An Internet of Things Forensics Framework Validated by Machine Learning

Pantaleon Lutta

A thesis submitted in partial fulfilment of the requirements of Staffordshire University for the degree of

Doctor of Philosophy

May 2024

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this thesis are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other University. This thesis is the result of my own work and includes nothing which is the outcome of work done in collaboration, except where specifically indicated in the text.

Pantaleon Lutta

Acknowledgements

I wish to convey my deepest appreciation to my principal supervisor, Professor Mohamed Sedky. It is truly difficult to articulate the extent of knowledge and inspiration I have gained from him during the course of my research. His patient guidance, invaluable advice, and unwavering motivation have been a constant source of support. I extend my heartfelt gratitude for his continuous understanding and unwavering support over the span of the past five years. Without his exceptional mentorship and persistent assistance, the completion of this thesis would not have been attainable. I consider myself incredibly fortunate to have had the privilege of collaborating with him.

My gratitude also extends to my esteemed supervisory team members; Mr. Mohamed Hassan, Dr. Benhur Bastaki, Dr. Ange Aly, and Dr. Uchitha Jayawickrama for their instrumental contributions, consistent moral support, and time invested in refining this research. A special mention to Dr. Nasser Alshammari for his support and Mr. Chris Howard for taking his time to proofread this thesis.

Acknowledgment is due to the Department of Computing at Staffordshire University for granting me the scholarship that enabled the pursuit of this research endeavour. The Graduate School's patience and understanding, evident in the numerous extensions granted, have been instrumental.

I am indebted to my fellow research colleagues within the PhD lab. Our collaborative efforts and mutual support transcended the realm of colleagues, resembling more of a tightly knit family. I extend my best wishes for their research work.

Finally, my heartfelt thanks go to my family and friends for their unwavering support throughout this academic odyssey. Your consistent inquiry of "Is it done yet?" serves as a reminder of your genuine interest. I am elated to finally affirm that indeed, it is done. Thank you from the depths of my heart!

List of Publications

Lutta, P., Sedky, M. and Hassan, M. (2020) ‘The Forensic Swing of Things: The Current Legal and Technical Challenges of IoT Forensics’, *International Journal of Computer and Information Engineering*, 14(5), pp. 159–165. Available at: https://publications.waset.org/10011217/the-forensic-swing-of-things-the-current-legal-and-technical-challenges-of-iot-forensics.

Lutta, P., Sedky, M., Hassan, M., Jayawickrama, U. and Bakhtiari Bastaki, B. (2021) ‘The complexity of internet of things forensics: A state-of-the-art review’, *Forensic Science International: Digital Investigation*. Elsevier Ltd, 38, p. 301210. doi: 10.1016/j.fsidi.2021.301210.

Published Datasets

Lutta, P. (2024) ‘Smart Home Dataset’. Mendeley Data, 1. doi: 10.17632/ZGSW84B2FF.1.

Submitted Publication (immediate access but not yet peer reviewed)

Lutta, P., Sedky, M., Hassan, M., Bastaki, B. and Aly, A. (2024) ‘Towards a Practical IoT Forensic Process Through a Simulated Smart Home Environment’. doi: 10.2139/SSRN.4699694.

Publication under Preparation

Enhancing Digital Forensics Analysis of Big IoT Data: Leveraging HI-SDR for Anomaly Detection.

Abstract

The Internet of Things (IoT) has witnessed unprecedented growth, revolutionising the way we interact with connected devices and services. While IoT offers numerous benefits, it presents unique challenges for digital forensics due to the sheer volume and diverse formats of data generated. The varied array of devices, operating systems, and communication protocols further complicates investigations, demanding tailored approaches for information extraction. The absence of standardised regulations within IoT forensics adds to the complexity, hindering consistency and reliability. The real-time nature of IoT also requires novel forensic methods that align with dynamic data flows. This thesis presents a comprehensive review of IoT forensics, addressing the complexities of investigating connected environments and contributing novel methodologies and frameworks to the field of digital forensics.

This thesis proposes a comprehensive IoT framework that addresses the legal and technical challenges of IoT forensic processes to be validated by Machine Learning techniques that aid in the examination and analysis of digital forensic data collected from smart home environments.

The thesis begins with a comprehensive review of the status of IoT forensics through a systematic literature review that explores the current legal and technical challenges of IoT forensics and emphasising the uniqueness of IoT forensics.

A novel IoT digital forensics investigation framework is presented, offering a structured approach to investigations in IoT environments. This framework outlines four key phases, from preparation, live investigation, offline investigation to presentation, and is designed to tackle the unique challenges posed by IoT investigations, particularly the high volume of data. The framework is further validated through the integration of machine learning techniques, demonstrating its practical applicability in smart home environments.

The scarcity of datasets that depict real life IoT scenarios for digital forensics use is a big challenge for IoT forensics researchers. Therefore, this thesis explores different smart home simulation strategies and tools and employs a simulator. The simulator is used to simulate a dataset based on hypothetically created digital forensic case scenarios that mimic a real-life smart home inhabitant.

A new approach is proposed for the use of Hash Indexed Sparse Distributed Representation (HI-SDR) as an input to state-of-the-art anomaly detectors. This technique enhances the accuracy of anomaly detection algorithms, contributing to improved digital forensics investigations in IoT environments. HI-SDR improves feature representations, enabling robust detection of anomalies such as intrusions, variant activities, and deviations from the smart home norm, even in the presence of noise. The results demonstrate that the inclusion of HI-SDR enhances the overall performance of anomaly detection. For instance, there was an impressive improvement of 17% in accuracy and an astonishing leap of over 45% in recall compared to the state-of-art models (OCSVM and Isolation Forest). Additionally, in the case of Isolation Forest, the precision score witnessed a remarkable boost from 27% to 49%, an uplift of 22%. Moreover, the F1 measure, a pivotal metric capturing the equilibrium between precision and recall, experienced a substantial 29% improvement, ascending from an initial score of 36% to an impressive 65%. These percentages underscore the evident enhancements attributed to the strategic combination of HI-SDR and machine learning models. This strategic combination of HI-SDR and machine learning models not only addresses the challenges posed by the unique characteristics of IoT data but also contributes substantively to the advancement of digital forensics in smart environments.

This thesis demonstrates the effectiveness of the proposed framework through the integration of machine learning algorithms, specifically the HI-SDR employed for anomaly detection. This significantly improves the accuracy and efficiency of identifying suspicious activities in smart home environments, and hence aids in the analysis of high volume of data for digital forensic purposes.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ADLs	Activities of Daily Living
AD	Anomaly Detection
aLOCI	approximate Local Correlation Integral
AI	Artificial Intelligence
AIoT	Artificial Intelligence of Things
ANN	Artificial Neural Networks
ACPO	Association of Chief Police Officers
CCTV	Closed-Circuit Television
CoT	Cloud of Things
CSP	Cloud Service Providers
CBLOF	Cluster-Based Local Outlier Factor
CMGOS	Clustering-Based Multivariate Gaussian Outlier Score
COF	Connectivity-Based Outlier Factor
DSS	Decision Support System
DOS	Denial of Service
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
IDFIF-IoT	Digital Forensic Investigation Framework for the Internet of Things
DF	Digital Forensics
EU	European Union
GDPR	General Data Protection Regulations
HI-SDR	Hash-Indexed Sparse Distributed Representation
HTM	Hierarchical Temporal Memory
HBOS	Histogram-Based Outlier Score
IIoT	Industrial Internet of Things
INFLO	Influenced Outlierness
IoA	Internet of Anything
IoE	Internet of Everything
IoT	Internet of Things
IDS	Intrusion Detection Systems
k-NN	k-Nearest Neighbour
LoS	Last on Site
LEA	Law Enforcement Agencies
LOCI	Local Correlation Integral

LDCOF	Local Density Cluster-Based Outlier Factor
LOF	Local Outlier Factor
LoOP	Local Outlier Probability
ML	Machine Learning
NIST	National Institute of Standards and Technology
OCSVM	One-class Support Vector Machine
OpenSHS	Open Smart Home Simulator
ODIN	Out-Of-Distribution Detector
PC	Personal Computers
PCA	Principal Component Analysis
rPCA	Robust Principal Component Analysis
SWGDE	Scientific Working Group on Digital Evidence
SLR	Systematic Literature Review
SDR	Sparse Distributed Representation
SNN	Spiking Neural Network
SOPs	Standard Operating Procedures
SCADA	Supervisory Control and Data Acquisition
SVM	Support Vector Machine

CHAPTER 1. INTRODUCTION

This chapter serves as an introduction to the thesis and provides an overview of the driving motivations behind the research's central themes. It introduces the research questions and outlines the specific research objectives, shedding light on the contributions to knowledge that this study brings. Finally, this chapter details the research methodology employed and concludes by providing a comprehensive breakdown of the thesis structure, chapter by chapter.

1.1 Background

The Internet of Things (IoT) landscape has experienced rapid growth in recent years, marked by the proliferation of diverse IoT devices and applications. While certain IoT devices adopt a passive approach to conserve energy and activate only, when necessary, there is a notable presence of constantly connected and accessible devices that can be operated remotely over the internet from virtually any location. The pervasive interconnection within the IoT ecosystem has ushered in a hyperconnected era where various entities, including individuals, IoT devices, cloud storage, mobile applications, and data, seamlessly interact with one another. This heightened level of interconnectivity has blurred the boundaries between virtual reality and the physical world, rendering many online services as tangible experiences rather than mere virtual events. Consequently, new terminologies such as hyperconnectivity, Internet of Everything (IoE), and Artificial Intelligence of Things (AIoT) have emerged to encapsulate and describe this transformative phenomenon (Kim, Park and Lee, 2023; Mohamed, Koroniotis and Moustafa, 2023).

This rapid growth and usefulness of IoT has seen it being deployed in critical and strategic infrastructure sectors like healthcare, transport, agriculture, home automation, and smart industries among many others (Yaqoob *et al.*, 2019). According to a report by Cisco (2016) on the state of IoT, it is expected that by 2030, there will be over 500 billion connected by the internet. The report also stated that the IoT business was estimated to have a revenue turnover of around $14.4 trillion by the year 2022. This revelation indeed shows that the number of connected IoT devices has already surpassed the human population in the world. The benefits of comfort and reliability of IoT technologies to human beings have brought with them security concerns. Policy makers and law enforcement agencies have been left to scamper in finding ways to address these security and privacy concerns. This is due to its large-scale connectivity and over reliance on the internet for communication making it susceptible to cyberattacks. As was way back highlighted by Hegarty, Lamb and Attwood (2014), digital forensics experts still face a daunting task of handling these cyberattacks because of the unique and complex challenges posed by IoT like big IoT data, cross border legal jurisdiction, and maintaining a chain of custody. Kebande and Ray (2016) note that recently, researchers have been drawn to finding solutions to these challenges, however, this is still in its infancy.

IoT forensics can be defined as a branch of digital forensics that combines three levels namely, device level forensics, network level forensics and cloud level forensics. This is explained further by Zawoad and Hasan (2015) who stated that IoT forensics involves the investigation of IoT infrastructure (device, network, and cloud). This is whereby local memories of IoT devices could be investigated for potential evidence, network log files could be retrieved to reveal user activities and the cloud being a major storage of IoT device data could be a source of potential evidence.

IoT forensics is a cumbersome process as there is no standardisation of the IoT products, no or limited historical data is stored on the devices and them being always connected makes them extremely volatile (Conti *et al.*, 2018).

The complexity around the extraction of data from IoT environments is a major setback in the ability of producing legally admissible evidence in a court of law Kebande and Ray (2016).

These complexities are attributed to the following reasons as brought out by Hegarty, Lamb and Attwood (2014).

- The IoT spectrum poses a huge uncertainty of the originality of data, the storage mechanisms, the attributes associated with the data, and the privacy rights of the data.

- There is a struggle to secure and maintain a chain of custody because of the highly volatile IoT data.

- There are difficulties in applying the traditional digital forensics tools to extract data which could be stored on the cloud.

- Legal complexities due to cross-border jurisdiction, multi-tenancy, and service level agreements.
- Varied and proprietary storage mechanisms of data which has very limited visibility due to IoT devices being resource constrained.

1.2 Research Gaps

Most of the research surveyed have proposed models and frameworks that have majorly focussed on conceptual levels that are more theoretical. Further investigation and research are required to tackle among others the following key issues:

a) Development of process models, methodologies and tools that are practical

Although sound principles have been applied in the proposed models and frameworks to tackle the complex challenges of IoT forensics, there is still exists a need to conduct robust experiments that can be validated scientifically. Any new methodologies, techniques, approaches, and tools developed must also undergo a scientific validation.

b) Smart analysis and presentation of evidence

Due to the huge data generated by IoT devices (which can be referred to as 'Big IoT Data'), it is important that the research community finds a way to create techniques that are smart to analyse the data. This data has varied data formats and is generated from heterogenous devices that may make it difficult to analyse and produce reports that are admissible in a court of law.

c) Provision of forensic readiness

The production of IoT equipment and provision of IoT services that are readily adaptable and integrated into the current digital processes is still a challenge in digital forensics investigations. Even though measures have been taken to address security features in IoT, issues related to forensics readiness for IoT systems still remain clouded (Bajramovic *et al.*, 2016).

d) Mitigating the privacy risks

Privacy is a contentious issue in relation to investigation processes that involve personal and protected data as stipulated under the European Union (EU) data protection laws and General Data Protection Regulations (GDPR). Full disclosure must be given to the owners. This involves letting them know that their data will be used for the investigation process and should be made aware of how the data was accessed and by whom. Those who access the data must put in place protective measures that forbids unauthorised access, any form of manipulation and loss.

e) Solutions to resolve legal issues

Evidence admissibility in a key issue in digital forensics, however, many of the models discussed in this survey have not addressed the legal aspects related to how evidence is acquired. The challenges relating to cross-border jurisdictions are imminent in cloud forensics which is a huge part of IoT systems. There needs to be propositions for solutions for legal challenges as IoT relies heavily on the cloud both for application services and architectural structure.

f) The need for digital warrants

As evidenced by research done by Oriwoh *et al.* (2013) on the Next Big Thing (NBT) and Harbawi and Varol (2017) on Last on Site (LoS) algorithm models, it is difficult to determine the scope of the investigation. This is because, potentially new evidence sources are likely to be discovered during the investigation process. With the challenges related to limited visibility and high volatility of the data exposing it to manipulation and compromise, it calls for the need for mechanisms that are practical. This can be resolved by the implementation of digital warrants to enable successful retrieval of evidence from newly discovered sources.

1.3 Research Motivation

Due to its ever-changing and dynamic nature, coupled with the complexity it introduces to digital forensics investigations, the IoT phenomenon has attracted significant attention from scholars and researchers. It therefore becomes increasingly important to focus on IoT forensic research. This field is driven by the growing adoption of IoT devices and the rapid integration of cloud-based technologies, which present new challenges and opportunities for digital forensics.

In recent years, several trends have emerged that further emphasise the need for research in the application of machine learning approaches to IoT forensics. First, the exponential growth of IoT devices has led to an unprecedented volume of digital data generated from various sources. This data, often characterised by its variety, velocity, and volume (the three Vs of big data), poses significant challenges for traditional forensic techniques. Machine learning algorithms offer the potential to analyse and extract valuable insights from this vast amount of IoT-generated data, enabling more efficient and effective forensic investigations (Sarker, 2021; Dunsin *et al.*, 2023).

Secondly, the convergence of IoT and cloud computing has introduced complex data flows and distributed storage models. This trend presents unique digital forensic challenges, as investigators must consider the distributed nature of data and the potential fragmentation of evidence across multiple cloud providers and IoT devices. Machine learning techniques can aid in reconstructing the digital trails left by IoT devices, analysing fragmented data, and identifying patterns and anomalies that might be crucial for investigative purposes (Diro *et al.*, 2021).

Furthermore, the evolving legal landscape adds another layer of complexity to IoT forensics. Courts and legal systems are grappling with the admissibility and reliability of digital evidence obtained from IoT environments. Traditional digital forensics investigation methods may not adequately address these challenges. Therefore, exploring the application of machine learning approaches that adhere to digital evidence admissibility and validity criteria becomes imperative.

By undertaking research in this domain, researchers can contribute to the development of innovative techniques that address the aforementioned trends and challenges. This includes exploring machine learning algorithms for data analysis, developing methodologies for handling distributed and fragmented evidence, and proposing frameworks that ensure the admissibility and validity of digital evidence in court. Such research has the potential to shape the future of IoT forensics and equip investigators with the necessary tools and techniques to effectively tackle complex investigations in IoT environments.

1.4 Aim

The aim of this research is to propose a novel Internet of Things forensic framework that addresses the legal and technical challenges of IoT forensic processes to be validated by a Machine Learning technique that aids in the examination and analysis of forensic data collected from a smart home environment.

1.5 Objectives

The objectives of this research are as follows:

1. To investigate through a systematic literature review, the current state of Internet of Things digital forensics methodologies, models, and frameworks.
2. To study the key factors (legal and technical) affecting IoT forensics and recommend measures for standardisation of rules to aid the digital forensics investigation process.
3. To review the automated processes in digital forensics and explore the application of Machine Learning concepts to aid in the analysis of IoT forensic evidence.
4. To propose, design, develop, implement, and test a new IoT forensic framework and validate it through a novel Machine Learning approach detailing possible solutions to address legal and technical challenges in IoT forensic processes.
5. To critically evaluate the proposed IoT forensic framework and the Machine Learning approach deployed.

1.6 Research Questions

To achieve these objectives, the following are the research questions:

1. ***Research Question***: What is the current state of IoT digital forensics methodologies, models, and frameworks, and how can they be improved to address the legal and technical challenges in the field?
2. ***Research Question:*** How can standardisation of rules be achieved to mitigate legal and technical challenges in IoT digital forensics, and what role can Machine Learning play in enhancing the investigation process?
3. ***Research Question:*** What is the effectiveness of the proposed IoT forensic framework and the proposed approach of integration of the Machine Learning technique in addressing legal and technical challenges, and how does it compare to existing methodologies?

1.7 Contributions to Knowledge

The following are the primary and secondary contributions to knowledge by this research:

Primary Contributions

1. A novel IoT forensic framework has been proposed, addressing legal and technical challenges in IoT forensic processes.
2. Validation of the proposed framework by selecting acceptable Machine Learning technique/algorithm for analysing IoT forensic data.

Secondary Contributions

3. Systematic Literature Review of IoT forensics and a review of the current legal and technical challenges of IoT forensics.
4. Generation of new IoT forensic datasets representing a simulated smart home environment. These datasets are made public for future research projects.

1.8 Research Methodology

According to Edgar and Manz (2017), the comprehension of the purpose of science and the knowledge of the cyber security domain pose challenges to conducting research. This understanding, coupled with the complexity of modern research, hinders researchers from proposing experiments. Although it is widely known that scientific methods form the foundation of scientific inquiry, their application is further complicated by the certainty associated with contemporary research. Despite the scientific method being a simplified abstraction process followed by researchers, it remains a difficult and intricate endeavour (Edgar and Manz, 2017).

While certain methods are commonly defined and yield defensible and justifiable results across various research fields, the constantly evolving technological advancements necessitate changes in these methods over time. Consequently, researchers are compelled to seek efficient and effective approaches to conduct their investigations.

1.8.1 Overview of Research Methods

Research methods are broadly categorised as quantitative and qualitative. Quantitative methods gather numerical data for statistical analysis and explanation (Edgar and Manz, 2017). Williams (2007) suggests their preference due to analysis flexibility. Quantitative designs encompass quasi-experimental, descriptive, correlation, and experimental approaches.

Qualitative methods focus on descriptive data collection and analysis, especially in studying human subjects' social and emotional aspects. Though categorisable, qualitative data lacks mathematical quantification (Edgar and Manz, 2017).

Mixed methods, per Wisdom and Creswell (2013), combine quantitative and qualitative data in a single study for comprehensive integration, avoiding strict separation.

This research employed a research approach based on the onion research methodology developed by Saunders, Thornhill and Lewis (2009) and involved a mixed methods methodology. The authors Saunders, Thornhill and Lewis (2009) associate a research approach to an onion; in this scenario, the outer most layer is the philosophy of the research. After the development of a research philosophy, a research approach is adopted and thereafter the research now goes to the third layer which is the research strategy. Upon defining the research strategy, the researcher moves to the next layer which data collection.

Figure 1.1 is the Saunders, Thornhill and Lewis (2009) onion research methodology.

