_res = smote.fit_resample(X_train, y_train)
```

```
# Scaling numerical features
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train_res)
X_test_scaled = scaler.transform(X_test)
```

Table 6: Data split, SMOTE, and Scaling

The data is divided into a training set and a test set in 80-20 proportions (test_size=0.2). This is to make sure that 80% of the information can be used in training the model, and 20% will be withheld to assess it. The reason it is vital is that this split plays a key role in validating the model, avoiding overfitting, and the model generalises properly to previously unseen data. SMOTE is used to oversample the minority class within the training set to make sure that the model will not be biased towards the majority one. SMOTE can address the imbalance in the class distribution by creating synthetic samples of the underrepresented class, thereby leading to better model performance (particularly in the case of imbalanced classes). The numerical features are scaled with StandardScaler, and this step standardises the data such that there is a mean of 0 and a standard deviation of 1 within all the features, as this is vital to many machine learning approaches that are sensitive to the scale of input features.

EVALUATION

6.1 Model Evaluation

```
AdaBoostClassifier
AdaBoostClassifier(random_state=42)
```

AdaBoost Classifier Classification Report

	precision	recall	f1-score	support
0	0.74	0.73	0.73	5154
1	0.93	0.91	0.92	9790
2	0.89	0.93	0.91	7114
accuracy			0.87	22058
macro avg	0.85	0.86	0.85	22058
weighted avg	0.87	0.87	0.87	22058

Accuracy: 87.39%

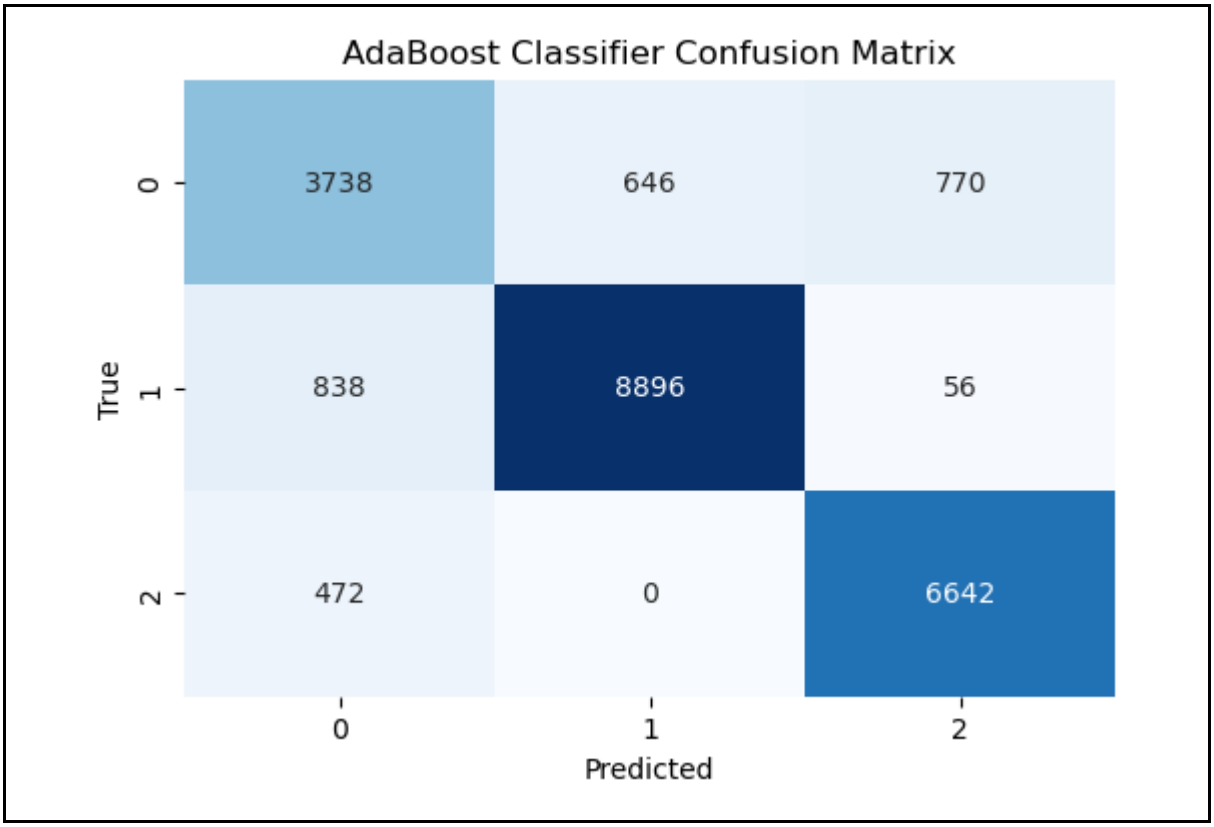

Table 7: AdaBoost classifier model classification report and confusion matrix