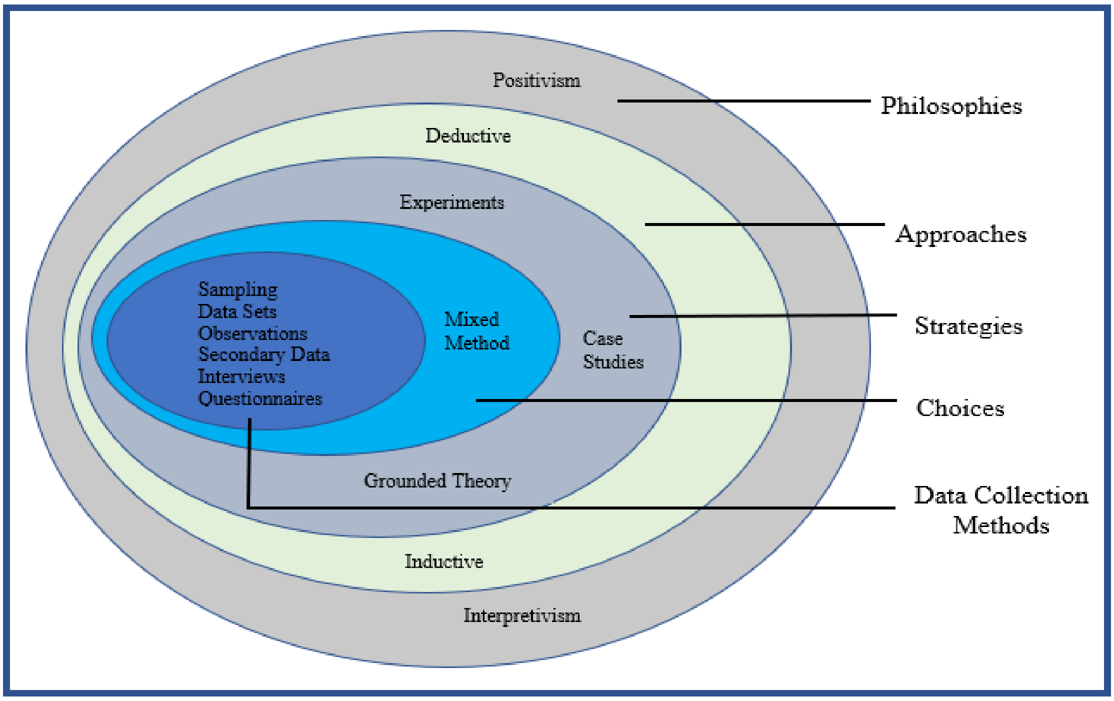


Figure 1.1 Adopted Research Methodologies based on Saunders, Thornhill and Lewis (2009) Onion Research Methodology

1.8.2 Research Philosophy

Kulatunga, Amaratunga and Haigh (2007), along with many other researchers, emphasise the importance of considering research philosophies. Easterby-Smith, Thorpe and Jackson (2002) assert that neglecting a thorough reflection and understanding of philosophical issues can have detrimental effects on the quality and value of research outcomes. By engaging in thoughtful consideration of philosophies, researchers can identify the most appropriate research methodology from the outset of their study. These research philosophies assist researchers in determining the type of evidence needed, as well as how it should be collected and analysed to address the research problem effectively. Furthermore, the adoption of research philosophies enables researchers to resolve research questions by identifying, adapting, or creating research designs that may go beyond their existing knowledge or expertise (Easterby-Smith, Thorpe and Jackson, 2002). According to Baker (2004), there are two distinct research philosophies: positivism and interpretivism.

Positivism - This is commonly recognised as a "scientific" approach that relies on well-structured and measurable methods inspired by the scientific community's practices in studying natural phenomena. In this approach, researchers maintain a certain level of detachment from the subjects they study and frequently employ observation as a means of gathering information.

Interpretivist - Interpretivist methodology is inclined towards gathering qualitative data and utilises techniques such as unstructured interviews and participant observation, which generate this type of data. Researchers following the interpretive approach acknowledge that they will both influence and be influenced by the research activity, resulting in a natural relationship between them. They believe that it is crucial to analyse how humans interpret various activities and contend that methods beyond those employed in positivism are capable of achieving this.

In contrast, this research adopted a positivist approach as its philosophical perspective, primarily due to its objectivity and reliance on logical inferences. Positivism places significant emphasis on measuring and verifying facts. Stage and Manning (2003) elaborate that this philosophy establishes an objective relationship between the researcher and the research topic, while also allowing for the incorporation of other models.

Furthermore, the authors highlight the advantage of the positivist philosophy in basing hypotheses on statistics and quantifiable measures obtained through experiments, which facilitate the manipulation of variables. Additionally, this philosophy permits the use of mixed methods, encompassing both qualitative and quantitative approaches, enabling generalisation of the secondary data (Stage and Manning, 2003).

1.8.3 Research Approach

The choice of research methods employed by a researcher depends largely on the research paradigms that they adhere to, which ultimately determine the selection of strategies for data analysis. The researcher's opinions, or epistemological perspectives, guide the rules within a specific domain or general rules, thereby reflecting their beliefs regarding what to retain, discard, or modify. These perspectives play a crucial role in the research plan, evaluation, and monitoring process. Epistemological perspectives also influence a researcher's judgment of the validity and relevance of literature materials used in their research (Hogan and Maglienti, 2001). These authors further argue that the theories in scientific and social research, particularly those aligned with a positivist philosophical perspective, tend to produce observations and conclusions that are independent of the theories themselves.

For this research, a quantitative research design was preferred due to its emphasis on creating testable hypotheses and generalisable theories (Amaratunga *et al.*, 2002) in the field of application of machine learning in digital forensics. The application of machine learning algorithms relied on these hypotheses and theories to inform the algorithm's design.

Quantitative methods possess fundamental and distinctive properties that enable the verification and application of research findings. This is particularly advantageous when investigating behaviours and mechanisms, as it allows for understanding of how different jurisdictions implement laws and the legal challenges they face, thereby facilitating the formulation of formalised and standardised approaches to aid in IoT forensics.

Another benefit of the positivist approach, as highlighted by Amaratunga *et al.* (2002), is its capacity for comparison and replication, which helps determine the reliability and validity of hypothesis verification in a study.

In this research, an inductive approach was adopted. As Bell and Bryman (2007) point out, an inductive approach aligns with the common understanding of the relationship between theory and research, where results are derived through logical reasoning.

1.8.4 Research Strategy

The grounded theory was based on the inductive approach. Cohen, Glaser and Strauss (2017) explain that in the grounded theory, the research is started with an open mind without any preconceived knowledge of what the results will look like. The resulting data forms a new theory which is placed in context through experiments.

The grounded theory strategy was utilised in this research to thoroughly review existing literature and identify any gaps, leading to the generation of new ideas that address the research questions and objectives effectively.

This research adopted a case study strategy approach to provide practical and viable solutions to the research questions posed, ultimately accomplishing its overall aim.

1.8.5 Simulation

Edgar and Manz (2017) acknowledge the inherent complexity of the cyber space, making it impractical to mathematically model every aspect, especially when considering human behaviour. To address this challenge, the authors propose the use of computer simulations to explore system interactions, theoretical limits, and component performance.

Simulation, as defined by Edgar and Manz (2017), involves the application of computer processes to replicate cyber or physical processes, generating similar responses and outputs. It effectively mimics the behaviour of real systems. By setting up complex models, simulations enable the investigation of various parameters, allowing for extensive exploration beyond manual computations. Simulation is considered a valuable tool for empirical research because it can serve as a proxy for generating data from real systems. Simulating a system offers robust control and enables the rapid exploration of multiple scenarios (Edgar and Manz, 2017).

Simulations can also generate hypotheses for experimental purposes. By instantiating and simulating theoretical models, the resulting output can be treated as a hypothesis. As emphasised by Edgar and Manz (2017), simulation methodology is particularly useful in research involving Machine Learning. It facilitates the exploration of boundaries and constraints within theoretical models and provides tools for predicting probabilities based on mathematical models. Empirical simulation and simulation for decision support exemplify how simulation fits into this research, particularly in conjunction with Machine Learning.

Table 1.1 describes how the research objectives were carried out in relation to the methodological approach employed.

1.8.6 Data Collection and Analysis

The data collection methods were through observations, analysis of secondary data, sampling, and datasets.

Simulations of a smart home environment populated with relevant data from IoT devices from a simulator were used in different scenarios depicting the real-world phenomena under different case scenarios.

Triangulations as part of the research were used to analyse the information and data gathered, So, to satisfy the objectives of the research both qualitative and quantitative methods were used.

Critical analysis was used in the critical evaluation of the new framework and the algorithm proposed through critical analysis methods.

Objective	Methodological Approach	Description
To investigate through a systematic literature review, the current state of Internet of Things digital forensics methodologies, models, and frameworks.	Systematic Literature Review Rapid Review, etc.	Systematic Literature Review (SLR) Rapid Review (RR) is a streamline approach for producing evidence - typically for informing emergent decisions
To study the key factors (legal and technical) affecting IoT forensics and recommend measures for standardisation of rules to aid the digital forensics investigation process.	Systematic Literature Review Rapid Review, etc.	SLR and RR Identify gap filling measures
To review the automated processes in digital forensics and explore the application of Machine Learning concepts to aid in the analysis of IoT forensic evidence.	Systematic Literature Review Rapid Review Experiments Gap analysis	SLR and RR Identify gap filling measures. Laboratory experiments by simulation of datasets
To propose, design, develop, implement, and test a new IoT forensic framework and validate it through a novel Machine Learning approach detailing possible solutions to address legal and technical challenges in IoT forensic processes.	Experiments Simulations Grounded Theory Case studies Probabilistic models and probabilistic graphical models	SLR and RR Laboratory experiments by simulation of datasets Hypothesis of theories
To critically evaluate the proposed IoT forensic framework and the Machine Learning approach deployed.	SWOT (Strengths, Weaknesses, Opportunities, and Threats) analysis Comparative Analysis Statistical Analysis	Rigorous review of the proposed framework against available state-of-the-art ML model coupled with SWOT analysis.

Table 1.1 Table showing Objectives and Research Methodology to be used.

1.8.7 Critique of the Research Methodology

This research heavily relied on the mixed methods approach, but it has certain limitations. As noted by Wisdom and Creswell (2013) implementing mixed methods can be challenging. This challenge was evident in this research, which consisted of both practical and impractical aspects. The impractical aspect involved relying primarily on qualitative data for legal aspects and the development of the IoT Forensic Framework, while the practical aspect involves implementing machine learning concepts and relying on quantitative data.

Despite these limitations, the authors (Wisdom and Creswell, 2013) hold mixed methods in high regard, and this research aligned with their belief that mixed methods are ideal for investigating complex scenarios and gaining a better understanding of the research results.

In a different context, Edgar and Manz (2017) acknowledge the value of simulations for researching in the cyber space domain, as they closely mimic real systems. However, they cautioned that the reliability of simulation results hinges on the quality of the models employed. If the models lack sufficient empirical support, the application of the results may raise doubts. The authors also emphasise the importance of properly validating the models used, suggesting that suitable models should be validated through data collected from real systems via study or experimentation.

In conclusion, the research methodology has presented an overview of different research methods used in cyber security. This research follows a mixed methods approach and onion research methodology to effectively address the research questions while considering the advantages and limitations of each method. The methodology employed in this thesis includes quantitative analysis, grounded theory, case study strategy, simulation, and critical analysis of data. These methods aim to effectively address the research questions and gain valuable insights in the field of study.

1.9 Structure of the Thesis

This thesis is structured as follows:

Chapter 1 – Introduction

This chapter provides the background of the research area, brief of the research gaps, research motivations, the aim and objectives of the research, research questions, contributions to knowledge and concludes with an overview of the research methodology.

Chapter 2 – Literature Review

This chapter explores the existing literature in the field of IoT forensics. The literature review aims to unearth the current state-of-state in the field of IoT forensics by exploring the factors affecting IoT forensics, the legal and technical challenges, and how to overcome them. Through a systematic literature review, the findings identify open challenges and opportunities for IoT forensics to which this research contributes to.

Chapter 3 – Automated Processes in Digital Forensics

This chapter reviews the increasing role of automated processes in Digital Forensics. It highlights the recommendations made by researchers in the application of automation in Digital Forensics. The chapter also deeply explores anomaly detection and reviews the state-of-the-art anomaly detection algorithms.

Chapter 4 – Proposed Theoretical IoT Forensic Framework

This chapter delves into the proposed theoretical framework for conducting IoT forensic investigations. The framework is presented in a step-by-step guide comprising of four distinctive phases of the IoT forensic investigation process.

Chapter 5 – Simulation and Dataset Generation

This chapter explores forensic analysis techniques in smart homes, discussing the challenges and opportunities unique to these environments. It also reviews smart home simulation tools and justifies the choice of Open Smart Home Simulator (OpenSHS). Building on the theoretical framework from Chapter 4, this chapter develops forensic scenarios for simulation and dataset generation aligned with the proposed IoT Forensic Framework.

Chapter 6 – The Application of HI-SDR in Anomaly Detection

This chapter builds on the smart home datasets generated in Chapter 5 and discusses Sparse Distributed Representation (SDR) generation, highlighting its importance for forensic analysis. It explores properties critical for robust forensic analysis in smart home datasets and examines the role of SDR encoders. The chapter introduces a High-Indexed SDR (HI-SDR) encoder chosen for the research experiments and proposes an approach based on this encoder. It concludes by discussing the practical forensic application of the proposed approach and its effectiveness on dataset representation for better anomaly detection.

Chapter 7 – Test and Evaluation

This chapter evaluates the performance of the proposed HI-SDR approach for anomaly detection in the generated smart home dataset. The evaluation uses performance metrics (accuracy, precision, recall, and F1 measure) to gain deeper insights into the approach's capabilities. It then compares the HI-SDR approach's performance to the state-of-the-art machine learning model (Isolation Forest) and One Class Support Vector Machines – OCSVM) chosen for the experiments to determine the most effective approach for anomaly detection in this context.

Chapter 8 – Summary, Conclusion, and Future Work

This chapter concludes the research by examining the research process and deriving significant findings regarding the research contributions. It also acknowledges the limitations of the study and offers suggestions for future research directions.

Figure 1.2 below illustrates the overall overview of the research structure.

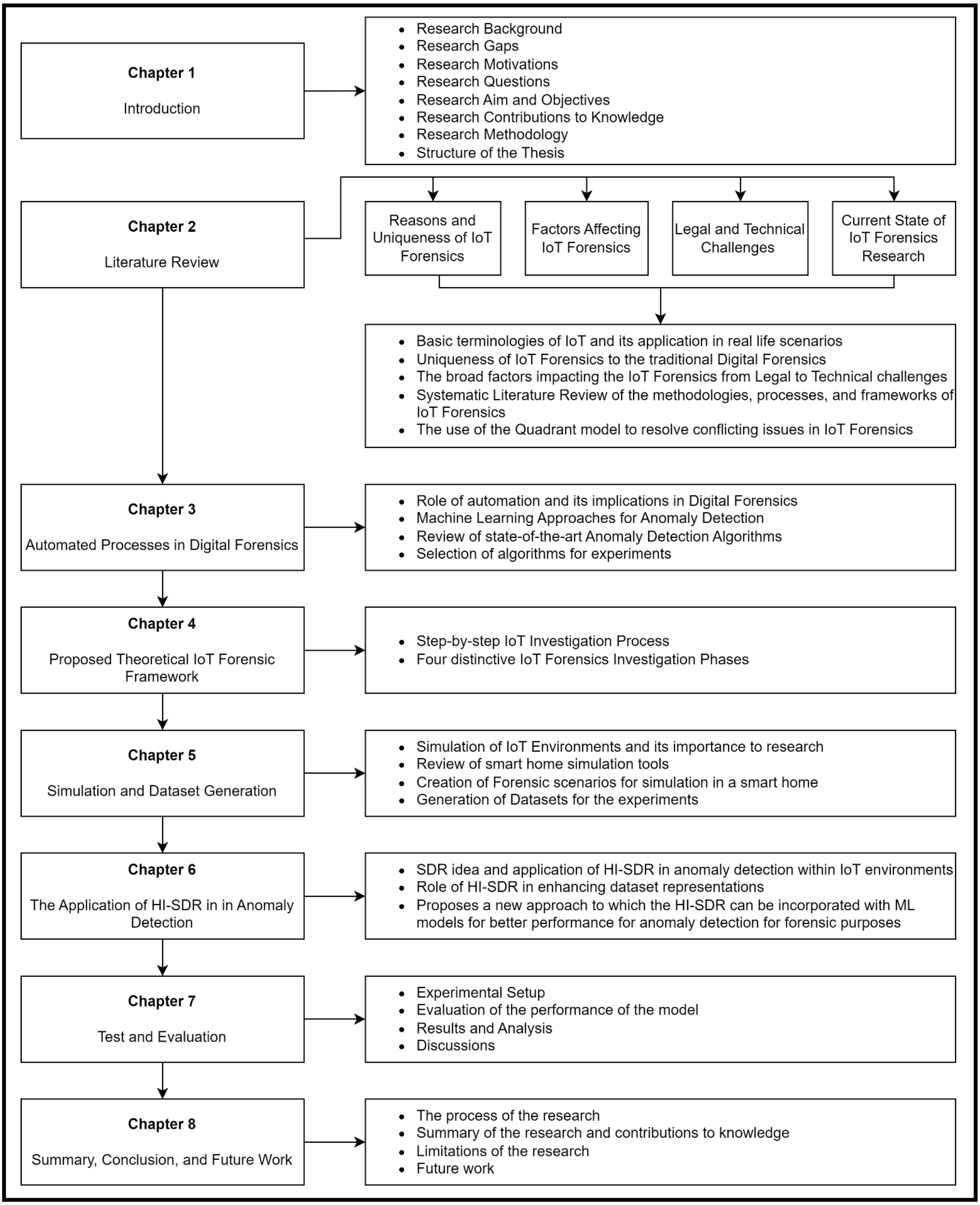

Chapter 1
Introduction
Research Background
Research Gaps
Research Motivations
Research Questions
Research Aim and Objectives
Research Contributions to Knowledge
Research Methodology
Structure of the Thesis
Chapter 2
Literature Review
Reasons and Uniqueness of IoT Forensics
Factors Affecting IoT Forensics
Legal and Technical Challenges
Current State of IoT Forensics Research
Basic terminologies of IoT and its application in real life scenarios
Uniqueness of IoT Forensics to the traditional Digital Forensics
The broad factors impacting the IoT Forensics from Legal to Technical challenges
Systematic Literature Review of the methodologies, processes, and frameworks of IoT Forensics
The use of the Quadrant model to resolve conflicting issues in IoT Forensics
Chapter 3
Automated Processes in Digital Forensics
Role of automation and its implications in Digital Forensics
Machine Learning Approaches for Anomaly Detection
Review of state-of-the-art Anomaly Detection Algorithms
Selection of algorithms for experiments
Chapter 4
Proposed Theoretical IoT Forensic Framework
Step-by-step IoT Investigation Process
Four distinctive IoT Forensics Investigation Phases
Chapter 5
Simulation and Dataset Generation
Simulation of IoT Environments and its importance to research
Review of smart home simulation tools
Creation of Forensic scenarios for simulation in a smart home
Generation of Datasets for the experiments
Chapter 6
The Application of HI-SDR in in Anomaly Detection
SDR idea and application of HI-SDR in anomaly detection within IoT environments
Role of HI-SDR in enhancing dataset representations
Proposes a new approach to which the HI-SDR can be incorporated with ML models for better performance for anomaly detection for forensic purposes
Chapter 7
Test and Evaluation
Experimental Setup
Evaluation of the performance of the model
Results and Analysis
Discussions
Chapter 8
Summary, Conclusion, and Future Work
The process of the research
Summary of the research and contributions to knowledge
Limitations of the research
Future work

Figure 1.2 Overview of the Thesis Structure

CHAPTER 2. LITERATURE REVIEW

This chapter thoroughly reviews the pertinent literature that relates to the motivations and contributions presented in Chapter 1 of this thesis. The purpose of the literature review is to ascertain the existing body of scholarly work that is relevant to this study and to identify areas where gaps and unanswered questions exist, thereby providing a contextual framework for the current research. As such, it offers background information, contextualises the motivations outlined in Chapter 1, and evaluates the extent to which the existing literature addresses these specific areas in each case.

This chapter explores the current legal and technical challenges of IoT forensics by emphasising the uniqueness of IoT forensics, uses a quadrant model to expose conflicting scenarios in IoT forensics process, and finally, recommends the need for application of intelligent systems like machine learning techniques to semi-automate the IoT forensic process for profiling and surveillance.

A systematic literature review is conducted to bring to light the existing research gaps in IoT forensics.

2.1 Introduction

The continued growth of IoT devices has enabled the sharing of information within people and between the devices themselves. The direct communication between these devices is facilitated over the internet by the Application Programming Interface (API) and is controlled by intelligent devices of the cloud servers that enhance smartness to low-computing resource incapacitated IoT devices (Yaqoob *et al.*, 2019).

There are indeed many beneficial aspects brought about by IoT applications more so in the areas of transportation (automotive), retail, health care, engineering, construction, smart cities and many others (Kim, Ramos and Mohammed, 2017). According to a report by Cisco (Cisco, 2016) on the state of IoT, by the year 2030, it is expected that over 500 billion devices will be connected by the internet. The report also states that the IoT business is estimated to have a revenue turnover of around $14.4 trillion by the year 2022. This revelation indeed shows that the human population has already been surpassed by the number of connected IoT devices.

IoT devices have limited computing capabilities in relation to processing and storage of data, due to this, Al-Masri, Bai and Li (2018) note that IoT environments make extensive use of the cloud computing services. The authors further depict that as result of the continued growth of customers for cloud-based services, it is evident that there is growing over dependency on cloud-storage media. This translates to the need for having digital forensics tools that can handle large volumes of data enabling the extraction of data that could be potential evidence. There is also a need for training forensic investigators on how to collect evidence from the cloud. As is always the case in most forensic processes, it can be a time-consuming exercise as forensic tools may take a lot of time to analyse huge amounts of data. This results in a slow forensic examination process thereby complicating and making it difficult, more so in the collection of data from the cloud which could be stored in distributed locations.