The AdaBoost classifier shows reliable performance with an overall accuracy of 87.31%. The classification report indicates high precision and F1-scores, recall, for class 1 (0.93, 0.91, 0.92), suggesting that the model performs well in predicting the majority class. However, "class 0 (precision 0.74)" and "class 2 (precision 0.89)" show slightly lower performance. The lower precision for class 0, particularly with 646 instances misclassified as class 1, could be a concern for cloud security, as it may lead to false alarms or unnecessary actions.

The AdaBoost classifier shows that class 1 (true positives are 8896) is predicted well, but there are notable misclassifications for class 0, with 646 instances misclassified as class 1 and 770 as class 2. Additionally, class 2 has 472 misclassifications as class 0. These issues indicate that class imbalances need to be addressed for improved data breach detection and resilience. The research objective of evaluating data breach resilience through predictions may benefit from this model's performance, but improvement in handling minority class predictions is necessary to avoid misclassifications that could result in overlooked breaches.

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

Random Forest Classifier Classification Report

	precision	recall	f1-score	support
0	0.99	0.99	0.99	5154
1	1.00	1.00	1.00	9790
2	1.00	1.00	1.00	7114
accuracy			1.00	22058
macro avg	0.99	0.99	0.99	22058
weighted avg	1.00	1.00	1.00	22058

Accuracy: 99.54%

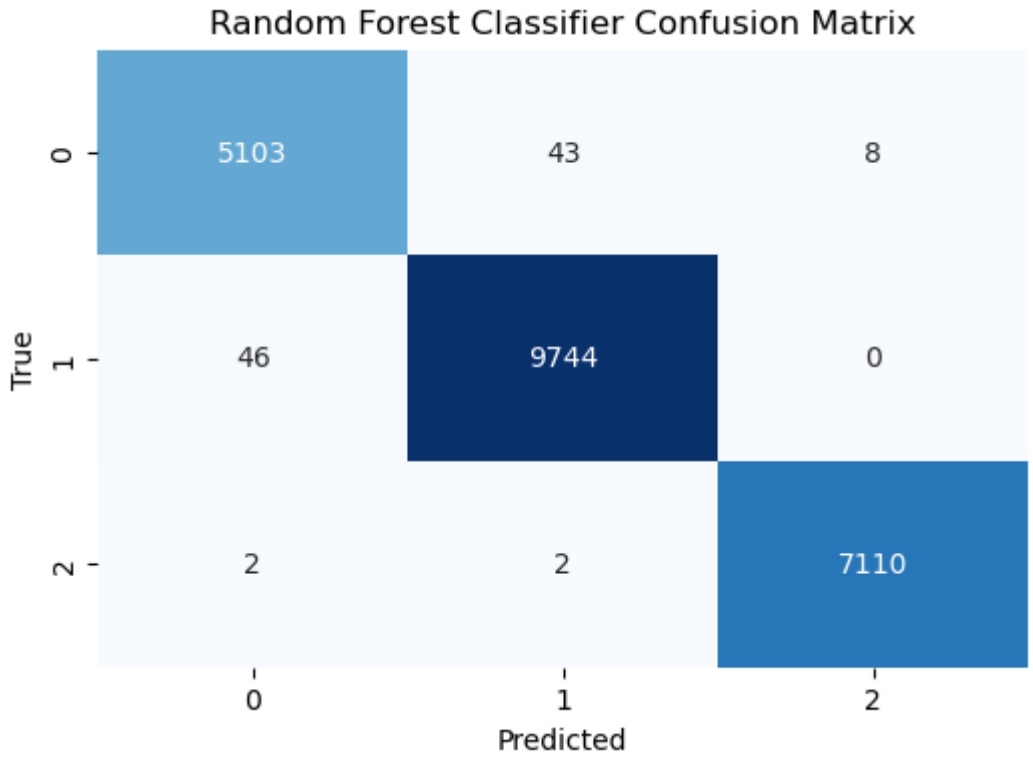

Table 8: Random Forest classifier classification report and confusion matrix

The Random Forest classifier achieves an impressive accuracy of 99.54%, with near-perfect precision, recall, and F1-scores for all classes (1.00 for classes 1 and 2, 0.99 for class 0). The confusion matrix confirms minimal misclassifications, with 5103, 9744, and 7110 true positives for classes 0, 1, and 2. This high performance suggests that the model is highly effective for predicting data breaches, aligning with the research objective of evaluating long-term breach resilience in cloud security when organisations are facing problems data breaches related to ransomware. Despite the strong results, continuous monitoring and updates are recommended to maintain performance. Overall, the model offers a reliable and scalable solution for cloud security predictions when ransomware happens.

SGDClassifier

SGDClassifier(random_state=42)

SGDClassifier Classification Report

	precision	recall	f1-score	support
0	0.77	0.60	0.67	5154
1	0.88	0.95	0.91	9790
2	0.83	0.88	0.85	7114
accuracy			0.84	22058
macro avg	0.83	0.81	0.81	22058
weighted avg	0.84	0.84	0.84	22058

Accuracy: 84.27%

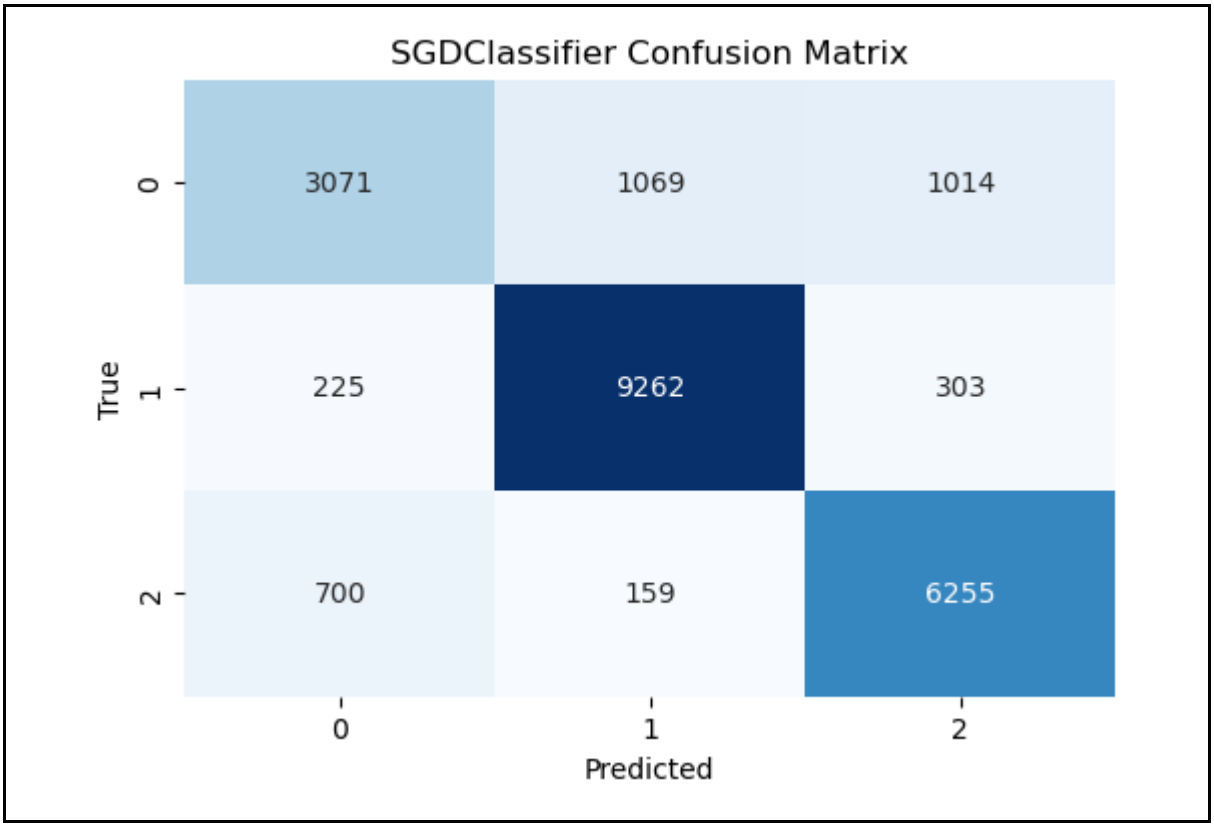

Table 9: SGD classifier classification report and confusion matrix

The SGD classifier achieves an accuracy of 84.27%, which is an improvement over simpler models but still lower than that of Random Forest or AdaBoost. The classification report highlights strengths and weaknesses, precision for class 0 is 0.77, indicating a high number of false positives, which could lead to unnecessary alerts or actions in the context of data breach ransomware detection. However, recall for class 1 is strong at 0.95, making the model effective at identifying breaches or risk events. The recall for classes 0 and 2 is lower (0.60 