Despite this positive outlook of the emergence of IoT technologies, it brings with it various security attacks and threats as noted by (HaddadPajouh *et al.*, 2018). These threats could be in form of attacks from viruses, illegal surveillance, Denial of Service (DOS) attacks among other many threats and attacks. Digital forensics experts are often called in to investigate these incidences.

It is unfortunate that in the design and development of IoT devices, not much attention is paid to the security, due to cost implications, therefore leaving them exposed to susceptible threats. This gives room for hackers who exploit IoT devices' vulnerabilities and carry out illegal activities that cripple the cyberspace.

IoT forensic process brings with it unique and complex challenges. This is because digital investigators are required to create new investigative processes that are specific to IoT by drawing upon techniques and methods used in acquiring evidence from other established areas of digital forensics. The evidence in IoT devices is different from the traditional digital device (computers and mobile phones), this is because data from IoT devices can be in vendor specific formats that deviate from the normal electronic documents or file system formats.

It is evident as noted by Kebande and Ray (2016), that the IoT systems' complexity together with the inadequate or even no unified standards hinder the process of digital forensics by preventing the acquisition of valuable digital evidence by Law Enforcement Agencies (LEA) from IoT based forensic cases. The authors also concur that the available traditional methods, tools, and standards for digital forensics are unable to handle the highly heterogeneous IoT infrastructure that is distributed across the globe.

As highlighted by Al-Masri, Bai and Li (2018), even though research in digital forensics in cloud forensics is essential, much of the current research has focused more on the challenges encountered when carrying out digital forensic investigations in the cloud. There is minimal research that has proposed solutions that can be used to work around these challenges through practical models for digital forensics in the cloud. A lot of the available research focuses more on data storage, access control and the security of data in the cloud. The recent emergence of IoT which extensively uses the cloud computing platforms, has made it necessary to find solutions that are able to aid the forensic process.

As observed by Yaqoob *et al.* (2019), many surveys have been conducted in the digital forensics interdisciplinary domains such as mobile phones, smart cities, cloud computing, wireless networks and smart transport systems. However, these studies do not conclusively tackle the IoT challenges. The authors proceed and state that many studies have been conducted on IoT security rather than IoT forensics.

Even though many conceptual models and frameworks have been developed to try and solve the complex challenging characteristics of IoT forensics process, there still exist many unresolved challenges such as standardisation, legal jurisdictions, and the forensic analysis of big IoT data (Chernyshev *et al.*, 2018). This is further highlighted by Harichandran *et al.* (2016) who note that most research that relates to the digital forensic investigative process in IoT is more theoretical than practical.

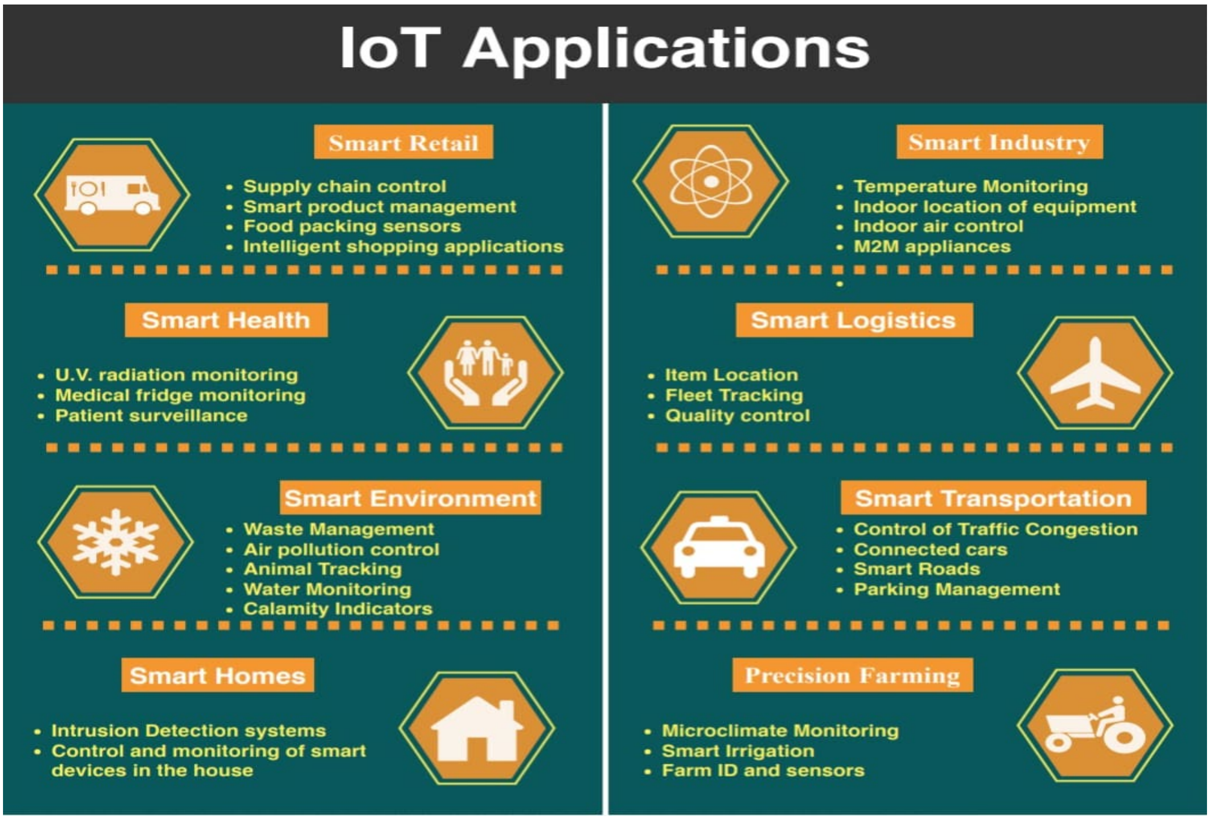
Generally, as also observed by Harbawi and Varol (2017), the forensic process of IoT is still in its early stages, there are few and limited researches that have been conducted. The conducted research, however, lack in-depth analysis and experimental results which could be as a result of unavailable testing data from IoT devices and/or limited IoT environments. On the other hand, the few studies with experimental tested models are specialised to specific scenarios which means that they cannot be used for general wholesome IoT forensic investigation processes.

Additionally, as noted by Babun *et al.* (2018), that not much room for forensic analysis is provided for by the currently developed IoT solutions. The authors further claim that due to the limited computing resource capabilities for many IoT devices coupled with the unique cloud-based infrastructure makes it even difficult to store data in the devices for forensic purposes. Most popular IoT programming platforms like Samsung SmartThings, OpenHAB and others do not provide any means to have access and indefinitely store data in the cloud.

2.2 Internet of Things Forensics

IoT being an emerging technology allows small devices (things) to perform tasks as smart objects. The interconnection between these devices (things) is facilitated by different network media types. The communication between the devices generally makes applicable decisions through the sensor data read.

IoT technology can be applied among various application areas for example in home automation, wearable technology, smart environment, smart retail, smart industry, transportation, health, and Agricultural farming. This is best illustrated in Figure 2.1:



IoT Applications

Smart Retail
- Supply chain control
- Smart product management
- Food packing sensors
- Intelligent shopping applications

Smart Industry
- Temperature Monitoring
- Indoor location of equipment
- Indoor air control
- M2M appliances

Smart Health
- U.V. radiation monitoring
- Medical fridge monitoring
- Patient surveillance

Smart Logistics
- Item Location
- Fleet Tracking
- Quality control

Smart Environment
- Waste Management
- Air pollution control
- Animal Tracking
- Water Monitoring
- Calamity Indicators

Smart Transportation
- Control of Traffic Congestion
- Connected cars
- Smart Roads
- Parking Management

Smart Homes
- Intrusion Detection systems
- Control and monitoring of smart devices in the house

Precision Farming
- Microclimate Monitoring
- Smart Irrigation
- Farm ID and sensors

Figure 2.1 IoT Application

The word forensics can loosely be referred to as the application of science and technology in an investigation process for the purpose of establishing facts in a criminal or a civil litigation.

Digital forensics is a discipline that combines the basics of computer science and laws where the collected digital data (evidence) is analysed and presented as admissible in a court of law for prosecution purposes.

Forensic computing is a process that involves the identification, preservation, acquisition of data of potential evidence and analysis of the data to produce a report to be presented in a court of law in a way that follows the laid down procedures and acceptable laws in a particular jurisdiction.

IoT forensics can therefore be termed as a process of applying the process of digital forensics in a setup that contains IoT devices.

The authors Zawoad and Hasan (2015) have defined IoT forensics by combining three digital forensics levels, namely: device, network, and cloud level forensics. The device level forensics involves collection of local memory data from IoT devices. The network level forensics is where network logs are extracted and analysed. Finally, the cloud level forensics involves analysing the data generated and stored by IoT devices to the cloud services. The cloud services serve a huge role in IoT operations. This is due to IoT devices having low storage and computational capacity thereby relying heavily on the cloud services which offer benefits like convenience, large capacity, scalability, and on-demand accessibility.

Digital forensics investigation process has been vibrant recently due to the emergence of IoT technology which is now seen as a big threat to information security. The large volumes of data generated by IoT devices and in turn reshared between the devices contains a huge potential of evidential data due to the large number and variety of IoT devices that are spread within a wider application area.

As noted by Chernyshev *et al.* (2018), the digital evidence retrieved from an IoT setup can be useful because the evidence can be used by parties involved to support or contest any hypothesis claimed in the investigation process. This can be referenced to a New York Times report by Hauser (2017) where a murder case was determined by data from a wearable device (Fitbit). The complexity around the extraction of data from IoT environments is a major setback in the ability of producing evidence that is legally admissible in a court of law (Kebande and Ray, 2016).

2.3 What is unique about IoT Forensics?

Forensics of IoT is still in its infancy as noted by Kebande and Ray (2016). The authors highlight that even though researchers have been attracted to this field, current Digital Forensics tools and techniques are not well equipped to handle the heterogeneous and distributed nature of the IoT setup. This has posed a challenge to the digital investigators and law enforcement agencies in the investigation process that can gather, examine, and analyse potential evidence from IoT platforms and present evidence that is admissible in a court of law.

Generally, conventional digital forensics scenarios include tangible devices such as Personal Computers (PC), mobile phones and tablets that contain data of potential evidence. In an IoT setup, there is a significant change in the sources of evidence as there is increased number and types of devices of interest that are intangible due to different location sites, and the distributed nature of IoT, where the potential evidence may be stored on the cloud.

It is argued by Induruwa (2011) that the cloud, due to its convenience, scalability and on demand accessibility plays a fundamental role in an IoT forensics. The author states that with the inclusion of the cloud, the issues related to redistribution in different locations and multi-tenancy make IoT forensics different.

It is observed by Oriwoh *et al.* (2013) that in traditional digital forensics, the investigators use accepted methodologies that follow the standards, guidelines and principles provided by widely recognised bodies like; Association of Chief Police Officers (ACPO) and Scientific Working Group on Digital Evidence (SWGDE). The authors note that in an IoT setup, these methodologies may be limited due to the increased scope of IoT crimes. Recently, Wachter (2018) emphasised on the privacy rights enshrined in the EU General Data Protection Regulation (GDPR) which make IoT forensics further interesting. This is because IoT devices and their (IoT) services have a tendency of collecting, sharing, and storing huge volumes of data that contains personal data that is of varied types. However, it can be noted that the personal data generated from IoT devices is unstructured and could be spoofed which makes the forensic process very challenging.

In a forensic investigation, search and seizure is a very important step. Harbawi and Varol (2017) have argued that whereas search and seizure can be easily achieved in a traditional digital forensics investigation, it becomes a challenge in IoT forensics and IoT devices are configured to work passively and autonomously. Additionally, D'Orazio, Choo and Yang (2017) note that even though the identification of an IoT device can be done, there may be no well recognised methods or tools that can help a forensically sound process of collecting residual evidences from the IoT device.

Moreover, Conti *et al.* (2018) observe that even though there could be a few methods that could be used to create forensic images of IoT devices, these methods do not adhere to the ethical considerations when evidence is being collected from the devices that are run in an environment which has multi-tenancy. These authors continue and state that while collected data could be preserved using the current techniques like hashing, the challenge in IoT setup comes in the preservation of the digital forensic crime scene. Different IoT nodes could still have real time and autonomous communication thereby making it hard to fully locate the crime scene that has been compromised.

Traditional digital forensics techniques could be used to acquire and analyse some IoT devices, there still exists a challenge of these devices possessing vendor specific software, different file systems structures and diversity of communication protocols that add complexity (Harichandran, et al., 2016).

Another challenge mentioned by Dehghantanha and Franke (2014), is that many IoT devices do not store metadata that includes temporal information such as timestamps.

A summary of the characteristics that make IoT forensics different from other traditional digital forensics are as follows:

- More challenging due to the immense growth of IoT devices and their distributed nature,
- The IoT devices are heterogeneous in nature and require specialised tools to retrieve data,
- Existing IoT devices could be resource constrained,
- The data collected is huge and diversified, this brings complication in the forensic process,

The proprietary protocols, laws and regulations for implementation are widely spread and not standardised.

2.4 Systematic Literature Review of IoT Forensic

The Systematic Literature Review adapted the search methodology guidelines proposed by (Kitchenham, 2007).

The aim of this review was to analyse the current state of research in relation to IoT forensics, expose the key challenging factors, explore the practicality of the reviewed literature, and discuss open issues and requirements for future research directions.

The research questions for this review were:

i) What are the key factors affecting IoT Forensics?

ii) What are the current IoT forensic methodologies, models, and frameworks?

iii) How practical and realistic are these methodologies, models, and frameworks?

iv) What are the open challenges and requirements for future research directions?

The strategy used to find relevant literature are presented in the search protocol that answers the research questions.

The online databases used in this review were: IEEE Xplore, Springer Link, ACM Digital Library, Wiley Online Library, Science Direct, and Google Scholar.

The usefulness of a search string is to capture the keywords in the research questions to find the desired results. To connect the keywords, the search employed Boolean operators (AND and OR). To attain exact words, the quotation marks were used, and the search string was:

(“Digital forensic framework” OR “Digital forensic methodology” OR “Digital forensic model” OR “Digital forensic challenges”) AND (“IoT” OR “Internet of Things” OR “Smart homes” OR “Cloud of things”)

The search string was executed on the online databases, from the results, keywords from the titles were read so that irrelevant papers were filtered out. For further refining of the results, the search applied inclusion and exclusion criteria to analyse the abstracts and full text reading.

The exclusion phase was done by excluding papers that are not peer reviewed, and papers of low quality and without any scientific basis.

The inclusion criteria used for this SLR was based on online published papers from 2011 to 2018 and only studies that are in digital forensics field and specifically, IoT forensics. This SLR was subsequently published (Lutta *et al.*, 2021).

For this thesis, however, the literature search has been expanded to include published peer reviewed work up to the year 2023.

Further exclusion exercise to refine the results was based on non-English papers.

Figure 2.2 illustrates a flowchart that summarises the search methodology employed.

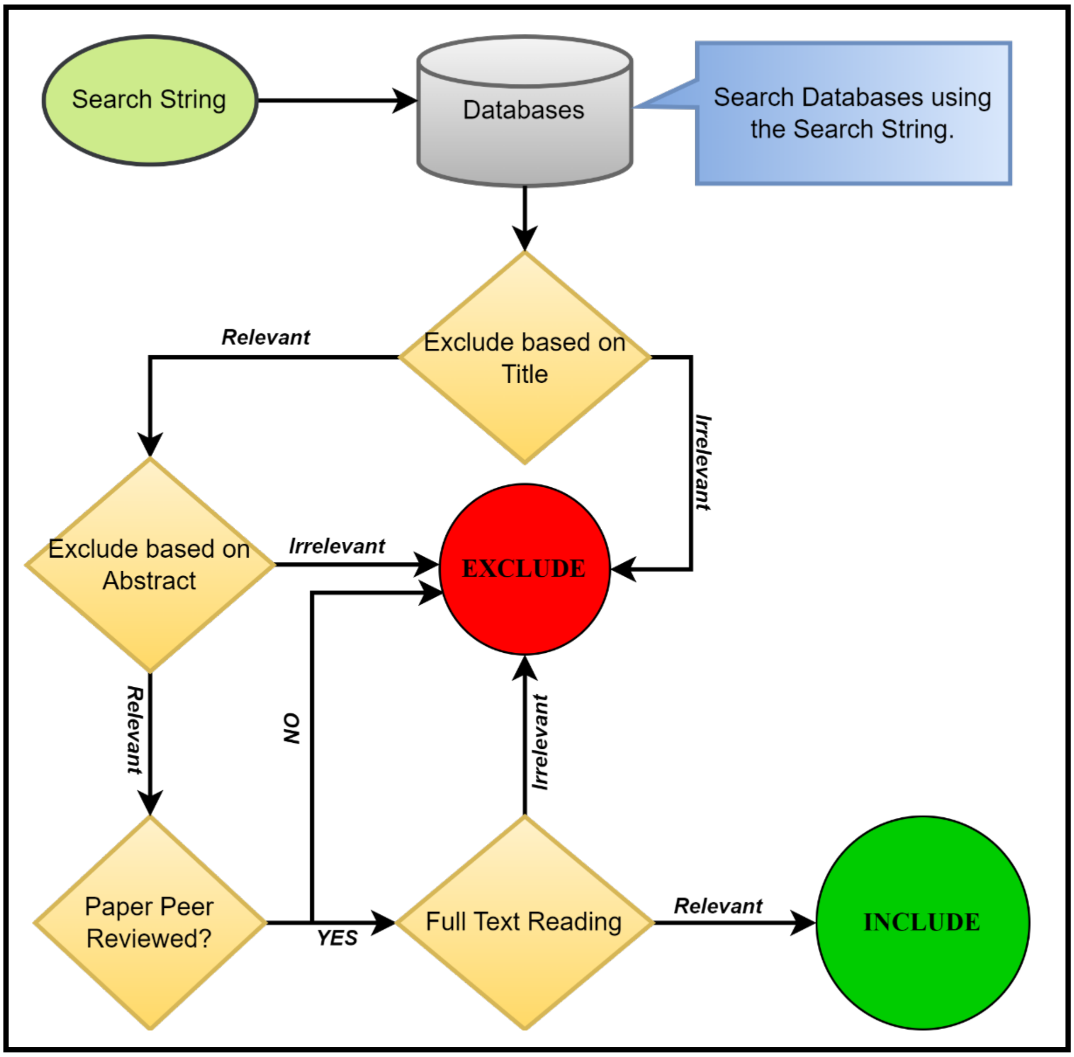


Figure 2.2 Search Method Flowchart

2.5 Factors Affecting IoT Forensics

The complex and unique challenges brought about by IoT environments in relation to forensic investigative process have attracted recent advancements in the research. These efforts are however still in their early stages of development and majorly focusing on the theoretical process models based on hypothetical case studies (Harbawi and Varol, 2017).