and 0.88), suggesting that some instances of these classes may be missed. The confusion matrix shows significant misclassifications, especially class 0, which is often predicted as class 1 or class 2. While the SGD classifier performs well for predicting breaches (class 0), its performance for classes 1 and 2 needs improvement. This could be achieved through further tuning or a different approach, like "Hyperparameter tuning," to reduce false positives and missed predictions, crucial for accurately predicting data breach resilience in cloud security.

	Model	Accuracy	Precision	Recall	F1 Score
0	AdaBoost	0.873878	0.873645	0.873878	0.873502
1	Random Forest	0.995421	0.995420	0.995421	0.995420
2	SGDClassifier	0.842687	0.837862	0.842687	0.836970

```
Best Model based on Accuracy: Random Forest
Best Model based on Precision: Random Forest
Best Model based on Recall: Random Forest
Best Model based on F1 Score: Random Forest
```

Table 10: Best-fitted Model

The model comparison table shows that "Random Forest" outperforms "AdaBoost and SGD classifiers" across all evaluation metrics. With an accuracy of 99.54%, and precision, recall, and F1 scores all at 1.00 for classes 1 and 2, it is the best model. AdaBoost and SGDC classifiers exhibit lower performance in comparison to Random Forest, particularly in recall and F1 scores for certain classes. Given the results, "Random Forest" is the most reliable and robust model for predicting data breaches, making it the best choice for evaluating long-term data breach resilience in cloud security. Random Forest's high accuracy and precision will enhance long-term cloud security by effectively predicting and mitigating ransomware attack risks.