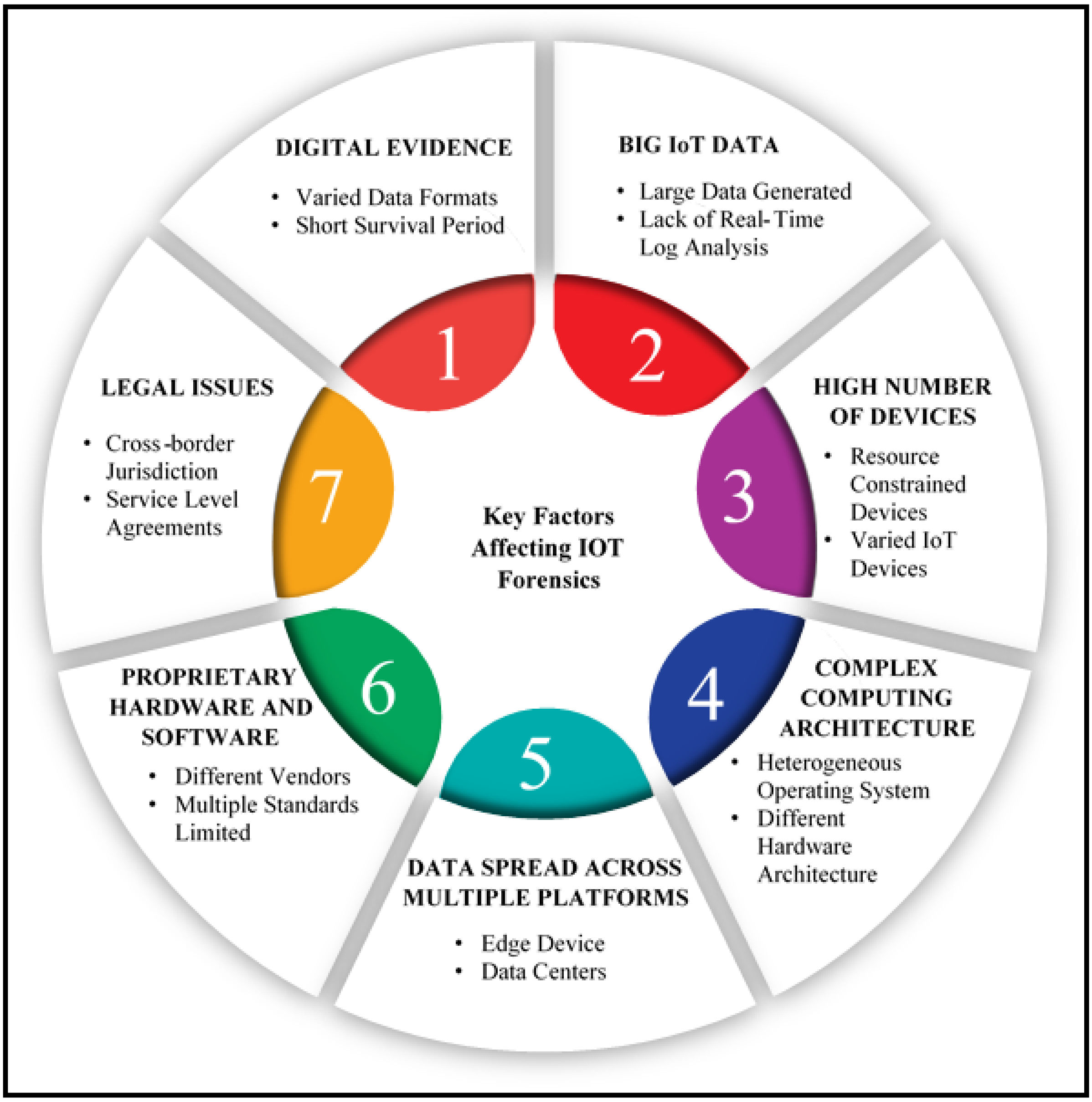

DIGITAL EVIDENCE
• Varied Data Formats
• Short Survival Period
BIG IoT DATA
• Large Data Generated
• Lack of Real-Time Log Analysis
HIGH NUMBER OF DEVICES
• Resource Constrained Devices
• Varied IoT Devices
COMPLEX COMPUTING ARCHITECTURE
• Heterogeneous Operating System
• Different Hardware Architecture
DATA SPREAD ACROSS MULTIPLE PLATFORMS
• Edge Device
• Data Centers
PROPRIETARY HARDWARE AND SOFTWARE
• Different Vendors
• Multiple Standards Limited
LEGAL ISSUES
• Cross-border Jurisdiction
• Service Level Agreements
1
2
3
4
5
6
7
Key Factors Affecting IOT Forensics

Figure 2.3 Factors affecting IoT Forensics

Key IoT challenges that pose difficulties in digital forensics investigations are established by (Hegarty, Lamb and Attwood, 2014). The authors identify fundamental areas that researchers should focus to provide solutions. The paper takes a view of the traditional digital forensics process (identification, preservation, analysis, and presentation) and relates it to how it can fit into IoT forensic, however, the authors did not perform any practical analysis for implementation.

Figure 2.3 depicts the challenges:

2.5.1 Digital Evidence

The authors Yaqoob *et al.* (2019) lament that the key challenges exhibited by the huge data to the investigators are the varied data formats and the limited solutions for real-time log analysis. The short survival period and the limited visibility of the evidence can also be viewed as challenges to the investigation process more so in the circumstances where traditional digital forensics processes are applied in IoT forensics.

2.5.2 Big IoT Data

Due to the large data generated by IoT devices which are resource constrained and diversified across a huge spectrum. It is noted by Yaqoob *et al.* (2019) that this large quantity of data generated presents digital forensics expert the difficulty of collecting and extracting evidential data in an efficient manner.

A research by Feng and Zhao (2018) summarises the review on digital forensics trends used for Big Data and the challenges encountered in the acquisition of evidence. A Smart City project is used as a case study where IoT services collect Big Data and store it in the cloud. The authors note that one of the major challenges of the forensic process is due to the distributed nature of the cloud environment making it very difficult for the data acquisition techniques to retrieve evidence. In the case study, an example is given of a driverless car (public transport vehicle) which sends huge amounts of data to the cloud. This data is in turn used to control the operations of the car and provide local information by suggesting the best services to the customers. A scenario is created where this data is hacked into, and the car is crashed. It is depicted that it would be hard for a digital forensic investigator to again access to the data. The paper does not provide any viable practical solution to the challenges it highlights; however, it provides a summary of the challenges that have been solved and not solved under cloud services which could be beneficial to the research community.

A research by Alabdulsalam *et al.* (2018) discusses IoT forensics and brings out its uniqueness of IoT forensics to traditional forensics by highlighting the challenges encountered. In their experiments, the authors used a smartwatch as a case study and described how to acquire forensic data from an apple smartwatch. The three levels of IoT forensics (device, network, and cloud) have been emphasised. The paper describes the main challenges of IoT as; location of data, limitation of digital media due to lifespan, weak requirements signing up for cloud services, lack of security in IoT devices, device type identification, and the proprietary data formats. The limitation of the currently available forensic tools to handle IoT forensics has also been discussed, this is more so stressed by the fact that most of the IoT data is found in the cloud and not many forensic tools can collect data in the cloud due to the data volatility. In their conclusion, the authors concur that there is a need to develop an efficient generic model to handle IoT forensics.

The challenges encountered with the IoT's big data ecosystem and recent IoT applications are highlighted by Cartier *et al.* (2018). They observed that there is need for tools and libraries for better management of IoT-big data.

2.5.3 High Number of Devices Spread across the Globe

IoT forensics challenges are looked at by (Macdermott *et al.* (2018) with a view of Internet of Anything (IoA) era. The IoA is depicted by the author as an explosion of connected devices due to anything and everything online being connected. The author state that the main forensic challenge of IoT/IoA is the procedure for the acquisition of data in those connected devices.

The laws surrounding accessing data in the cloud are looked at by Walden (2013). Given that the cloud stores a huge amount of data transmitted between IoT devices, the author states that one of the major challenges that digital investigators face is the collection of data in the cloud setup. Cross-border cooperation for mutual legal assistance should be encouraged to enable acquisition of data from different territories. The same challenges were also raised by Jahankhani and Hosseinian-Far (2017).

2.5.4 Complex Computing Architecture

As highlighted by Zawoad and Hasan (2015), IoT devices have limited computing capabilities and rely heavily on cloud services for their functionalities. It therefore follows that data will be collected from the cloud infrastructures and analysed leading to a form of cloud forensics investigation.

The complex challenges in cloud forensics are highlighted by the authors Birk and Wegener (2011) who acknowledge that evidence can vary significantly when collected via Virtual Machines (VMs) from multiple cloud deployment models; the authors recommended a service-specific solution. An example is given where if evidence was to be collected from an Infrastructure as a Service (IaaS) environment, it is recommended that collection is done by use of snapshot analysis or creation of forensic images through cloning. The setback with this solution is that it is not feasible for cloud service providers to clone all of their cloud servers as this will require a lot of storage space.

A survey authored by Venčkauskas *et al.* (2015) seeks to analyse the state of cyber-crime in IoT environments. The authors discussed issues relating to how the traditional digital forensics methodologies could be integrated into IoT cases. The authors clearly indicated the types of crimes in IoT and where potential evidential data could reside in cloud environments and how to extract said data. However, the authors failed to carry out any practical example of how to implement their recommendations leaving the paper more theoretical than practical.

2.5.5 Data Spread across Multiple Platforms

Three layers (cloud/server, network and endpoints) are outlined by Rughani (2017) where potential evidence can be located. The author attempted to identify issues and challenges encountered during the acquisition of evidence from IoT environments in a crime scene. Even though one of the author’s aims was to help investigators in acquiring data from IoT crime scenes, there is no practical example to illustrate the same.

The challenges of forensic analysis encountered at the physical infrastructure on whose basis lies the operating systems of Industrial IoT (IIoT) are highlighted by Eden *et al.* (2017). A review of the available tools that can handle a forensic process of Supervisory Control and Data Acquisition (SCADA) is done resulting into a SCADA incident response model.

2.5.6 Proprietary Hardware and Software

According to Varadharajan and Bansal (2016), the data from IoT devices is heterogeneous unlike the data from traditional data devices. The authors further note that IoT data may stream at rates that are unpredictable. Additionally, the security and privacy measures employed in many IoT devices do not address issues like ownership, management, and regulations.

The forensic challenges faced by vehicular fog computing are highlighted by Huang, Lu and Choo (2017) who pointed out the difficulty in physically checking every fog node deployed in the system. The authors, however, provided countermeasures like evidence based digital forensic and traffic-based evidence approach.

2.5.7 The Legal Challenges

The prevalence of IoT enhancement through approaches that leverage big data techniques for the purposes of improving the assurance of information are surveyed by Underwood (2016). The author notes that it is expected that IoT will stress the organisational frameworks in relation to the current technical and legal spectrum. This will however be significant more so in forensics and safety audits. The nature of work for information security experts, forensic investigators and system auditors has been hugely changed by the prevalence of Big Data. It is more complicated by the emergence of IoT devices that add huge volumes and various forms of work to be performed by these experts.

A survey conducted by Hon, Millard and Singh (2016) reviewed cloud computing and Internet of Things (both combined as “cloud of things”) in relation to key legal issues emanating from European Union (EU). The wider perspective on legal and regulatory aspects of cloud of things, major challenges, and complexities in the past, present, and future are highlighted.

The following aspects are covered at length in the survey:

- Cloud of Things (CoT) concepts and challenges are explained in relation to the definition of “things”, what they do, how they communicate and the role of the clouds and their security challenges thereof.

- Legal relationships and liabilities involved in cloud of things; the establishment of different parties in CoT and their relevant roles, the contractual obligations, the ownership of the data and software intricated in CoT, and the potential sources of liability and the role played by the insurance.

- Handling of personal data in CoT; issues related to personal data in the cloud of things under the EU data protection laws and General Data Protection

Regulations (GDPR), expounding on what data is regulated, to whom the responsibility falls, the applicable laws, what rights do users have over their data, the location, and transfers of the data.

- The governance of CoT; tackles the key issues relating to identity, authenticity and trust, consumer protection, standards and the demonstration of how legal obligations can be complied.

The paper outlined the various fundamental legal considerations as presently portrayed in the cloud of things. Although nothing much has been covered in relation to cloud and IoT forensics, this research is deemed resourceful in the application of laws in the process of IoT forensics. However, the laws are only limited to EU regions.

There is no physical access of the storage facilities and that digital forensics investigators rely heavily on the Cloud Service Providers (CSP) for cooperation on the retrieval of evidence, this was highlighted by Feng and Zhao (2018). The cross-border technicalities that make it hard to establish a chain of custody as required by law have been highlighted as a challenge to IoT forensics.

A paper by Losavio *et al.* (2018) analysed IoT and smart cities in relation to the legal challenges encountered in digital forensics, privacy and security and noted that competence of digital forensics experts in matters law was a major hinderance. The authors did a comparative review of legal regimes in China, Korea, Hungary, European Union, and the United States of America, analysing how digital forensics and investigations are carried out. The GDPR of the EU has been identified as well-defined to aid the process. The authors state that the US case decisions can be a basis for analysing current legal problems paving way for future regulations. They conclude by stating that the legislation needs to be clear on issues relating to the balance between public security and individual privacy freedoms.

It is noted by Rughani (2017) that unlike in traditional digital forensics investigation where the process is well defined by the National Institute of Standards and Technology (NIST), no specific guidelines are provided for in an IoT crime scenario.

The solutions suggested by Birk and Wegener (2011) are based on different use cases like the verification of Service Level Agreement (SLA) and enforcement of compliance aspects.

2.6 The Legal Challenges of IoT Forensics in Context

The emergence of IoT era and the ever-advancing technology in nearly all the digital gadgets indicates that the digital forensics domain is reaching a tipping point. The traditional forensic tools that worked are increasingly becoming obsolete (Garfinkel, 2010). More complex reverse engineering techniques are required as forensically relevant data is being stored in proprietary file formats. Users and criminals alike are splitting and storing data in the cloud bringing with it legal challenges (privacy and confidentiality rights) which limit the amount of data investigators can gain access to (Silva, Reed and Kennedy, 2016).

The forensics process in an IoT environment is complex. The IoT devices themselves are a challenge in the forensic realm as there are many different devices in the market (Harichandran *et al.*, 2016), what makes it even more cumbersome is the lack of standardisation for IoT devices. The data stored on the devices could be so little and of no historical or evidential value. The IoT devices are always connected which makes them more volatile (Silva, Reed and Kennedy, 2016). This adds an extra layer of complexity in the forensic process. Privacy is also a key element in maintaining the confidentiality of data as it may lead to exposure of Personally Identifiable Information (Ziegeldorf, Morchon and Wehrle, 2014).

Furthermore, Singh *et al.* (2018) mentioned accountability as one of the IoT forensics challenges. The authors stress that this is because different entities manage the composition and the interactions between the IoT components. This is further argued by the authors that IoT technology is opaque due to the over usage of the IoT components thereby behaving in ways that vary from the original intention. Another key challenging aspect brought out by the authors is that the ownership, management, and operation of IoT components is done by people or companies that may be of diverse geographical locations governed by their own native laws and regulations.

The integration of IoT devices brings with it the challenges related to security more so as highlighted by Mukundan, Madria and Linderman (2014). The authors note that confidentiality and integrity compromise is a key security and forensics hindrance. The need to assure the user that only authorised parties get access to the data is an issue. There is a compromise of data integrity if unauthorised access is gained to the data.

To differentiate between Digital Forensics and IoT Forensics, a clear definition and understanding of an IoT environment is required. According to the National Institute of Standards and Technology (NIST) by Megas *et al.* (2017) on IoT Cybersecurity Colloquium, it is noted that there is no common agreement on the definition of IoT. One definition from this NIST publication described IoT as things like sensors and devices (excluding computers, smartphones, and tablets) that are connected through the internet to communicate and/or transmit data with or between themselves. Another definition refers to IoT as devices or things that are not fully operational computers, instead they are built for a specific purpose containing sensors which enable them to communicate through the internet. Another definition proposed by Baig *et al.* (2017) IoT is defined as connecting smart devices like sensors to a network through the internet.

There are several attempts acquainting IoT, however they are generic or broad, which may not reflect the actual meaning of IoT. In this research, 'things' were considered as devices (for example, agents, sensors, and actuators) that can communicate, detect and/or measure data with very limited or no processing power and have low storage capacity. Therefore, this research defines IoT as pervasive connected devices through the internet that collect, detect and/or measure data. We refer to things with very limited human control, although it could be manageable and/or configurable. Things could be classified based on their functionality, there are some things that can process data, while other things can detect and/or measure data and perhaps several others just observing (monitoring) data motion.

IoT forensics can be defined as a branch of digital forensics that combines three levels namely, device level forensics, network level forensics and cloud level forensics. This is explained further by Zawoad and Hasan (2015) who stated that IoT forensics involves the investigation of IoT infrastructure (device, network, and cloud). This whereby local memories of IoT devices could be investigated for potential evidence, network log files could be retrieved to reveal user activities and the cloud being a major storage of IoT device data could be a source of potential evidence.

The key players in the IoT forensic investigations are the Law enforcement agencies, IoT manufacturers, IoT users (these might be the suspects in a case) and the digital investigator (this could fall under law enforcement agency). These parties involved in the IoT forensics have different accountability and responsibilities. There are conflicting interests that emanate during the forensic process to apportion liabilities and obligations. The users have a right to privacy and confidentiality of their data that must be upheld. The law enforcement agencies in their pursuit for keeping the internet world safe, may use means like profiling and surveillance that may infringe on user privacy rights.

Most research on how IoT relates to digital forensics is argued by Harichandran, et al. (2016) as being more theoretical than practical. There is a need to study and link the conflicting aspects of IoT forensics to identify potential practical solutions that overcome the challenges.

The aim of this section of this thesis is to review the current legal and technical challenges of IoT forensics by devising a quadrant model that links conflicting aspects in IoT forensics and recommending potential ways to bridge the challenges related to data protection laws and privacy.

2.6.1 Legal Implications in IoT Forensics

Chike (2018) noted that the lack of universal rules and regulations coupled with standards and protocols will hinder IoT from being integrated in various organisational networks. Due to the continued use of IoT devices, there has been a rise in the creation of new regulations.

The collected data from IoT devices can be misused in a discriminatory way that goes against the user privacy, it is therefore upon the organisations who hold this data to ensure that only authorised access is granted. The inability of organisations to put in place management control measures for Internet of Things complexities persists to be a risk concern. Policy makers have been left to scamper in finding measures to combat these security and privacy concerns.

The nature of the law is complex with many layers and is distributed across different domains meaning that there are different interpretations and application to people impacted. It therefore follows that it is difficult to assign accountability due to the complexity of IoT and the different interpretation of the law.

In the ever-evolving landscape of cloud computing, the challenge of ensuring the independence of location becomes more pronounced due to diverse regulatory frameworks and data governance standards across the globe. Striking a delicate balance between providing seamless access to cloud resources and complying with region-specific data protection laws poses a formidable obstacle. Navigating through these complexities demands innovative solutions that not only transcend geographical boundaries but also safeguard user data and uphold the integrity of cloud services in an increasingly interconnected world.. This is noted by Hon, Millard and Singh (2016) who state that the use of IoT devices, some of which are highly portable coupled with complex supply chains may exhibit challenges especially in determining which country's laws to use to apportion rights and liabilities.

The challenging accountability aspects in IoT environment as identified by Singh *et al.* (2018) are governance and responsibility, privacy and surveillance, and safety and security

In IoT regulations, two areas of significance are brought out, these are legal obligations and liabilities, and regulation of personal data.

Obligation and Liability

For a forensic process to run smoothly, full disclosure and transparency is of utmost importance. Accountability can therefore only be apportioned if the manufacturers of IoT systems are transparent about the workings of the system. Silva, Reed and Kennedy (2016) state that it is within the law for a technology manufacturer whose service leads to a loss or injury to demonstrate that the actions taken were reasonable or fair, failure to which, the manufacturer faces liability.

It would be reasonable to eliminate the human element by implementing a machine learning algorithm to be run on the data and produce a report which is only to be accessed by authorised parties. However, as this approach may be acceptable by the law enforcement agencies, it may not be acceptable to both the suspects (data owner) and the Cloud Service Providers (CSP). There must be assurance of confidentiality and integrity to the data owners that their data is safe and the CSPs do also need assurance that their cloud service infrastructure is not compromised.

Transparency obligations are enshrined in the data-protection law to data subjects and regulators. When forensically assessing liability, user's liability is mostly based on negligence where no reasonable actions were taken to avert likely risks. Users are expected to be aware of the workings of a particular IoT device before using. Manufacturers are not obliged by law to explain how the developed technology works other than to keep up with the data protections requirements (Silva, Reed and Kennedy, 2016)

Privacy and Data Protection

The data protection laws, as emphasised by Singh *et al.* (2018), are underpinned by basic principles which are; being fair, legitimate processing, being limited to the purpose, being accurate, data minimisation, storage limitation, integrity, and confidentiality.

The European Union (EU) General Data Protection Regulation (GDPR) articles (European Union, 2016) have a key principle of EU data protection law which stipulates that the processing of personal data should be done in a manner that is lawful, fair and transparent. As required and emphasised in the Association of Chief Police Officers (ACPO) guidelines, the forensic process must be conducted in a manner that should create audit trails that can be accessed by a third party and achieve the same results.

It is challenging to apply data protection rules on user data because technologies that generate and produce individual data have evolved dramatically with the ever-growing IoT environment. It can be observed that almost all data is seen as personal data with strict rules governing personal data more so of special interest categories.

It is also difficult to apportion liability due to the dynamic supply chain of IoT which is multi-layered with multiparty ownership that could be spread across many geographical locations with different regulations of operations.

As part of data protection and privacy, individuals have a right to be forgotten – in this case, they may ask that their personal data is deleted; this is may not be feasible in an IoT environment that is hugely distributed.

2.6.2 Personal Data in IoT

The emergence of IoT has resulted in major concerns related to privacy, security, trust and governance. These concerns are unsurprising as they have been deemed as the potential greatest hinderance to adoption of IoT. The capability of IoT devices like Closed-Circuit Television (CCTV) to capture data that is not necessarily of the owner of the device but any other person in the vicinity without their knowledge is a breach of privacy (Hon, et al., 2016).

Walden (2013) note that many issues related to the privacy and data protection have arisen from cloud services which includes government agencies accessing people's private data illegally. The other issue arising from privacy and data protection is the use of personal data for inappropriate purposes like profiling/discrimination (Collins *et al.*, 2014).

It should be noted that huge volumes of data are collected by IoT devices, in most cases this collection is done without the knowledge of the IoT device users. The level of knowledge of these users of how their data is collected and used is very limited to enable them give free and informed consent.

What personal data is regulated?

Personal data is any data that relates to an identifiable living individual. This data is protected under the data protection laws. The identification of a natural person can be done both directly or indirectly through identifiers like their names, number of identification (ID number), data related to their geographical location, and or their online identity through their IP addresses. Although still personal, data can be pseudonymised (remove identifiers or replace) to help in the reduction of privacy risks which makes it hard to identify individuals. It should, however, be noted that GDPR does not cover information relating to institutions, foundations and corporations which are legal entities because their data is not personal data. Privacy rights can be referred to as the right to one's personality.

The EU GDPR data protection laws stipulate that the storing or accessing of personal data of a user held by an organisation must only be consented to by the user. This therefore means that the user must give consent for any action on their data. Article 8 of the EU GDPR in particular covers many rights related to the protection of personal data (Kokott and Sobotta, 2013).

Who is responsible for personal data?

Controllers control the purpose and how the data is processed under the EU data protection laws. The controllers are therefore primarily responsible and liable to comply with the laws. In instances where data is processed by third parties on behalf of controllers, the third parties must abide by the regulations. In most scenarios, it is observed that the service providers are the controllers and processors of personal data.

The EU GDPR regulations have introduced huge fines for breach of user data privacy. There is direct obligation and liabilities to controllers and processors of personal data with those who breach security obligations being fined amounts not exceeding 20 million Euros or 4% of total annual global turnover, whichever is higher (European Union, 2016).

Apportioning this liability during the IoT forensics process may be difficult to implement. This is due to many players being involved and the complex supply chain which makes identification of players very difficult.

What rights do IoT users have?

The rights of IoT users correspond to the obligations that the controller must abide by when they process users' personal data. In the event of damages caused due to unlawful processing of their data, the users have a right to seek compensation. They have rights to access their personal data, refusal for their data to be processed in relation to decision making that is automated. Users can consent for their data to be processed or if the controller has a legal justification to process the data for legitimate purposes. However, under the EU GDPR regulations, conditions for valid consent is strict because the consent must be given freely by the user (Hon, et al., 2016).

The EU GDPR regulations Article 21 gives the user the right to object. This means that, without user consent to process the personal data, data controllers must provide and demonstrate compelling legitimate reasons that override those of the users. This regulation is vague because even the very definition of compelling reasons is not provided leaving a vacuum as to how to distinguish between a legitimate compelling reason and an illegitimate one.

Article 22 of the EU GDPR data protection laws gives a user the right to choose whether or not to go through individual decision-making processes that are automated (e.g., profiling). This is also another unclear area because data controllers find it difficult in handling objections because they are forced to cease provision of all services. This leads to a situation where the users who are more concerned about privacy of their data are left with the option of either taking up the service or leaving it altogether (Wachter, 2018).

Under the GDPR laws, data controller and processors have an obligation to inform the users of how the collection, usage, disclosure and storage of their personal information is carried out and how the users may exercise their rights over that data. A report from the UK's privacy regulator - Information Commissioner's Office (ICO) (2016) indicates that out of ten controllers of IoT, six don't adequately inform their customers on the usage of their personal data.

The report showed that:

- Of the analysed devices, 59 per cent of them failed to sufficiently inform the user of how the collection, usage and disclosure of their personal data was being done.
- On the issues of storage, 68 per cent of the devices did not show how the data was being stored.
- On the user's right to be forgotten online, 72 per cent of the devices could not explain how a user could erase all their data from the devices.
- And finally, 38 per cent of the devices did not have contact information that a customer could contact in case they had concerns related to privacy of their data.

There were concerns raised relating to medical devices used by General Practitioners (GPs). Although these devices sent encrypted emails back to GPs, there were issues pertaining to the infringement of data protection laws as follows:

- Through the IoT device, control is lost in the processing of data.
- The quality of users' consent is undermined as is it difficult to get it.
- The users risk losing the whole package of services from IoT service providers if they don't give consent for processing of their data in a particular way.
- The original purpose for the processing of the data is possible abused as it may be processed more than required.

- The transmission of the personal data is at a high risk as the medium used may be prone to hackers who may steal the data.
- The data collected may be used in ways that were not initially intended because it collected from varied devices from different sources.

2.7 The Current Studies in IoT Forensics

The current digital forensics approaches in the internet of things have recently been surveyed by several authors (Abdel-Fattah *et al.*, 2023; Al-Hussaeni *et al.*, 2023; Hassan, Samara and Fadda, 2022; Ganesh, Venkatesh and Prasad, 2022) among other prior authors.

It was earlier stressed by Adjei, Babu C and Yakubu (2018) who indicated that there is indeed a need for an improved proactive model under which IoT crime scenarios can be handled. These authors concluded that none of the frameworks and models proposed from the sampled papers can be used to extract data in a timely and reliable way. This narrative is still prevalent currently as observed by Salem, Owda and Owda (2023) who suggest the necessity of crafting an appropriate framework for IoT digital forensics in order to address the obstacles and security breaches that are widespread in the diverse architecture of IoT settings.

However, it can be noted that there have been several proposed IoT forensic processes that have included methodologies, models and frameworks which have contributed to the advancement of research in this area.

This thesis discusses these processes, methodologies, models, and frameworks in chronological order, beginning with the oldest and progressing to the most recent

The Next Big Thing process model was developed by Oriwoh *et al.* (2013), in this research, the authors propose a process model based on the challenges faced in the identification phase of the IoT forensic process. It was designed to help in the determination of potential sources of evidence. The triage is presented in a 1-2-3 zone approach whereby zone 1 consists of the identification of the person involved in the crime and potential evidence to be identified. Zone 2 covers all the possible devices within the network (routers, firewalls, switches, Intrusion Detection Systems (IDS) and gateways). All the devices and services (web, database, and cloud servers) outside the network are identified in Zone 3. This process model considers the fact that any potential evidence stored in the devices could easily become unavailable due to theft, tampering, or destruction. With this realisation, other elements within the IoT environment related to the evidence must be recognised by the investigator because they may contain valuable artifacts to aid the investigation process.

This process can be beneficial to the IoT forensics process more so in the identification phase. The challenge with this process model, however, is the development and testing. This is because it cannot be assumed that the investigator will have direct access to all the devices or even the cloud servers where the evidence could be stored. The resource limited IoT devices and the volatility of the cloud needs to be considered. The process does not also have clear laid down directions for investigators to follow while conducting the analysis.

The Next Big Thing was later integrated by Perumal, Md Norwawi and Raman (2015) through the top down forensic approach methodology which was designed to provide a novel approach that enables IoT forensics investigators through defined Standard Operating Procedures (SOPs). It is an integrated model of the 1-2-3 zone model. The top-down forensics approach methodology tries to solve the challenge to do with the preservation of volatile data. Previous approaches in digital forensic investigations were vigorously conducted by this study. The study proposed approaches that can be helpful to the investigators of IoT environments. The setback is that it may not be feasible to implement its automation in a real practical environment as the authors have also not tested it practically.

The Last on Scene (LoS) algorithm was proposed by Harbawi and Varol (2017) as a model based on the Next Big Thing process model. The LoS algorithm works by identifying the location of evidence in such a way the first device to be investigated is one that was seen last on communication chain. The authors of the LoS algorithm model claim that the model saves time and resources for digital investigators because only data of interest is sought, and therefore if found in zone 1 the process terminates, and a report is compiled. The investigators do not have to go through all the zones looking for potential evidence. As implied by the authors themselves, the LoS Algorithm is a theoretical framework meaning that its practical implementation or application has not been performed. The legal implication aspect has also not been factored into this framework; this means that it may be inadmissible in a court of law.

Designed by Zawoad and Hasan (2015), this model encapsulates the IoT digital forensic processes and techniques. The authors define the term IoT forensics process in three levels of digital forensics: device, network, and cloud level forensics. The model employs a secure trusted central repository that aims to deal with the problem of IoT domain not being standardised. A chain of custody being a key part of a digital forensic investigative process, this model focuses on ensuring that a chain of custody is maintained. Unfortunately, there is no practical implementation of this model.

Proposed by Kebande and Ray (2016), this process model is based on a generic approach that analyses digital forensics data in the IoT setup through process concurrency. The model is presented to capture data at all the three levels of the IoT forensics.

Through the process concurrency, the model aims to establish IoT forensics readiness and increase the rate at which the digital evidence extracted is admissible in a court of law. From the readiness point of view, this model will require a momentous consideration to proactive scenario-driven activities to ensure that the potential evidence is captured with the IoT setup and that implementation for extraction and preservation of the evidence is done in a procedure that is well-defined and documented. It is through this that the evidence will be forensically sound. The drawback with this model, however, is that it is purely based on theoretical approach in the collection of the forensic data. There is no physical experimental in its implementation and evaluation thereby casting doubts on its practicality.

As an extension of DFIF-IoT, Kebande *et al.* (2018) proposed an Integrated Digital Forensic Investigation Framework (IDFIF-IoT) which claimed that DFIF-IoT was generic with processes that relied on ISO/IEC 27043 international standards while IDFIF-IoT includes organisational policy making it more policy oriented. This framework is still more theoretical than practical and as also pointed out by the authors themselves, the framework needs more development so as to identify more critical aspects of forensics.

The authors,(Rahman, Bishop and Holt (2016) explored the mobility forensics in its context to IoT. The process of data acquisition and the classification methods for smart home devices are discussed in detail. An analysis of an attack scenario of the collected data is also discussed and a model is proposed that handles such scenarios.

The proposed model seeks to address; what happened, when it happened, how it happened, who and/or what did it, why it happened and what data was collected? This paper contains valuable information that can be used as a framework for controlled IoT forensic investigations. However, it is limited to only one device being tested. The model proposed was not implemented, deployed and neither was it tested. The authors also assumed the full availability of data, this is usually not the case for forensic investigations.

IoT mobility forensics model is used by Ryu, Moon and Park (2018) to describe a process of data retrieval from smart devices and how this data can be classified and analysed. An analysis was performed based on a scenario of attacking the collected data and proposing a forensic model that fits such scenarios. The authors claim to collect data using Wireshark; however, they do not reveal from where this data is preserved as this is very crucial in a criminal investigation. They do not also tell if this data is live data, and if yes, how can it be a criminal case when all is planned and acted? If no, where was this data stored? Internally or in the cloud?

In Banday (2018) experiments, mobility forensics is used whereby cookies are collected from kid trackers to locate a missing child. The forensic model proposed tries to establish what happened, why it happened, when it happened, how it happened, how data was collected, and what data is needed from the trackers. However, as also noted by the author, none of the processes proposed in the model have been tested or tried.

The Cloud-Centric Framework that is able to isolate Big Data as forensic evidence from IoT (CFIBD-IoT) (CFIBD-IoT) framework proposed in this study consists of three layers. It recommended a standardised technique of how to acquire and isolate evidence. Authored by Kebande, Karie and Venter (2017), the research investigated how the spread of IoT has led to the complexity of the investigation process. A case study of BitTorrent is used as a focus point where cyber criminals have explored the avenues opened up by IoT through information theft and side channel attacks facilitating crime-as-a-service.

The anonymisation techniques have been used to hide the privacy of the users thereby allowing private communication, this has made it possible for cyber criminals to exploit the feature and attack IoT setups. The challenge is that even though the law enforcement agencies may get access to the client machine, they may not have access to evidence that may be stored in the cloud.

The Privacy-aware IoT-Forensics (PRoFIT) model proposed by Nieto, Rios and Lopez (2017) incorporates privacy in its investigation process by making use of the requirements of ISO/IEC 29100:2011. Assurance for privacy encourages IoT devices to participate in digital forensics investigations in a voluntary basis. The model emphasises on the importance of collaboration between devices that are nearby to aid in the collection of the evidence and determine the context within which the crime falls. This makes it ideal to fit into a concept of a digital witness. The evaluation of the proposed model was conducted in a coffee shop which was IoT enabled with an actual malware propagation. Like many other models proposed, the PRoFIT model lacks the practical part and therefore remains a theoretical model.

A research by Al-Masri, Bai and Li (2018), Fog-Based Digital Forensics Investigation Framework (FoBI) utilises the fog computing model by which intelligence is pushed by a gateway to the network edge. An example is given whereby a last known location of a device can be traced and any malfunction can also be identified using the log files. When a suspicious activity is found during the FoBI investigation analysis, the nodes or other IoT devices are notified of the potential threat so that the propagation of the threat to other IoT devices it minimised or eliminated. The FoBI framework, though workable, is not suitable for a general IoT forensic investigation. It can well be implemented in a home, or a controlled environment and its main purpose would be to track user activities and notify of any suspicious activities. The fact that a FoBI management software has to be installed on a node or a gateway may raise questions related to surveillance and may fail the test of judicial process in a court of law.

A research by Babun *et al.* (2018), IoTDots is a novel digital forensics framework for smart environments. It comprises of IoTDots-Modifier (ITM) and IoTDots-Analyser (ITA) as the main components. Through the ITM, applications on the smart device can be analysed by way of looking for relevant information that can be of forensic value. The applications on the smart device are then modified by insertion of particular logs which in turn send the forensically relevant data to the IoTDots Logs Database (ITLD) at runtime. During the forensic investigation process, data processing and machine learning techniques are applied through the ITA on the ITLD data. This process involves the learning of the state of the IoT environment and the behaviour of the users in the time of interest of the forensic process. Violations are then identified by the events and actions against the security policies put in place.

This framework is one of its kind in IoT forensics as it has practical and experimental evidence. However, it is specific to a controlled group of IoT device users and may not be viable for random devices as IoT environments are flooded with many different devices. This is because, as rightly indicated by the authors, some IOT devices are resource constrained and may not have smart applications installed on them, this means that this framework cannot work on such devices. Another drawback on this framework is that one of the components (IoTDots-Modifier) goes against the forensic principle of modification of the evidence and therefore may not pass the test of a court of law. The authors do not specify if they have the full consent of the users as per the European General Data Protection Regulations when installing these components on the devices. Moreover, this framework appears to be a security framework rather than a forensic one because, critically studying it implies that it is a tracking system.

A research by Kebande, Karie and Venter (2018), the authors proposed an architecture that is able to forensically incorporate Digital Forensics Readiness (DFR) within the IoT environments by planning and preparing for any intrusion to the IoT setup. The authors stated that before their paper, there was no known model or framework that could incorporate DFR for the purpose of incident preparedness in IoT setups. The framework has three distinct entities which are: Proactive Process (detects pre-incidents), IoT Communication Mechanism (provides smart communication strategies on the intelligent network for machine-to-machine devices) and Reactive Process (handles digital investigations in post-event response process). Although this framework has a practical and experimental results, it does not show how the general digital forensics processes of preparation, identification, acquisition, preservation, analysis, and reporting. This is exhibited by its inability to show the chain of custody and the acquisition of potential data at all levels of IoT forensics (device, network and cloud levels). A practical demonstration of a report or a process that is admissible in a court of law needs to be specifically outlined and presented. The framework is based majorly on how an IoT environment can best prepare for a potential security incident.

The Forensic Investigation Framework for IoT (FIF-IoT) as described by Hossain, Karim and Hasan (2018) is a framework that uses public digital ledger to forensically investigate IoT-based systems. The framework operates by storing in a Bitcoin-like public digital ledger all the interactions that the device makes with other devices, users, or cloud. The stored data is used as evidence. The setting of the framework allows evidence acquisition and enables the verification of evidence during the investigation process. This framework though well thought and explained; the experiments subjected on it cannot warrant its use for a forensic process that can stand the test of a court of law. The authors claim that there is integrity preserved yet they do not show how this is achieved in their experiments.

An IoT forensics framework proposed by Hossain, Hasan and Zawoad (2018) called Probe-IoT uses public digital ledger in searching for evidential facts in incidents in systems that are IoT based. Through the framework, interactions between IoT entities like IoT devices, IoT users and the cloud, are collected as evidence and stored securely in a Bitcoin like technology. The authors claim that Probe-IoT framework guarantees confidentiality, integrity, non-repudiation, and anonymity for the stored evidence data. This is because it is stored in public ledger. The framework also provides a mechanism in which during an investigation of a malicious incident, the integrity of the stored evidence is verified by authentication for any retrieval.

This research provides a tight security in accessing the evidence collected and can be extended to any evidence that is not necessarily IoT based. It would have been better if a real-life simulation of collection of data in a typical IoT forensic investigation was performed to show how this data is acquired. After the acquisition, the authors should have demonstrated how it is securely preserved using the framework and how its access by different parties as outlined in the paper is implemented.

In this paper, the forensic artifacts retrieved from Nest's IoT devices (thermostat, indoor and outdoor cameras) are analysed by the authors, Dorai, Houshmand and Baggili (2018). These devices were controlled by an iPhone. The source of the data from for the logical backup of the iPhone. Google Home Mini was also integrated by the authors as another method to control the Nest devices being studied. It is claimed by the authors that their work produced a first usable forensic tool named FEAAS from open-source research. The tool, as the authors state, consolidates evidentiary data into a readable report depicting user activities and what might have triggered the activities thereof.

From the experiments and the analysis done by the authors, it is evident that they had possession of all the devices and access to all the databases storage sites. The authors have simulated how smart home can be controlled and also given details of when, what, and how the events take place. All this information could be very valuable in a case as the investigators can get access to the relevant information. However, this is usually not the case in many digital forensic cases because in most cases, the investigators have no access to the control phone which in this case, the authors have retrieved the logical data from. The tool created could also be restricted to the mentioned devices alone.

This research seeks to use data reduction which entails selectively imaging data. The acquisition process is automated and huge amount of data is quickly analysed in time. The authors, Quick and Choo (2018b), state that the paper outlines a process of analysing huge volumes of data for forensic purposes. This data includes that from dissimilar devices.

It is noted by the authors that as many devices interconnect through the internet and upload huge amount of data to cloud platforms distributed around the globe, it is important to identify relevant potential data of evidence for forensic purposes. Securing of the crime scene is also problematic because the wireless crime scene may leak data as the investigators process physical devices.

The authors further note that the analysis of dissimilar devices is a challenge as many of these devices that flood the market do not adhere to forensic readiness principles. The data from these devices could as well be proprietary and the manufacturers are in most cases hesitant to give out details about the data structures used for fear of leaking their secret to their competitors. The reverse engineering that may be performed on these kinds of devices may not pass the test of a court of law as stated by the authors.

Although the research was aimed at performing analysis in a faster way, the time taken for acquiring data in these experiments is still too much, there is need to look for mechanisms to ease the process of acquisition. However, there are useful forensic tools that the authors have proposed and used in their research that are very essential in the digital forensics' realm. The research cannot be fully relied on as the authors state that they had limited access to the data and could not therefore view or query the data to reveal the number of dissimilar devices contained in the data.

The authors, Nik Zulkipli, Alenezi and B. Wills (2017) proposed two approaches for conducting IoT investigations based on low security mechanism and constraints encountered in IoT setups. The real time approach for IoT forensics proposed in this paper appears to be too general. The authors have implemented what is perceived to be done in traditional digital forensics into IoT forensics. This mode of approach will only work if the investigator has a full access to the device, the network and where the data is being transmitted to and/or from (maybe the cloud). It could be a measure for IoT forensics readiness in a controlled environment. No practical work has been performed by the authors to illustrate their proposals.

A summary is provided by Shin *et al.* (2018) of methods to collect and analyse data to improve the digital forensic process in IoT environments. Amazon Echo and Z-Wave devices as part of smart IoT devices together with a router were analysed to reveal important forensic evidence that can be extracted. This paper however lacks the practical solutions that can be applied in scenarios of the general IoT forensics as it focusses more on Amazon Echo, Z-Ware, and a home router.

A three-layered architecture is proposed by Al-Sadi, Chen and Haddad (2018) which keeps track of the three level of the IoT forensics (device, network and cloud) and showcases where potential evidence can be found within these layers. The authors have outlined different types of open-source tools that can be used in every level but fail to give experiments on how this can be done. This research remains a theoretical work like many others.

A research by Bouchaud, Grimaud and Vantroys (2018) focussed on the collection of data from IoT devices. The authors discussed the mode of data identification and the methodology of data classification from IoT devices to find the best available evidence. Tools and techniques to for identification and location of IoT devices are also proposed. The authors also claimed to develop a concept of "digital footprint" in the crime scene based on frequencies and interactions mapping between devices. The classification methodology used in this paper is too general and may be limiting to other IoT scenarios. The issues to do with synchronisation of data and the aspects that address the legal issues also need to be discussed further as the authors noted.

A framework is proposed by Goudbeek, Choo and Le-Khac (2018) for forensic investigation in IoT environments (smart homes). The authors simulated the three case studies to illustrate all the three levels of IoT forensics (device, network, and cloud). They claimed that their research fills the gap on how to acquire any type of data that may be potential evidence in a smart home setup. This framework looks to be very helpful to the digital forensic investigators, however, as these case studies are only simulations, it may be reasonable if the framework is applied in a real-life situation.

A forensic investigative framework is presented by Rondeau, Temple and Lopez (2018) to be used in Industrial IoT applications. Their framework is based on the fact with which they allude that most forensics investigations happen at the higher layer digital domain meaning that the lowest layer domain remains hugely unexploited. They have therefore performed forensic investigations on the lowest physical layer of the network and illustrated what evidential data can be found within that lower physical level.

A framework called Trust Internet of Vehicles (Trust-IoV) is proposed by Hossain, Hasan and Zawoad (2017) whereby evidence that is trustworthy from internet of vehicles systems is collected and stored. From the experiment results of the framework, it is shown that in scenarios where there are strong adversaries, the framework can work with very minimal strains.

A proposal by Le *et al.* (2019) on a permission blockchain based mechanism for IoT forensics which enhances integrity, authenticity, non-repudiation in the process of collecting and preserving evidence.

The system provided Oriwoh and Sant (2013) was aimed at providing security and forensics capabilities for smart homes. The strategies involved in this system can be helpful in an investigation more so for first responders as it has forensic readiness capabilities.