6.2 Findings

The entire analysis shows how meticulous data cleansing and machine learning-based classification can efficiently evaluate and enhance cloud security robustness against sustained data compromises in case organisations face a ransomware attack. Preliminary data cleansing and preparation ensured clean inputs through the removal of outliers, duplications, and ambiguous data types. Visual interpretation of feature distributions and inter-feature relationships helped identify prominent cloud security risk indicators and inform feature engineering and feature encoding. Synthetic oversampling (SMOTE) to balance the dataset eliminated class imbalance and allowed for unbiased model training. Among the compared classifiers, ensemble techniques outperformed consistently, as the Random Forest model displayed the highest robustness and reliability in separating benign from malicious traffic patterns, while AdaBoost provided high precision through the emphasis on misclassified examples. The SGD classifier, although computationally efficient at handling large volumes of data, was less sensitive to minority breach classes and could benefit from additional tuning.

6.3 Discussion

Upon reviewing the theoretical and practical assessments of cloud breach detection frameworks, one can identify several major insights and challenges that might be used to shape contemporary research regarding the development of cloud security resilience. Possibilities of expanding the possibilities of detecting ransomware have been revealed by theoretical methods such as cyclostationary behaviour analysis and advanced machine learning methods such as Boruta feature selection and LSTM networks, among others. The methods can make it easier to extract the periodic features that can be used to effectively detect dynamic threats in cloud systems (Nkongolo et al., 2022). Nevertheless, realistic issues pose the problem of the high computational burden of such models and the aspect of relying on synthetic data, especially when used on real-time cloud systems. Also, overfitting and data imbalance still prevail, which may harm the overall performance in a general context and in real-time.
Conversely, in comparing the ML models, “Random Forest, AdaBoost, and SGD classifiers,” evaluation reveals more definite results. The “Random Forest model” has shown a high accuracy of 99.54% as it predicted the data breach problem, including ransomware attacks, very well because of its precision, recall, and F1 scores (Raghunath et al., 2022). Such performance points to its capability to work with complicated, real-life data. Nevertheless, although such classifiers as AdaBoost and SGD are helpful, they also have certain issues such as false detectives and imbalances in classes, especially the class 0, inducing irrelevant alerts and interfering with the accuracy of a model in identifying a certain type of a breach (Rani, Nag and Shahriyar, 2024).
To solve these problems, strategies, such as the use of SMOTE to deal with the imbalanced data and hyperparameter tuning to enhance the work of the classifier, were applied (Yeboah et al., 2024). These are some helpful solutions to optimising the accuracy of a model and minimising the danger of misclassifications. Although the “Random Forest” outcomes are encouraging, it is imperative to constantly oversee and revise them to make the model responsive to emerging malicious developments like new types of ransomwares. Therefore, the steps between the theory of breach detection and the development of complex methods and the very practice of new model training, relationship tuning and model adaptation are crucial to reliable and long-term resilience to cloud breach. Use of Data Analysis with Advanced Machine Learning models will be the most promising one, as it provides the flexibility to manage real-time challenges that involve cloud security information.

Conclusion and Future Work

Conclusion

The goal of the study was to assess the durability of cloud security and resilience to data breach in the long-term scenario by detecting the ransomware attacks characterized by cyclostationary patterns with the help of machine learning models. The authors used AdaBoost, SGDClassifier, and Random Forest algorithms in the assessment of the cloud breach resilience. The Random Forest model became the most accurate with higher results in all indicators, such as accuracy, precision, recall, and F1, and therefore is credible to detect the manifestations of especially long-term data breaches ransomware.
The entire goals of the study were achieved. The first one, literature review on cloud security and understanding of gaps, was also perfectly met, as the analysis showed that the previous literature resulted in a gap of integrated frameworks. The second goal of discussing the machine learning models

was achieved by the extensive comparison of AdaBoost, SGDClassifier, and Random Forest. The third goal, which was to determine the best model with the best performance in detecting a long-term breach, was fulfilled with Random Forest being the best performing model. The fourth goal, which had been to propose approaches towards enhancing cloud security resilience, was also achieved as the recommendation was to use fallback protocol, continuously retrain, hybrid detection pipelines, and align to compliance. These results present a specific and handy method of enhancing cloud security and compliance resilience about the development of threats, such as ransomware. Summing up, the overall purpose of the research was achieved with the help of the combination of literature review, empirical analysis, model assessment, and strategic suggests that play a significant role in promoting long-term cloud security resilience.

LIMITATIONS

Although the research offers excellent findings on cloud security resilience there are several weaknesses to it. To begin with, the study relies on the secondary datasets, which is Kaggle UGRansome, which may not reflect the real-world cloud settings in their entirety and therefore the findings lack generalizability. Also, training of models on synthetic data, especially under cyclostationary behavior could be biasing and limit the performance of the model in managing varied and real time situations. The computational requirements of such state-of-the-art models as Random Forest or AdaBoost may be an obstacle to scaling the detection systems deployed in live cloud environments, where resource limitations are the usual thing. Lastly, the models used in the study can have problems with identifying complicated trends and managing dense data that need to be tuned and enhanced to be applied in dynamic threat environment more widely.

FUTURE RECOMMENDATION

Future works will be justified on scaling and flexibility of machine learning models in detecting cloud breaches in real time, especially in mix-and-match environments in which emerging threats using ransomware are more widespread. The generalizability of the models would be enhanced by using more diverse and practical datasets, other than purely synthetic and Kaggle datasets. Besides, advanced methods in model interpretation like explainable AI (XAI), etc. should be investigated as a potential solution to the problem of deciphering complex patterns identified through machine learning models. The other suggestion is the incorporation of hybrid systems of detectors that have several algorithms in place to avoid the weaknesses of a single model. Moreover, a constant retraining of the model on new data, as well as a pre-meditated alignment with compliance and the enforcement of fall-back systems, will guarantee the long-term resilience of cloud security. Finally, it will also be important to deal with resource constraints and computing efficiency regarding real-time deployment in cloud deployment.

7.4 Theoretical and practical implications

7.4.1 Theoretical implications

The proposed study helps to fill the gap in the theoretical knowledge regarding cloud security resilience as it combines machine learning models namely ensemble methods with the analysis of the cyclostationary behaviour to implement the recognition of ransomware attacks. It adds to the current literature by filling the gaps in the research of long-term breach detecting and the necessity in the adaptive and scaled clouds security approaches. The paper also contributes to the use of Technology-Organisation-Environment (TOE) framework in which technology, organisational ability, and environmental pressure are related to cloud security practices.

7.4.2 Practical implications

In practice, the results underline the significance of applying superior machine learning frameworks, such as Random Forest, when dealing with real-time protection cloud applications. The study suggests pragmatic solutions, and one of them is hybrid detection systems and ongoing model retraining, as well as compliance alignment, to enhance cloud breach resilience in the long term. Organizations can use these findings to improve their cloud security models with superior defense against advanced threats such as ransomware.

REFERENCES

Abdullayeva, F. (2023). Cyber resilience and cyber security issues of intelligent cloud computing systems. *Results in Control and Optimization*, [online] 12, p.100268. doi:https://doi.org/10.1016/j.rico.2023.100268.

Ahmadi, S. (2024). Systematic Literature Review on Cloud Computing Security: Threats and Mitigation Strategies. *International Journal of Information Security*, [online] 15(02), pp.148–167. doi:https://doi.org/10.4236/jis.2024.152010.

Akinade, A.O., Adepoju, P.A., Ige, A.B. and Afolabi, A.I. (2024). Cloud Security Challenges and Solutions: A Review of Current Best Practices. *International Journal of Multidisciplinary Research and Growth Evaluation*, 6(1), pp.26–35. doi:https://doi.org/10.54660/.ijmrge.2025.6.1.26-35.

Aldoseri, A., Khalifa, K.N.A. - and Hamouda, A.M. (2023). Re-Thinking Data Strategy and Integration for Artificial Intelligence: Concepts, Opportunities, and Challenges. *Applied Sciences*, [online] 13(12), p.7082. Available at: https://www.mdpi.com/2076-3417/13/12/7082.