A framework is provided by Chi, Aderibigbe and Granville (2019) for acquiring data saved stored on the cloud by IoT devices. The setback in this framework is that the authors have not exhaustively provided the information relating to how they have developed their forensic tool, and the tool only seems to work with android phones.

The acquisition of data, as shown by the authors Meffert *et al.* (2017), can be done both from the devices and the cloud. The author's Forensic State Acquisition from Internet of Things (FSAIoT) framework was however not possible to retrieve deleted or historical data from IoT devices. More experiments should also be done to reveal the extent to which varied IoT devices can be able to work with this framework.

A concept is proposed by Bouchaud, Grimaud and Vantroys (2018) where traces of IoT devices can be tracked down and identified. A central bridge device is used to connect to other devices in the surroundings. The identified devices are ranked based on their importance of interest. The main setback with this concept is that the world is flooded with varied devices which may not be identified. Zia, Liu and Han (2017) proposed an application specific IoT forensics investigative model where data is acquired, examined, and analysed resulting into a generated report.

The authors Quick and Choo (2018a) proposed a national repository knowledge base for digital forensics experts. The knowledge base, with the necessary security control measures, could be expanded to allow for inclusion of methods that are suitable to aid in data reduction in a digital forensic process.

An investigation is done by Koroniotis *et al.* (2018) on how Machine Learning techniques can be used to develop a mechanism for network forensics to track suspicious activities of botnets based on network flow identifiers. This piece of work can be used to enhance IoT forensics especially in cases of compromised IoT devices through botnets, however, as it is an intrusion detection mechanism, it remains to be a forensics readiness process.

A forensic framework is proposed by Chhabra, Singh and Singh (2018) for big data in IoT environments for precision and sensitivity. The framework employs a Machine Learning (ML) approach using the Google's MapReduce as the basis for understanding traffic, extracting, and analysing the data. Open-source tools that support parallel processing and scalability have also been used in the framework. Comparatively analysed against other ML models, the framework exhibited a performance metrics of 99% sensitivity.

Ryu *et al.* (2019) proposed a framework (Blockchain-based Framework IoT Forensics) leveraging blockchain technology to improve forensics investigations in the realm of IoT. This framework boasts enhanced security by utilising cryptography on the blockchain, making it tamper-proof and fostering trust in the collected evidence. Additionally, by recording all interactions between IoT devices on the blockchain, the chain of custody for evidence is streamlined, simplifying the process of tracking its origin and ownership. However, some limitations exist. The vast amount of data generated by numerous IoT devices could potentially congest public blockchains, raising scalability concerns. Furthermore, storing all data on a public ledger might pose privacy issues for some users. Finally, implementing and managing a blockchain-based system for forensics adds complexity for investigators. While this framework offers a secure and transparent approach to IoT forensics, addressing scalability, privacy, and implementation challenges is crucial for its widespread adoption.

A "Holistic IoT Forensic Model" was proposed by Sadineni, Pilli and Battula (2019) emphasising a comprehensive approach to digital forensics for IoT. This model leverages the established ISO/IEC 27043 standard, providing a structured and well-defined process for handling IoT forensics investigations. This can significantly improve consistency and reliability in evidence collection throughout the process. The model's strength lies in its comprehensiveness, encompassing proactive preparedness for forensic readiness, reactive initiation upon encountering an incident, and forensic analysis itself. This ensures a thorough investigation that covers all stages. However, some challenges exist. Putting the model into practice might be difficult due to the vast diversity of IoT devices with varying capabilities. Additionally, the model doesn't explicitly address potential privacy concerns that could arise during data collection and analysis specific to IoT forensics. Overall, this model offers a valuable foundation for standardized and comprehensive IoT forensics, but for real-world application, it would need to address practical implementation challenges and incorporate clear privacy considerations.

Islam *et al.* (2019) introduced the "IoT Comprehensive Framework" designed to offer a holistic approach to digital forensics investigations for IoT devices. This framework incorporates several key modules: a Data Acquisition Module for collecting evidence, a Preprocessing Module for filtering and preparing the data, an Analysis Module for in-depth examination, and a Secure Provenance Module for maintaining a secure record of access history to the evidence. Additionally, an Initialisation Process ensures proper configuration before investigation begins. While this framework provides a comprehensive plan for handling IoT forensics, some limitations exist. The effectiveness of data acquisition might vary depending on the specific device and its communication protocols. Furthermore, the success of the framework hinges on the secure implementation of its various modules, particularly the Secure Provenance Module, to ensure the integrity of the evidence trail. Overall, the framework offers a well-structured approach to IoT forensics, but for real-world application, addressing device-specific data acquisition challenges and robust security implementation will be crucial.

The Digital Forensics Investigation Model (DFIM) model, proposed by Qatawneh *et al.* (2020), offers a seven-stage framework for digital forensics investigations in the IoT domain. This model emphasises principles like security, privacy, accuracy, and data reduction. While the DFIM model presents a structured approach, some critical aspects need consideration. Firstly, the success of the model relies on the effectiveness of its "modifier" component, responsible for initialization and data acquisition. Difficulties might arise due to the heterogeneity of IoT devices and their varying communication protocols. Secondly, balancing the need for data reduction with preserving crucial forensic details is a challenge. Finally, the paper doesn't delve deeply into the specifics of how the model addresses privacy concerns during the investigation process. In conclusion, the DFIM model provides a valuable foundation for IoT forensics investigations, but for real-world application, it would benefit from addressing challenges related to device-specific data acquisition, data reduction strategies, and detailed incorporation of privacy considerations.

Koroniotis, Moustafa and Sitnikova (2020) proposed the "Particle Deep Framework" (PDF) for IoT network forensics. This framework leverages machine learning, specifically deep learning with a twist: Particle Swarm Optimisation (PSO) is used to fine-tune the deep learning model's hyperparameters. This approach aims to improve the model's ability to detect malicious activity on the network. The framework boasts impressive results in their study, achieving high accuracy with a low false alarm rate. However, some limitations are worth considering. Firstly, the effectiveness of the framework relies heavily on the quality and completeness of the training data used for the deep learning model. If the training data doesn't encompass a wide range of attack types, the model's accuracy in real-world scenarios could be compromised. Secondly, the computational cost of training and running deep learning models can be significant, potentially limiting its feasibility for resource-constrained environments. Finally, the paper doesn't explicitly discuss the interpretability of the deep learning model's decisions. Understanding why the model flags certain activities as suspicious can be crucial for investigators. Overall, the Particle Deep Framework offers a promising approach to IoT network forensics with its focus on deep learning and hyperparameter optimisation. However, addressing the limitations related to training data comprehensiveness, computational demands, and model interpretability will be essential for its wider adoption.

Saleh *et al.* (2021) presented the "Common Investigation Process Model for Internet of Things Forensics" (CIPM). This framework emphasises a standardised approach to IoT forensics investigations. The CIPM outlines four key stages: preparation, collection, analysis, and final reporting. This structured approach can benefit investigators by facilitating a well-organized and documented investigation process. However, some limitations exist. The CIPM offers a general framework, and its effectiveness in real-world scenarios might depend on the specific nature of the investigation and the capabilities of the involved IoT devices. Additionally, the authors don’t delve deeply into the specifics of handling challenges like data heterogeneity across diverse IoT devices or the complexities of secure evidence storage and management within the CIPM’s structure. Overall, CIPM offers a valuable foundation for standardised IoT forensics procedures. However, for real-world application, addressing the limitations related to device specific considerations, data security concerns, and detailed guidance within each stage of the process would be beneficial.

Ahmed, Yousef and Mohammad (2021) proposed an IoT Forensic Model that leverages third-party logs for forensic analysis. This approach aims to address challenges associated with traditional methods that require direct access to IoT devices, which can be difficult or even impossible in some cases. The model utilises logs collected from surrounding infrastructure, such as network routers or cloud platforms, to reconstruct events and identify potential compromises. While this offers a valuable alternative approach, some limitations need consideration. Firstly, the effectiveness of the model relies heavily on the availability and comprehensiveness of third-party logs. If relevant logs are missing or incomplete, the ability to reconstruct events accurately might be hampered. Secondly, extracting forensic data from third-party logs might require parsing and interpreting complex log formats, which can be challenging and time-consuming. Finally, the model needs to address potential privacy concerns, as third-party logs might contain sensitive information beyond the scope of the IoT device under investigation. Overall, the model offers a promising approach to IoT forensics by utilising third-party logs. However, for real-world application, it would benefit from addressing limitations related to log availability, parsing complexities, and incorporating strong privacy considerations.

Fagbola and Venter (2022) proposed a "Smart Digital Model for Shadow IoT" (SDMSI) to address the challenge of hidden or unauthorised IoT devices (shadow IoT) on a network. This framework leverages machine learning algorithms to analyse network traffic and identify patterns that deviate from expected behaviour, potentially indicating the presence of shadow IoT devices. The SDMSI can be beneficial for enhancing network security and identifying potential vulnerabilities. However, some limitations need to be considered. Firstly, the effectiveness of the model depends on the quality and completeness of the training data used to train the machine learning algorithms. If the training data doesn't encompass a wide range of legitimate IoT device activity, the model might generate false positives by flagging normal behaviour as suspicious. Secondly, machine learning models can be complex, and their decision-making processes might not always be easily interpretable. This can make it challenging for investigators to understand why the model identifies certain network activities as indicative of shadow IoT devices. Finally, the computational cost of training and running machine learning models can be significant, potentially limiting its feasibility for resource-constrained environments. Overall, the SDMSI offers a promising approach for detecting shadow IoT devices, but as observed from the previously reviewed work, for real-world application, addressing limitations related to training data comprehensiveness, model interpretability, and computational demands will be crucial.

Mazhar *et al.* (2022) proposed a Machine-to-Machine (M2M) framework for forensic analysis of IoT devices. This framework utilises machine learning to automatically detect attacks on IoT devices. Additionally, a third-party logging server is introduced to capture evidence of these attacks. While this approach offers automation and centralised evidence collection, some limitations exist. Firstly, the reliance on machine learning algorithms necessitates high-quality training data that encompasses various attack types. Inaccurate or incomplete training data could lead to the model missing genuine attacks or generating false positives. Secondly, the security of the third-party logging server is crucial, as it becomes a central repository for sensitive forensic evidence. Any breaches or vulnerabilities in this server could compromise the integrity of the entire investigation. Finally, the authors don't provide much about the specifics of how the framework handles the heterogeneity of IoT devices and their communication protocols, which can impact data collection. Overall, the M2M framework offers an innovative approach to IoT forensics with its focus on automation and centralised logging. However, as previously stated by this thesis, for real-world application, addressing limitations related to training data quality, third-party server security, and handling diverse IoT device capabilities would be essential.

Kim, Park and Lee (2023) proposed an improved IoT forensic model that prioritises identifying connections and interactions between devices (interconnectivity). This focus on interconnectivity is a major strength, as traditional forensics can miss crucial information about how devices interact within an IoT ecosystem. The framework utilises network traffic analysis to examine data flow between devices, potentially revealing communication patterns and suspicious activities. Additionally, the concept of improved digital twins is introduced, which likely represent more comprehensive virtual models of physical devices that incorporate interconnectivity information. However, some limitations exist. The details provided don't offer specifics on how interconnectivity is identified or how improved digital twins are constructed. This makes it difficult to assess the effectiveness and accuracy of these methods. Furthermore, scalability challenges arise when analysing network traffic and maintaining digital twins for a vast number of devices. Finally, the framework needs to address potential privacy concerns when analysing network data or creating detailed digital twins of devices. Overall, the framework offers a promising direction for IoT forensics by focusing on interconnectivity, but addressing limitations related to missing specifics, scalability, and privacy considerations will be crucial for its real-world application.

A framework for a Blockchain-based IoT forensics system was proposed by Makadiya, Virparia and Shah (2023). The framewrok explores using blockchain technology to improve digital forensics for IoT. While it highlights the importance of tamper-proof evidence and the potential benefits of blockchain's immutability for securing the chain of custody, there are some key areas for consideration. The framework leans on a permissioned blockchain, allowing access only to authorised participants. This ensures trust but goes against the fully decentralised nature often associated with blockchain. Additionally, while scalability is mentioned as an advantage for handling numerous IoT devices, the authors don't delve into how blockchain would handle the potential data influx. Furthermore, the focus is on data collection and verification, and it's unclear how blockchain would integrate with other forensic tasks like analysis and investigation across different devices. Overall, the concept of using blockchain in IoT forensics is interesting, but the framework would benefit from addressing the limitations of permissioned blockchains and exploring its application in the broader forensic workflow.

Table 2.1 gives a summary of the discussed frameworks, models, and methodologies above. The categorisation is based on the limitations and gaps in relation to the practical view of the proposed research as applied in the IoT forensic process. The main features of these frameworks have been identified.

Authors	Main Features	Practical View of the Forensic Process	Limitations and Gaps
(Birk and Wegener, 2011)	Service Specific solution for cloud forensics SLAs verification and compliance issues	Snapshot analysis or cloning in the Infrastructure as a Service cloud environment. Evidence Identification and Acquisition	Not feasible for all data to be cloned by the cloud service providers
(Oriwoh *et al.*, 2013), (Harbawi and Varol, 2017), (Hossain, Hasan and Zawoad, 2018), and (Meffert *et al.*, 2017)	1-2-3 Zones of IoT Forensics Systematic and structured approach to minimise the complexity of IoT investigation processes. Identification of more evidence sources in the absence of the primary source of evidence Last-on-Scene (LoS) algorithm Use of public digital ledgers to find evidence in IoT based systems. FSAIoT	Mapping the investigation process and helping to identify key areas of focus. Devices of interest identified in the focus areas established. Evidence Identification Guidance on the investigative process based on established zones	The identification of evidence is only partial. Difficulty in the development and testing No clear instructions/directions on how to carry out the analysis and the whole investigative process
(Kebande and Ray, 2016), (Hegarty, Lamb and Attwood, 2014), (Venčkauskas *et al.*, 2015), (Underwood, 2016). (Perumal, Md Norwawi and Raman, 2015), (Zawoad and Hasan, 2015), (Rahman, Bishop and Holt, 2016), (Banday, 2018), (Bouchaud, Grimaud and Vantroys, 2018), and (Oriwoh and Sant, 2013)	Review of the current tools for forensic readiness in IoT Preserving of volatile data/evidence Evidence acquisition and preservation Maintaining chain of custody The proactive (readiness) and reactive (investigation) IoT forensic process Identification and acquisition of evidence	Theoretical	Practical aspect to augment the implementation, deployment, analysis and evaluation. Too generic approaches may not be suitable for IoT forensics
(Alabdulsalam *et al.*, 2018), (Kebande, Karie and Venter, 2017), (Nieto, Rios and Lopez, 2017), (Dorai, Houshmand and Baggili, 2018), (Al-Sadi, Chen and Haddad, 2018), and (Zia, Liu and Han, 2017)	Incorporates privacy in the forensic process using the requirements of ISO/IEC 29100:2011 Collaboration of nearby devices	Identification, acquisition, and analysis of data	More vigorous experiments to explore how the current tools can be used to fit into the proposed frameworks and solutions
(Al-Masri, Bai and Li, 2018), (Babun *et al.*, 2018), (Kebande *et al.*, 2018), and (Kebande, Karie and Venter, 2018)	Builds intelligence at the edge of the of the network through a gateway. IoTDots-Modifier (ITM) and IoTDots-Analyser (ITA)	Forensic readiness Incident preparedness	May fail the test of the judicial process due to installation of a management software which may be viewed as surveillance in a public setup. No clear instructions/directions on how to carry out the analysis and the whole investigative process
(Hossain, Karim and Hasan, 2018) and (Le *et al.*, 2019)	Uses Blockchain like mechanism for evidence preservation.	Evidence preservation Chain of Custody	Vigorous experiments required for the purposes of admissibility in a court of law

	Provides privacy		
(Quick and Choo, 2018a)	Selective data imaging, automated acquisition, and quick analysis	Identification and acquisition of data	Need for finding ways to reduce the time taken for acquisition of data
(Goudbeek, Choo and Le-Khac, 2018)	To identify and acquire any kind of evidential data in a Smart Home environment	Provides guidance for quick reference for investigation processes involving smart home environments. All three levels of IoT forensics are covered	Application of the simulations used in this research should be carried out in a real-life scenario
(Sadineni, Pilli and Battula, 2019)	Comprehensive approach to digital forensics for IoT - Based on ISO/IEC 27043 standard	Offers structured process for handling investigations - Ensures thorough investigation stages	Difficulty in practice due to diverse IoT devices Lack of explicit privacy considerations in data collection and analysis
(Ryu *et al.*, 2019) and (Makadiya, Virparia and Shah, 2023)	Utilises blockchain for secure evidence tracking Cryptography ensures tamper-proof nature of evidence Ensures tamper-proof evidence and enhances security	Streamlines chain of custody for evidence Enhances security and trust in evidence Ensures tamper-proof evidence and enhances security	Scalability concerns due to congested public blockchains Privacy issues with storing all data on public ledger Complexity in implementing blockchain-based system Limited exploration of permissioned blockchain limitations Unclear integration with forensic analysis and investigation tasks across devices
(Islam *et al.*, 2019) and (Qatawneh *et al.*, 2019)	Comprehensive framework for IoT forensics Includes modules for data acquisition, preprocessing, analysis, secure provenance, and initialisation Seven-stage framework emphasising security, privacy, and data reduction	Provides a well-structured approach for investigation - Ensures proper configuration before investigation begins Focuses on security, privacy, and data reduction principles	Effectiveness of data acquisition varies with device and protocols - Reliance on secure implementation of modules, particularly Provenance Module Challenges in data acquisition due to device heterogeneity Balancing data reduction with preserving forensic details Lack of detailed privacy considerations
(Ahmed, Yousef and Mohammad, 2021)	Utilises third-party logs for forensic analysis	Offers alternative to traditional direct access methods	Reliance on availability and comprehensiveness of third-party logs Complexity in parsing and interpreting log formats
(Fagbola and Venter, 2022)	Uses machine learning for identifying shadow IoT devices	Enhances network security and vulnerability identification	Reliance on quality of training data Complexity and interpretability of machine learning models High Computational demands
(Mazhar *et al.*, 2022)	Utilises machine learning for automated attack detection	Offers automation and centralised evidence collection	Reliance on training data quality Security concerns with centralised evidence storage Handling device heterogeneity in data collection
(Kim, Park and Lee, 2023)	Prioritises identifying connections and interactions between devices	Focuses on interconnectivity for comprehensive investigation	Lack of specifics on interconnectivity identification and digital twins Scalability challenges Privacy concerns in network data analysis

Table 2.1 Summary of the Current Studies

2.8 Discussions and Analysis of the SLR

Most of the research surveyed by this SLR proposed models and frameworks that have majorly focussed on conceptual levels that are more theoretical. Further investigation and research are required to tackle among others the following key issues as also emphasised by Chernyshev *et al.* (2018) and Yaqoob *et al.* (2019):

Although sound principles have been applied in the proposed models and frameworks to tackle the complex challenges of IoT forensics, there still exists a need to conduct robust experiments that can be validated scientifically. Any new methodologies, techniques and tools developed must undergo a scientific validation.

Due to the huge data generated by IoT devices (which can be referred to as 'Big IoT Data'), it is important that the research community finds a way to create techniques that are smart to analysis the data. This data is generated from heterogenous devices which have vendor specific data formats that are varied making it cumbersome to analyse and produce reports that are admissible in a court of law when presented.

The production of IoT equipment and provision of IoT services that are readily adaptable and integrated into the current digital processes are still a challenge in digital forensics investigations. Even though measures have been taken to address security features in IoT, issues related to forensics readiness for IoT systems still remain clouded (Bajramovic *et al.*, 2016).

Privacy is a contentious issue in relation to investigation processes that involve personal and protected data as stipulated under the EU data protection laws and General Data Protection Regulations (GDPR). Full disclosure must be given to the owners. This involves letting them know that their data will be used for the investigation process and should be made aware of how the data was accessed and by whom. Those who access the data must put in place protective measures that forbids unauthorised access, any form of manipulation and loss.

Evidence admissibility in a key issue in digital forensics, however, many of the models discussed in this survey have not addressed the legal aspects related to how evidence is acquired. The challenges relating to cross-border jurisdictions are imminent in cloud forensics which is huge part of IoT systems. There needs to be propositions for solutions for legal challenges as IoT relies heavily on the cloud both for application services and architectural structure.

As evidenced by both NBT and LoS algorithm models, it is difficult to determine the scope of the investigation. This is because, potentially new evidence sources are likely to be found during the process of the investigation. With the challenges related to limited visibility and high volatility of the data exposing it to manipulation and compromise, it calls for the need for mechanisms that are practical. This can be resolved by the implementation of digital warrants which would help to effectively retrieve evidence from sources that are discovered later in the process or along the process.

In conclusion, various IoT forensics frameworks have been examined, highlighting their strengths and limitations. Blockchain-based frameworks offer enhanced security but face scalability and privacy challenges. Comprehensive models like the Holistic IoT Forensic Model and the IoT Comprehensive Framework ensure consistency but must address device diversity and privacy concerns. The Particle Deep Framework excels in accuracy but needs comprehensive training data. Standardised approaches like the Common Investigation Process Model provide structure but require detailed guidance. These frameworks provide valuable contributions but need refinement to address scalability, privacy, and implementation complexity for widespread adoption.

2.9 How can the Technical Challenges of IoT Forensics be Overcome?

Singh, et al. (2018) note that although technology is not a cure all solution in solving accountability issues in IoT forensics, it can be used to complement all the other aspects to enable come up with proper rules, regulations, and standards. To better align this thought, the authors have suggested that technical means will help in:

Control

This entails what the determination of what happens through a process that has active steps detailing how obligations and exercise of rights are met.

Auditing

Auditing will make visible what happens or what happened. This will be illustrated by proving evidence explaining the operations of the system, actions, and the recourse thereof. It is at auditing that digital forensics plays a major role in revealing what transpired in an event of loss of data, data breach or damages.

Control and audit augment the accountability considerations. The auditing will increase transparency in the IoT systems giving rise to informed decision making by users and provide evidence that can be very useful in investigation processes to apportion liability (Singh, et al., 2018).

To aid this research further, a quadrant model developed by Godfrey (2000) was used to help understand different scenarios at play in IoT forensics and propose a solution to the privacy, confidentiality and data integrity for a sound IoT forensic investigation process.

A quadrant model tries to complement conflicting elements in a social phenomenon. It relates to how different aspects ranging from law to social norms affect those involved. In most cases, these aspects are acceptable and effective, some aspects might be unacceptable but effective, others may be acceptable but ineffective and lastly aspects may be unacceptable and ineffective.

This research used this quadrant model and equated the acceptable and unacceptable elements to admissible and inadmissible (in a court of law) respectively as illustrated in Figure 2.4.

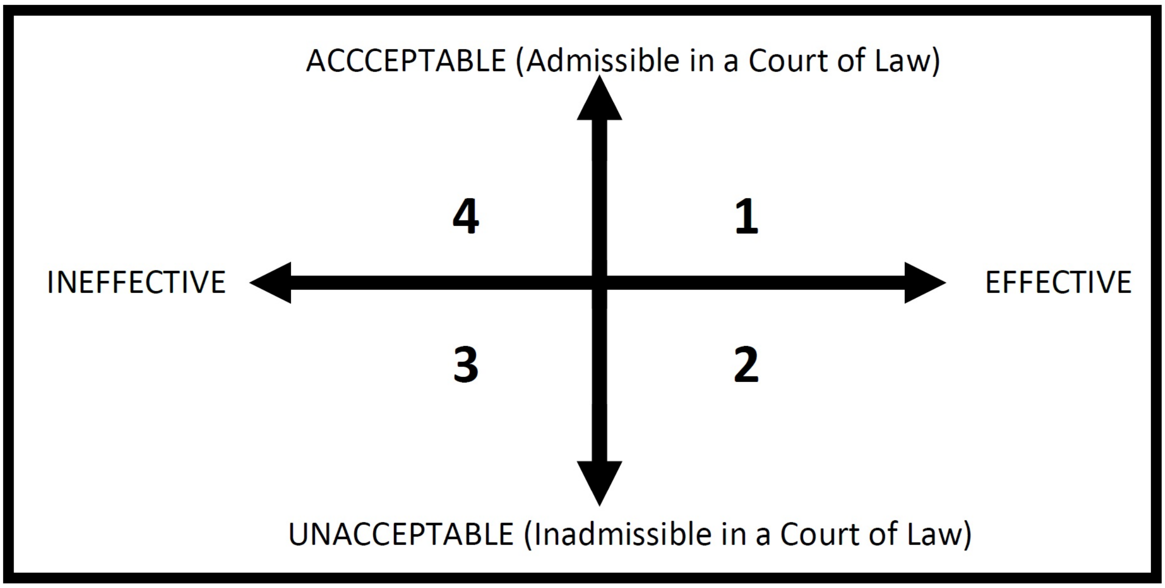

ACCCEPTABLE (Admissible in a Court of Law)
4
1
INEFFECTIVE
EFFECTIVE
3
2
UNACCEPTABLE (Inadmissible in a Court of Law)

Figure 2.4 The Quadrant Model

Quadrant 1 indicates actions that are effective and legally acceptable to all parties involved. These elements are compliant with the laws and therefore lead to an admissible report in a court of law. These can be for example, auditing and control, safety and security, confidentiality and privacy, data protection, and transparency.

Quadrant 2 is the problem area, consists of actions that are effective in increasing efficiency, but where parties have conflicting views. The activities involved in this quadrant are for example the use profiling, surveillance, tapping, eavesdropping, and cloning among many other inadmissible mechanisms. Law enforcement agencies may want to employ those mechanisms as a security measure; however, users may claim that their privacy is encroached, data being accessed by unauthorised entities. This may lead to issues related to legal obligations and liability between IoT users and IoT manufacturers.

Quadrant 3 consists of actions that are generally inadmissible in a court of law and at the same time ineffective. For these reasons, this quadrant will be ignored as it is unproductive.

Quadrant 4 are actions which are admissible in a court of law but do not contribute to increased efficiency. These elements are not admissible in a court of law. These actions can be for example, regulators banning some IoT devices and enforcing licensing for IoT devices. These actions, although admissible, they may be hard to implement meaning that they will be so ineffective and unproductive. This paper ignores the actions in this quadrant.

2.9.1 The Quadrant Model in Context

As the quadrant model is to complement conflicting aspects or interests, it is evident from this research that the conflicting parties in an IoT forensic investigation process are the users of IoT, manufacturers of IoT platforms, IoT service providers and Law enforcement agencies. All these parties have conflicting interests in that, whereas the law enforcement agencies may want to do profiling and surveillance on user activities, they are restricted by law as it is an infringement to the privacy and confidentiality of the user.

The authors Singhai and Sushil (2023), Bentotahewa *et al.* (2022), and Ahmed *et al.* (2024) claim that IoT Service Providers and IoT manufactures alike may also install backdoor applications onto IoT devices to snoop on user activities and in most cases collect users' private data for marketing purposes. The IoT Services Providers and manufacturers deny this wrongdoing whenever an investigation comes up. They blame users of negligence and would also not allow forensic investigators get to underlying structure of the technology used their devices even though they are expected to be transparent in their undertakings. These conflicting aspects or interests put in context complicate the IoT forensic investigation process. In the digital forensics' domain, forensic investigators are required to carry out their investigative process in a manner that is legally acceptable/admissible. The law enforcement agencies are also required to work within a specified terrain of regulations. All these activities are to be done without infringing the rights of a suspect.

This research therefore used the quadrant model to find reasons as to why and how the inadmissible but effective actions can be made effective and admissible in a court of law. Particularly, this is to show cause why profiling can be acceptable by the user and be effective to the law enforcement agencies and be admissible in a court of law, as illustrated in Figure 2.5.

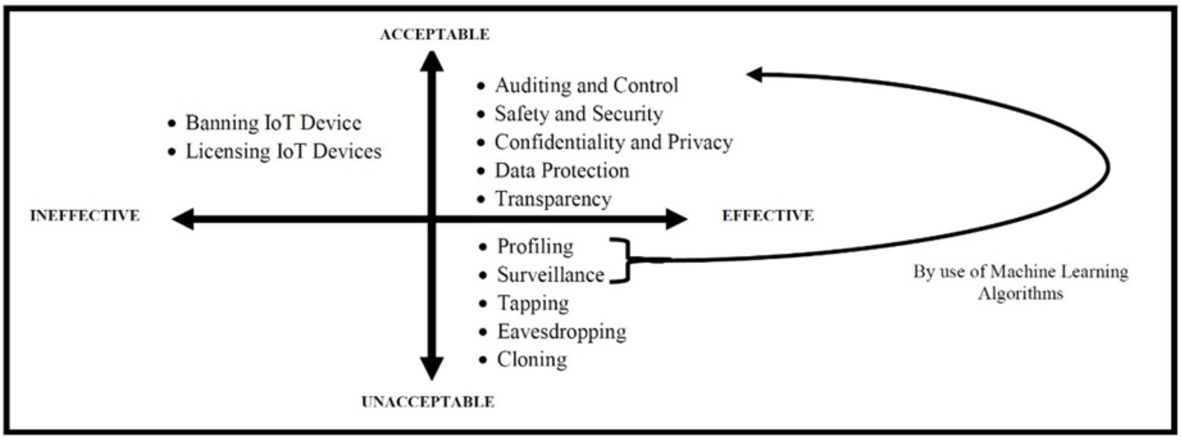


Figure 2.5 Profiling and Surveillance in IoT Forensics

2.9.2 Profiling and Surveillance in IoT Forensics

Profiling and surveillance are useful means (when used lawfully) through which law enforcement agencies can use to detect any security threats that are posed by IoT gadgets. As highlighted earlier in this chapter, IoT data is transmitted to the cloud. The cloud therefore serves as a platform through which a profiling or a surveillance mechanism can be deployed for profiling and surveillance to give alerts or reports. This paper proposes the use of Machine Learning algorithm as to implement this mechanism.

2.9.3 Machine Learning for Profiling and Surveillance

In their earlier work, Mitchell (1997) and Copeland (2000) explained that with experience, Machine Learning programs have the capability to improve automatically and learn without being explicitly programmed.

The use of Machine Learning for profiling and surveillance is to eliminate the human factor and give the owner of the data the confidence for their privacy and confidentiality, thereby ensuring only authorised access of the data is gained.

The human decision making as observed by Kamarinou, Millard and Singh (2017) is in most cases influenced by behaviours like stereotyping and prejudice. Some people make decisions based on the characteristics of profiles they perceive. This may distort evidence as it may be inaccurate, incomplete or none thereof because it may be wholly derived from stereotype and prejudice (Hildebrandt, 2008).

Machine Learning being a science that consists of algorithms that can detect patterns in data and as highlighted by Serge and Hildebrandt (2010), different profiles of individuals can be created through probabilistic processing of their personal by use of Machine Learning. This paper argues that Machine Learning algorithms can be deemed appropriate to be used in profiling and surveillance.

Singh, et al. (2016) also note that profiles only represent a version of reality which in some cases may not be the exact reality which is created from a process of data mining that includes algorithms and data used in the process.

Recent studies exploring deep learning for anomaly detection in IoT security and mobile network security showcase the potential benefits of ML in these areas for forensic purposes (Yue *et al.*, 2021; Gupta *et al.*, 2022).

2.10 Conclusion

As a fast-growing technology, IoT is providing the much-needed convenience to people through innovative IoT based applications. This has enabled devices to be connected in large numbers thereby sharing data with each other. Hackers have taken advantage of this data sharing capability exploited vulnerabilities leading to criminal activities. Through digital forensics solutions, these hackers can be tracked down and the causes of the attacks identified for appropriate actions to be taken.

The process of data acquisition in IoT environment continues to be a challenge and this gives rise to opportunities for research communities to develop new digital forensics methodologies, techniques, and tools. With the increase of attacks related to IoT, there is a massive need for successful prosecution of perpetrators. The current models and frameworks have laid a building block for future work that should be more practical and experimental. As this SLR reveals, there is a need for development of intelligent and more efficient tools that are scientifically validated to ensure reliable guiding procedures leading to successful digital investigations in IoT environments.

A plethora of digital things is encircling our world and shaping our life, they took their place in the harmonious complexity of the world. These things are connected pervasively through the internet in a very complex structure which may cause many challenges.

This literature review highlights the need for more advanced mechanisms for handling IoT forensics. This area is multi-layered and complicated as it has many players and needs more cooperation between parties involved.

The laws and regulations in place further make it a bit hard for law enforcement and forensic investigators to carry out their work as the issues of privacy and confidentiality come into play. The lack of comprehensive, widely accepted international standards, rules, and regulations to manage the IoT and cloud security are a big concern. as we continue to witness more complexity in IoT technologies with no laws to govern.

A concerted effort between multi-disciplined experts should be mooted to consolidate the main areas of conflicts and provide viable solutions for long term security measures. These efforts should consider the development of unconventional digital forensic technologies to improve the effectiveness of the whole investigation process as well as to increase the degree of the acceptance of the parties involved in the IoT forensic process.

Law enforcement agencies should carry out public awareness forums (using any reasonable medium) and educate the general public on the responsibilities they have to ensure they are safe online. Many IoT users fall prey to security scams because they are ignorant or negligent.

Whereas Machine Learning algorithms can be deemed resourceful in generating timely and accurate reports, it should be noted EU GDPR regulations state that the final decision on a person’s character should not be made solely relying on the automated process that violates the person’s interests.

Overall, it should be noted that using semi-automated decision-making process especially that of Machine Learning algorithms in profiling and surveillance is a sure bet of eliminating human elements that bring with them discrimination, stereotypes, and prejudices.

CHAPTER 3. AUTOMATED PROCESSES IN DIGITAL FORENSICS

This chapter reviews the increasing role of automated processes, AI, and ML in the field of digital forensics. It discusses how the digital revolution and the proliferation of connected devices have resulted in a vast amount of data, presenting both opportunities and challenges for law enforcement agencies. The chapter highlights the recommendations made by researchers to apply AI techniques in digital forensics and emphasises the potential of AI to transform the field by enabling more efficient analysis, detection, and prevention of cybercrimes. Specific areas where AI and ML have been used, such as malware analysis, image/video forensics, network forensics, and mobile forensics, are also examined.

The chapter also delves deeply into anomaly detection by examining its application domains and challenges, techniques, and algorithms. A review of the state-of-the-art anomaly detection algorithms is carried out.

The chapter concludes by proposing the best algorithms to be used for the analysis of the collected dataset for experimentation.

3.1 Introduction

The internet, mobile phones, and connected devices have significantly impacted daily life, generating vast amounts of data, including emails, contacts, videos, and photos (Qadir and Varol, 2020). Law enforcement agencies have adapted to this digital era, utilising digital intelligence through data mining and machine learning for crime analysis, focusing on detecting, predicting, and preventing criminal activities (Adam and Varol, 2020).

Over two decades ago, researchers like Mitchell (2010) advocated for applying artificial intelligence in digital forensics due to the increasing data volume. Milne (2012) stressed the importance of digital intelligence techniques for effective digital forensics investigations, involving cross-referencing and linking forensic data to support policing elements such as intelligence-led operations and resource allocation (Ćosić, Ćosić and Bača, 2012).

The authors Irons and Lallie (2014) proposed that artificial intelligence is crucial for overcoming digital forensic challenges, introducing the concept of "intelligent forensics" to identify and predict cybercrimes. This aligns with studies integrating AI and automation to improve digital forensics systems and case proceedings like Al Fahdi *et al.* (2016).

Qadir and Varol (2020) proposed a ML technique for behavioural forensics by using pattern recognition applications to handle large data sets in crime prevention

Toppireddy, Saini and Mahajan (2018) employed ML techniques for spatial analysis, creating a monitoring and prediction tool for crime network visualisation but lacking real-time data capture.

A study by Costantini, De Gasperis and Olivieri (2019) explored AI's integration into Digital Forensics, presenting AI as an enabling technology for evidence analysis. Their Decision Support System (DSS) tool, developed leveraging AI capabilities, aids in complex investigations. Xiao, Li and Xu (2019) applied AI for video-based evidence analysis, highlighting the importance of automation in digital forensics.

Magnet Forensics (2020) introduced Magnet AUTOMATE, an advanced digital forensic investigation tool based on a repeatable forensic workflow, designed to reduce case backlog and turnaround time for cybercrime cases.

Homem (2018) addressed automation challenges, proposing a resilient program using a Machine Learning-based triage method for remote evidence acquisition. This automated system assists forensic analysts in reducing the burden of evidence discovery and analysis, enabling faster resolution of critical cases and suspect identification (Iqbal *et al.*, 2018).

3.2 Areas where AI and ML have been used in Digital Forensics

Automated Digital Forensics processes are on the rise with the need to expedite the analysis phase and produce faster results to aid cybercrime cases in courts of law. AI-based applications are being utilised in the facilitation of the examination and analysing phases of the digital forensics process (Al Fahdi *et al.*, 2016). This technology enables forensic experts in examining and analysing digital evidence across a huge range of cybercrime such as, spyware, hacking, malware, data theft, and identity theft. However, as Butterfield *et al.* (2018) warns, with the advancements of the technology, so have the computer criminals advanced in their deployment of cybercrimes with more sophisticated crimes. This therefore calls for more advanced DF tools that are intelligent to handle these crimes.

Researchers have been drawn to this field of AI in digital forensics which has now yielded specialised sub-categories. The authors Kanimozhi and Jacob (2019) developed an AI-based Network Intrusion Detection System with an accuracy of 99.97% that makes use of the capabilities of ANNs in identifying intrusive traffic.

The literature below highlights various ways in which AI and ML have been used in DF through the integration of algorithms with computational methods.

Malware Analysis

Machine Learning has been suggested by Shalaginov *et al.* (2018) to be a promising tool for automation of the static and dynamic malware analysis process. The authors used different Machine Learning algorithms and datasets to analyse the static features of portable executable binaries. In their analysis they used VX Heaven and VirusShare datasets and found out that for malware analysis, the C4.5 and k-Neural Networks (k-NN) were the best algorithms.

In a similar fashion, the authors Sharma, Rama Krishna and Sahay (2019) utilised ML techniques to learn about the unknown malwares through the analysis of the occurrence of static characteristics. Among the ML algorithms used, the Logistic Model Tree (LMT) and Naïve Bayes Tree (NBT) gave the best results.

A research by Liu *et al.* (2017) has utilised a couple of both supervised and unsupervised ML algorithms for detecting malwares. This research made use of the available literature to develop a framework for automated malware analysis.

Machine Learning has also been proposed by Balram, Hsieh and McFall (2019) as a good technique for tackling new malware variants. A detailed model based on ML algorithm was proposed by Bijalwan (2020). The model detects Botnets and performs forensic analysis. Adversarial attacks have also been explored by Chen (2019) who used real samples from Comodo Cloud Security Centre to conduct an all-inclusive experiment.

Image/Video Forensics

An approach based on Deep Neural Network was designed by Szegedy, Toshev and Erhan (2013) to classify images for detecting objects.

A research by Shanableh (2013) highlighted the importance of videos having potential evidence which could be used in a court case. The authors demonstrated this by deploying a Machine Learning approach which detects any deletion process from video evidence to assess the authenticity of the video. The approach works through extraction of any discriminative features from videos images and bit streams that have been reconstructed. This approach is based on quantisation scales, intra-coded macroblocks, and the quality of reconstruction. The ML techniques are used to indicate the rates of True Positives or False Negatives.

The importance of double JPEG (Joint Photographic Expert Group) detection was mentioned by Chen, Shi and Su (2008). The authors emphasised this importance by proposing a scheme based on Machine Learning which could detect a double JPEG image compression by the use of the Markov random process. The technique of thresholding is applied to decrease the probability size of the transition matrices used for characterisation of the Markov random process features for double JPEG compression detection. Thereafter, the Support Vector Machine (SVM) method is executed for the classification process.

A research by (Platzer, Stuetz and Lindorfer, 2014) proposed an effective solution that detects nudity or pornography. This solution combines machine learning techniques with a novel skin detection approach which leverages on upgrading machine learning and introduction of other novel methods that aid in increasing the rate of detected images. This approach uses the positioning of skin areas and skin detection within a picture. The SVM algorithm is used to classify the images as either pornographic or non-pornographic.

The authors Saikia *et al.* (2017) introduced a Deep Learning (DL) approach which uses Region based Convolutional Neural Network (R-CNN) to automate the detection of objects in indoor environments. It relied mostly on images/video forensics. However, Nowroozi *et al.* (2021) warns about the adversarial image forensic by discussing the problematic structure and vulnerabilities associated with Machine Learning. They suggested the use of comparable solutions to avoid the attacks associated with this process. In the emergence of increasing child pornography, Anda *et al.* (2018) used facial features for forensic analysis to present an auto age estimation process. In their experiments, they used ANN, CNN, and SVM as the basic machine learning algorithms. A technique for detecting the past processing features of MOV, MP4. And £GP videos has been discussed by Sandoval Orozco *et al.* (2020) where the forensic artifacts from editing applications and social platform are used.

Network Forensics

A Deep Learning digital forensic framework was proposed by Karie, Kebande and Venter (2019) to handle huge cyber space data.

There is increasing spread of malicious Internet Protocol (IP) addresses, this is a big problem. Even though, the IP reputation system which in most cases handles this problem, this system has been deemed to be too expensive, consumes more time, and has high false positives. As a solution to overcome this hurdle, Usman *et al.* (2021) presented a Machine Learning based framework to handle network forensics.

An improved version of a network forensic framework called Particle Deep Framework (PDF) has been proposed by Koroniotis, Moustafa and Sitnikova (2020). The proposed framework identifies IoT network problems in three stages. In the initial stage, there is collection of data from the network, the framework then adopts Deep Learning features by using the Particle Swarm Optimisation (PSO), and finally, the abnormalities are traced through the use of Deep Learning Neural Network (DNN).

Mobile Forensics

The exponential growth and usage of mobile phones has resulted in a need to develop a mobile forensic field. Machine Learning algorithms of Decision Tree (Locally Weighted Learning (LWL) and Bayesian Network) have been utilised by Marturana *et al.* (2011) to automate the analysis phase of the mobile forensic. With the datasets used for a paedophile case study, it was found that the Decision Tree provides the most accurate results.