AlSalehy, A.S. and Bailey, M. (2025). Improving Time Series Data Quality: Identifying Outliers and Managing Missing Values in a Multilocation Gas and Weather Dataset. *Smart Cities*, [online] 8(3), p.82. doi:https://doi.org/10.3390/smartcities8030082.

Attou, H., Guezzaz, A., Benkirane, S. and Azrour, M. (2025). Cloud Security Enhancement Through RBFNN and AdaBoost-Based Model. [online] doi:https://doi.org/10.13140/RG.2.2.12954.96965.

Buchholz, K. (2025). *Infographic: Where Chinese or American Tech Is Used in Cloud Data Storage*. [online] Statista Daily Data. Available at: https://www.statista.com/chart/34404/share-of-cloud-availability-zones-by-national-origin-of-provider/ [Accessed 25 Jun. 2025].

Edge Delta (2024). *Top Cloud Security Statistics in 2024 | Edge Delta*. [online] Edge Delta. Available at: https://edgedelta.com/company/blog/cloud-security-statistics/.

Eveland, J. and Tornatzky, L.G. (1990). *Technological Innovation as a Process*. [online] ResearchGate. Available at: https://www.researchgate.net/publication/291824703_Technological_Innovation_as_a_Process.

Gama, F. and Magistretti, S. (2023). Artificial intelligence in innovation management: A review of innovation capabilities and a taxonomy of AI applications. *Journal of Product Innovation Management*, 42(1). doi:https://doi.org/10.1111/jpim.12698.

Hadwer, A.A., Tavana, M., Gillis, D. and Rezania, D. (2021). A Systematic Review of Organizational Factors Impacting Cloud-based Technology Adoption Using Technology-Organization-Environment Framework. *Internet of Things*, [online] 15, p.100407. doi:https://doi.org/10.1016/j.iot.2021.100407.

Ismail, A., Naeem, M., Syed, M.H., Abbas, M. and Coronato, A. (2024). Advancing Patient Care with an Intelligent and Personalized Medication Engagement System. *Information*, 15(10), p.609. doi:https://doi.org/10.3390/info15100609.

Kaggle (2023). Generative mathematical models for ransomware attack prediction using Chi-Square feature selection for enhanced accuracy. *Signal, Image and Video Processing*, [online] 19(10). doi:https://doi.org/10.1007/s11760-025-04396-x.

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.

Naeem, M., Ozuem, W., Howell, K. and Ranfagni, S. (2023). A step-by-step Process of Thematic Analysis to Develop a Conceptual Model in Qualitative Research. *International Journal of Qualitative Methods*, [online] 22(1), pp.1–18. doi:https://doi.org/10.1177/16094069231205789.

Nkongolo, M., Deventer, J.P. van and Kasongo, S.M. (2022). The Application of Cyclostationary Malware Detection Using Boruta and PCA. *Lecture Notes on Data Engineering and Communications Technologies*, [online] pp.547–562. doi:https://doi.org/10.1007/978-981-19-3035-5_41.

Nkongolo, M., Van Deventer, J.P., Kasongo, S.M., Zahra, S.R. and Kipongo, J. (2022). A Cloud Based Optimization Method for Zero-Day Threats Detection Using Genetic Algorithm and Ensemble Learning. *Electronics*, 11(11), p.1749. doi:https://doi.org/10.3390/electronics11111749.

Precedence Research (2025). *Cloud Computing Market Size to Surpass US$ 1,614.10 BN by 2030*. [online] www.precedenceresearch.com. Available at: https://www.precedenceresearch.com/cloud-computing-market.

Raghunath, K.M.K., Kumar, V.V., Venkatesan, M., Singh, K.K., Mahesh, T.R. and Singh, A. (2022). XGBoost Regression Classifier (XRC) Model for Cyber Attack Detection and Classification Using Inception V4. *Journal of Web Engineering*. doi:https://doi.org/10.13052/jwe1540-9589.21413.

Rani, P., Nag, A.K. and Shahriyar, R. (2024). A Data-Driven Classification Framework for Cybersecurity Breaches. *IT Professional*, 26(2), pp.39–48. doi:https://doi.org/10.1109/mitp.2024.3374096.

Rathna, S. (2024). Cloud-Centric Detection Framework for Ransomware and Zero-Day Attacks Using LSTM Networks. *International Journal of Engineering Research and Science & Technology*, 20(3). doi:https://www.researchgate.net/profile/Rathna-Suruli-Andavar/publication/392760250_Cloud-Centric_Detection_Framework_for_Ransomware_and_Zero-Day_Attacks_Using_LSTM_Networks/links/6851720c24267473b7785e19/Cloud-Centric-Detection-Framework-for-Ransomware-and-Zero-Day-Attacks-Using-LSTM-Networks.pdf.

S. Hassan Abdul Cader and Dr. K. Nirmala (2024). Predicting Security Breaches in AI-Powered Mobile Cloud Applications Using Deep Random Forest Algorithm. *Nanotechnology Perceptions*, pp.748–763. doi:https://doi.org/10.62441/nano-ntp.vi.2849.

Shimaoka, A.M., Ferreira, R.C. and Goldman, A. (2024). The evolution of CRISP-DM for Data Science: Methods, Processes and Frameworks. *SBC Reviews on Computer Science*, [online] 4(1), pp.28–43. Available at: https://journals-sol.sbc.org.br/index.php/reviews/article/view/3757.

Shukla, A., Katt, B., Nweke, L.O., Yeng, P.K. and Weldehawaryat, G.K. (2022). System security assurance: A systematic literature review. *Computer Science Review*, 45, p.100496. doi:https://doi.org/10.1016/j.cosrev.2022.100496.

Siedlecki, S.L. (2020). Understanding descriptive research designs and methods. *Clinical Nurse Specialist*, [online] 34(1), pp.8–12. doi:https://doi.org/10.1097/NUR.0000000000000493.

Stray, J., Halevy, A., Assar, P., Hadfield-Menell, D., Boutilier, C., Ashar, A., Bakalar, C., Beattie, L., Ekstrand, M.D., Leibowicz, C., Connie Moon Sehat, Johansen, S.L., Kerlin, L., Vickrey, D., Singh, S., Sanne Vrijenhoek, Zhang, A.X., Andrus, M., Natali Helberger and Polina Proutskova (2023). Building Human Values into Recommender Systems: An Interdisciplinary Synthesis. *ACM Transactions on Recommender Systems*. doi:https://doi.org/10.1145/3632297.

Suganya, M. and Sasipraba, T. (2023). Stochastic Gradient Descent long short-term memory based secure encryption algorithm for cloud data storage and retrieval in cloud computing environment. *Journal of Cloud Computing*, 12(1). doi:https://doi.org/10.1186/s13677-023-00442-6.

Tallam, K. (2025). *Security-First AI: Foundations for Robust and Trustworthy Systems*. [online] arXiv.org. Available at: https://arxiv.org/abs/2504.16110 [Accessed 25 Jun. 2025].

Tounsi, Y., Anoun, H. and Hassouni, L. (2020). CSMAS: Improving Multi-Agent Credit Scoring System by Integrating Big Data and the new generation of Gradient Boosting Algorithms. *CSMAS: Improving Multi-Agent Credit Scoring System by Integrating Big Data and the new generation of Gradient Boosting Algorithms*. doi:https://doi.org/10.1145/3386723.3387851.

Welsh, T. and Benkhelifa, E. (2020). On Resilience in Cloud Computing. *ACM Computing Surveys*, 53(3), pp.1–36. doi:https://doi.org/10.1145/3388922.

Yeboah, P., Kayes, A.S.M., Rahayu, W., Pardede, E. and Mahbub, S. (2024). A Survey of Machine Learning Approaches to Cyber Data Breach Detection: Current Challenges and Future Research Directions. [online] doi:https://doi.org/10.36227/techrxiv.171925731.16789355/v1.