These advancements of automation in DF have helped forensic experts to find solutions to legal significance in less time and with some realistic costs. It can be argued to an extent that this automation can be used to limit future risks and problems through a thorough analysis of current and previous digital evidence (Jarrett and Choo, 2021).

According to Jarrett and Choo (2021), the likelihood of the occurrence of future cybercrimes and attacks may be addressed by the use of intelligent technologies and other computational methodologies. There is more need to first understand how these intelligent technologies have been deployed thus far. This includes and not limited to crime scene investigation, photographing and documentation of the crime scene, identification, collection, preservation, and analysis of the forensic evidence and other forensic processes (Franke and Srihari, 2008).

The authors Franke and Srihari (2008) highlight on the scope of digital forensic by stating that it (DF) goes beyond computer-related crimes and therefore includes computational methodologies for analysing the physical evidence found at the crime scene. This empirical evidence upon further review, Tanner and Dampier (2009) suggest that it (empirical evidence) may be used in determining the integrity of the digital evidence and its creditability by ensuring that it is not subjected to any form of alteration and/or modification.

3.3 Implications of Automated Process in Digital Forensics

The application of Artificial Intelligence in Digital Forensics has been studied by several researchers. A prior research by Dilek, Cakır and Aydın (2015) focussed more on the advances in the application of intelligent techniques in Digital Forensics. The authors reviewed the implementation of these intelligent techniques in defending against cybercrimes.

According to Jarrett and Choo (2021), there are two challenges faced in the automated digital forensics systems. Firstly, the automated tools and systems only serve to facilitate the investigation process, therefore, the process still requires an expert human investigator to provide oversight (Jarrett and Choo, 2021). Secondly, these authors further state that the accuracy of the forensic process relies more on the human abilities of the investigators, this is because some of these intelligent tools are still under development which means that they have some inaccuracies, incompleteness, and therefore may not provide the desired robust information for forensic cases. To overcome these challenges, it is required that either the investigators are offered relevant training and skills development, or alternatively, to make use of highly qualified investigators. Earlier, it was claimed by James and Gladyshev (2013) that in a possible scenario, practitioners who are inexperienced rely more on automated systems and therefore act on insufficient information. This, the authors (James and Gladyshev, 2013) continue and say that it leads to a high probability of failed investigations. In addition to this, in some instances, the digital forensic investigations are awarded to third party contractors, who may not be certified as they are self-proclaimed. This calls for a robust certification institution and regulatory bodies to ensure that only certified DF investigators are the only ones allowed to handle these investigations (James and Gladyshev, 2013).

Another challenge being faced in this regard is the use of variant and complicated media formats which prove difficult to acquire or be analysed by the available automated systems and tools (AlFahdi, Clarke and Furnell, 2014).

3.4 Anomaly Detection

Anomalies could basically be referred to as unusual data points that appear dissimilar from most of the whole data. Other terms used to mean anomaly are: outliers, abnormalities, and deviants.

These anomalies have been defined by Hawkins (1980) in his book ‘Identification of Outliers’ as some sort of outlier which can be observed and seen to differ significantly from the other observations thereby causing suspicion indicating that it has been generated by a different method.

Depending on the interests and needs of the user, anomaly detection techniques may be employed to execute these needs.

3.5 Application Domains for Anomaly Detection

As was explained by Aggarwal (2015), anomaly detection can be applied in instances like; intrusion detection, credit card fraud, medical diagnosis, law enforcement for crime detection, medical diagnosis among others.

There are key roles that Anomaly Detection can play in many different applications. For instance, in the databases, a variety of anomaly and outlier detection can be applied during the pre-processing step during data preparation. Other techniques can be used for building models that have the capabilities for anomaly detection in different scenarios. This section will explore some application domains for anomaly detection.

i) Intrusion Detection

Intrusion detection is a critical area of computer security where malicious activity is identified and countered. Various anomaly detection techniques are employed to achieve this, but a major challenge lies in efficiently analysing vast amounts of data. Fortunately, the abundance of data also allows models to learn regular system behaviour through semi-supervised learning (Chandola, Banerjee and Kumar, 2009).

Intrusion can occur on either the host (unauthorised system access) or network level (external attempts to infiltrate a network). In host intrusion detection, system calls are analysed to identify abnormal sequences, while network intrusion detection leverages network traffic data and metrics for anomaly detection.

Recent research has delved into leveraging the power of deep learning for intrusion detection. A study by Vanin *et al.* (2022) explored combining deep learning with data augmentation techniques (artificially generating more training data) to achieve high accuracy in identifying cyber threats within network traffic data. This approach capitalises on deep learning's ability to learn intricate patterns, leading to more effective intrusion detection systems. Furthermore, a study by Mohammad *et al.* (2024) highlights the increasing importance of Machine Learning techniques in Network Intrusion Detection Systems (NIDS). Researchers are actively investigating various ML algorithms to improve NIDS efficiency and accuracy. This study emphasises the importance of selecting appropriate datasets for training and evaluating these models.

ii) Fraud Detection

Anomaly detection of fraudulent activities (transactions) for credit cards, banks, and commercial companies are identified in this domain by use of anomaly detection techniques. Criminals who engage in identity theft can also be identified by use of anomaly detection techniques. A customer’s normal behaviours are maintained by the bank, abnormal activities that could be detected in such scenarios can be where a cash withdrawal from unusual locations. An alert will be sent by the anomaly detection technique. The challenge faced by this technique is the need to have a technique that is capable of quickly detecting the abnormal behaviour. Mostly, online techniques are preferred.

The auto-associative neural network (Aleskerov, Freisleben and Rao, 1997) was used to develop the CARDWATCH technique that detects fraudulent credit card transactions. Another neural network (Brause, Langsdorf and Hepp, 1999) was used to detect anomalies with low rate of false alarm. The application of unsupervised fraud detection based on clustering techniques was used on several credit card datasets (Bolton, Hand and H, 2001). The method that uses a back-propagation algorithm with Naïve Bayes model (Phua, Alahakoon and Lee, 2004) was developed to detect fraud on an automobile insurance company dataset.

However, recent advancements explore a broader range of techniques to combat increasingly sophisticated fraud attempts. Deep learning, a form of artificial intelligence, is now a major player. These powerful models, particularly Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), excel at finding hidden patterns in data. Several studies like Fu *et al.* (2016), Raghavan and Gayar (2019), Cheng *et al.* (2022), and Alfaiz and Fati (2022) have explored using these models for fraud detection in credit card transactions by analysing sequences of transactions, identifying subtle anomalies that might indicate fraudulent activity, such as a sudden surge in purchases or transactions originating from unusual locations.

For fraud involving networks, like money laundering or telecom scams, Graph Neural Networks (GNNs) are a game-changer. These models analyse relationships within a network to detect suspicious connections. A recent study by Li *et al.* (2023) explored the use of GNNs to identify fraudulent rings in telecommunication networks. Their model analysed call data records, uncovering hidden connections between phone numbers involved in suspicious activity. This allowed them to identify and dismantle the entire fraudulent network.

Unsupervised anomaly detection remains valuable. Isolation Forests, a recent technique, efficiently identifies outlier transactions that deviate from normal spending patterns. For example, a study by Rajeev and Devi (2022) applied Isolation Forests to credit card fraud detection. Their approach effectively isolated transactions that fell outside the typical spending habits of a cardholder, allowing investigators to focus on those with a higher likelihood of being fraudulent.

iii) Health and Medical

Anomaly detection techniques can be used in the public health and medicine to purposes such as detecting mistakes like missed recordings. The data in this domain pertains to patient’s records of different types like name, age, blood type, weight, condition, address, among others. The characteristics exhibited by this data are of spatial and temporal. The anomaly detection techniques in this domain deal with point anomalies and the approaches applied are the unsupervised ones due to the availability of patient records.

It is however noted that a mistake in identifying an anomaly in this domain could be of dire consequences due to the nature/sensitivity of the subject. Bayesian networks were used to detect outbreaks of diseases (Wong *et al.*, 2003). Statistical techniques were used to detect anomalies in medical laboratory reference data as a pre-processing step (Solberg and Lahti, 2005). Another approach by Suzuki *et al.* (2003) used probabilistic mixture model to visualise outliers in medical test data.

Recent advancements in anomaly detection are further enhancing healthcare efficiency and accuracy. Researchers have explored utilising RNNs for anomaly detection in electronic health records (EHRs). RNNs excel at handling sequential data, making them suitable for analysing patient medical histories and identifying potential inconsistencies (Brown *et al.*, 2018). Furthermore, a 2023 study investigated the application of Generative Adversarial Networks (GANs) for anomaly detection in medical images. GANs can learn the underlying distribution of normal data, allowing them to effectively detect abnormalities within medical scans (Esmaeili *et al.*, 2023).

iv) Image and Video

In the image and video domain, the anomaly detection techniques are used to identify changes in still images, stream of images, and video clips. The sub-domains categorised under this domain are handwriting recognition, video surveillance, satellite imagery, spectroscopy, audio analysis, among others. One of the major setbacks in this domain is the high dimensionality coupled with the sheer amount of data points. Pokrajac, Lazarevic and Latecki (2007) proposed incremental Local Outlier Factor

(LOF) as an extension to the LOF algorithm to detect anomalies in data streams. This technique was evaluated on a dataset of video clips. To classify anomalous regions of images, Singh and Markou (2004) proposed a framework that uses neural networks as classifiers. A machine learning algorithm proposed by Davy and Godsill (2002) was based on SVMs to find sudden changes in audio systems. And finally, a regression model proposed by Da *et al.* (2005).

Deep learning has revolutionised anomaly detection in images and videos. CNNs excel at extracting features from image and video data. A 2019 study proposed a CNN-based approach for anomaly detection in crowd scenes, achieving high accuracy in identifying unusual activities (Saqib *et al.*, 2019). Furthermore, a 2023 study explored employing Autoencoders, a type of deep neural network, for anomaly detection in video surveillance. Autoencoders can reconstruct normal video data, allowing them to flag deviations as anomalies (Pavuluri and Annem, 2023).

v) Textual Data

Anomaly detection in this domain relates to detecting emerging stories and news. This could be analysing the twitter traffic to detect breaking news. In the past, studies like one by Miller *et al.* (2014) used clustering to detect spam within Twitter data. However, recent advancements are pushing the boundaries with the help of machine learning and deep learning.

Deep learning offers another exciting avenue. Techniques like RNNs are particularly adept at handling sequential data, making them ideal for analysing textual content. A 2023 study investigated employing Long Short-Term Memory (LSTM) networks, a type of RNN, for anomaly detection in news articles. LSTMs can learn long-term relationships within text data, allowing them to identify unusual writing styles or topics that might indicate anomalies (AGGARWAL, 2023).

vi) Wireless Sensor Networks

In the wireless sensor network domain, the anomaly detection uses readings from sensors that are distributed across a network to detect intrusion or identify faulty sensors. The data here could be in the form of several numerical discrete or continuous form, video, or audio. This data contains a fair amount of noise making it more difficult to detect anomalies. Bayesian Belief Networks (BNNs) were used by

Janakiram, Reddy V and Kumar (2006) to detect spatial and temporal anomalies in the sensors streaming data. To detect security attacks in wireless sensor networks, Phuong *et al.* (2006) proposed a statistical anomaly detection method. (Branch *et al.*, 2013) proposed a rule-based algorithm to detect anomalies in wireless sensor networks. A technique proposed by Idé, Papadimitriou and Vlachos (2007) was based on nearest neighbours. The technique detects changes in correlated streams of sensors.

Recent advancements are pushing the boundaries of anomaly detection in WSNs. Deep learning techniques are showing promise for WSNs with image data. For instance, a study by Xu and Lin (2023) explored using Convolutional Neural Networks for anomaly detection. CNNs excel at recognising patterns in images, making them suitable for tasks like identifying unusual objects or activities captured by sensor cameras.

3.6 Challenges of Anomaly Detection

As it has been shown in the definition of an anomaly as rare data points within a majority of the data, it therefore follows that anomaly detection can be abstractly considered as an exercise of identifying patterns or data points that lie outside of the normal region. This can therefore be done by identifying the normal region and thereby flagging anything that falls outside of that normal region as an anomaly. This may seem easy, but as shown by Chandola, Banerjee and Kumar (2009), there are problems associated with anomaly detection as highlighted below:

i) Differentiating and defining what is normal and abnormal may not be easy.

ii) The changing behaviour of what is normal is ever evolving and being different across different domains.

iii) Anomalies take up different forms and types depending on the domain within which they reside.

iv) Availability of representative datasets with or without labels.

v) Differentiating true anomalies from noisy data.

Chandola, Banerjee and Kumar (2009) further state that the majority of existing anomaly detection techniques have been developed to solve specific domain problems and therefore, may not be applied generally to other domains.

3.7 Categories of Anomalies

i) Point Anomalies

This is the case where anomalous data points are considered to be so different from the rest of the data. This is explained in *Figure 3.1* in which the regions N_1 and N_2 are regarded as normal since most of the data points are in these two regions. In contrast, O_1, O_2, and O_3 are considered anomalies because they are way too far from the normal regions.

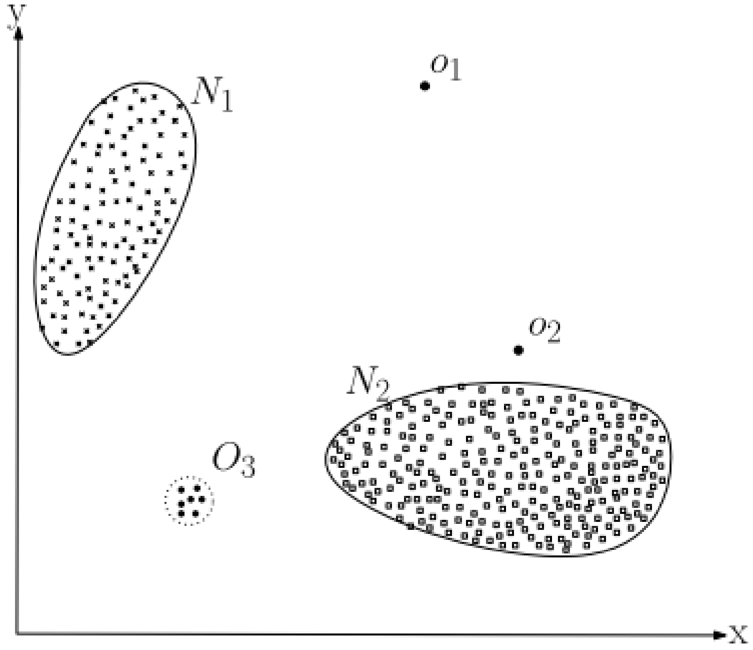


Figure 3.1 Point anomalies in a two-dimensional space (Chandola, Banerjee and Kumar, 2009)

ii) Contextual Anomalies

This is where the context of the data is anomalous and not the data point itself. From *Figure 3.2*, even though the data point t_1 is the same as the data point t_2, the latter point (t_2) is regarded as anomalous because it appears in an anomalous context.

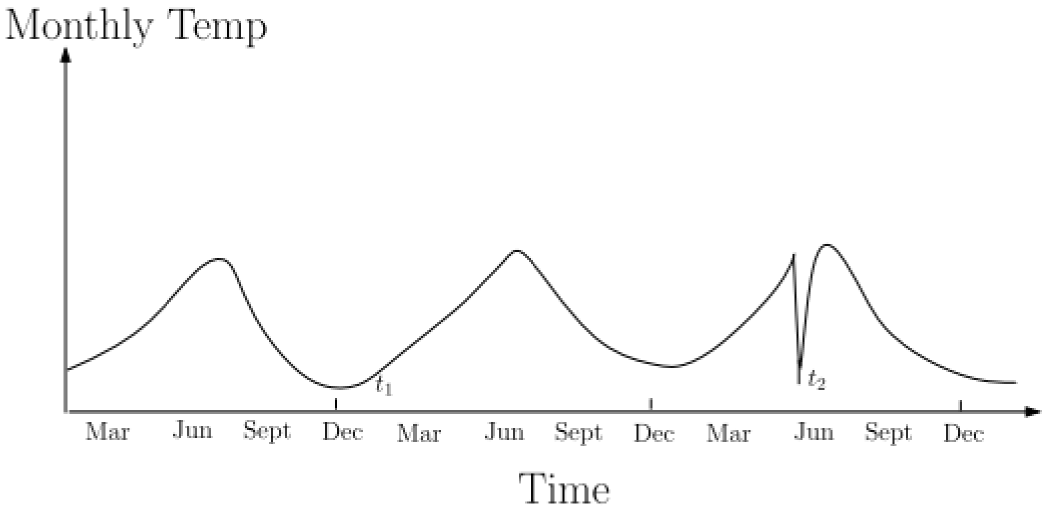


Figure 3.2 Contextual anomalies in a monthly temperature data (Chandola, Banerjee and Kumar, 2009)

iii) Collective Anomalies

This is where the collection of data points is regarded anomalous because of the collection of these data points together rather than the data points on their own. In *Figure 3.3*, the data points in the electrocardiogram are regarded as anomalous because their appearance as a collection in this data and not because of the data points themselves.

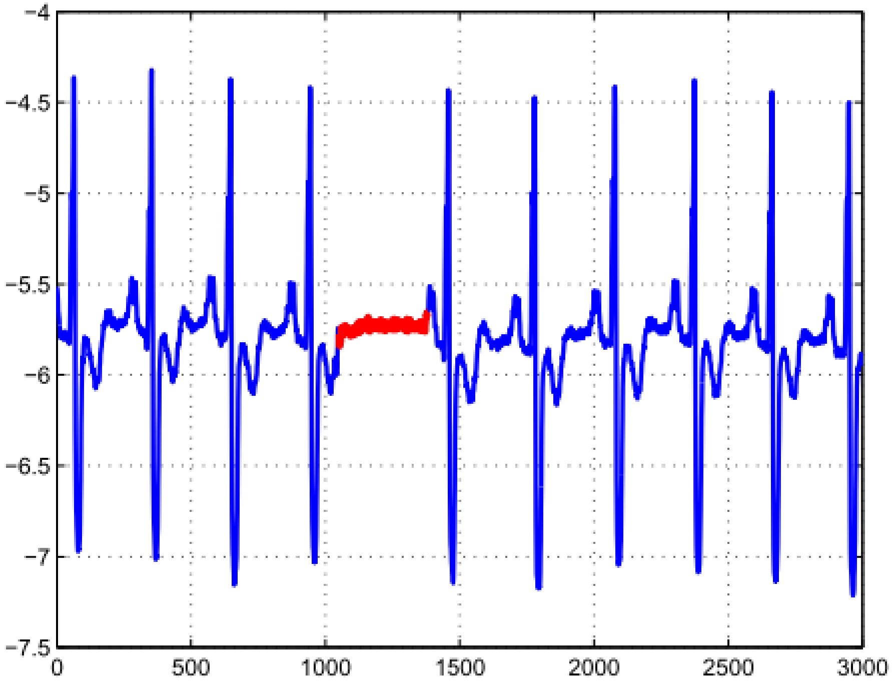


Figure 3.3 Collection anomaly in a human electrocardiogram (Chandola, Banerjee and Kumar, 2009)

The availability of data labels is a very important aspect in anomaly detection. To this end, the categorisation of anomaly detection techniques (Alshammari, 2018b) can be regarded with the availability of data labels as:

a) Supervised

In this category, it is assumed that there is a training dataset with labels that identify normal and abnormal data points. This is illustrated in *Figure 3.4*.

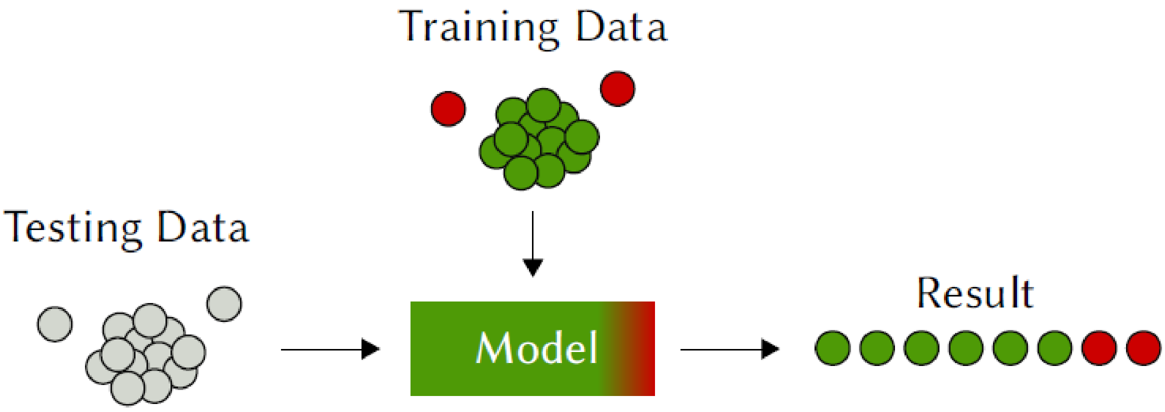


Figure 3.4 Supervised Anomaly Detection (Goldstein and Uchida, 2016)

b) Semi-Supervised

Here, the assumption is that the provided training data primarily consists of normal instances, and any deviation from the patterns established by these normal data points is considered an anomaly. This is illustrated in *Figure 3.5*.

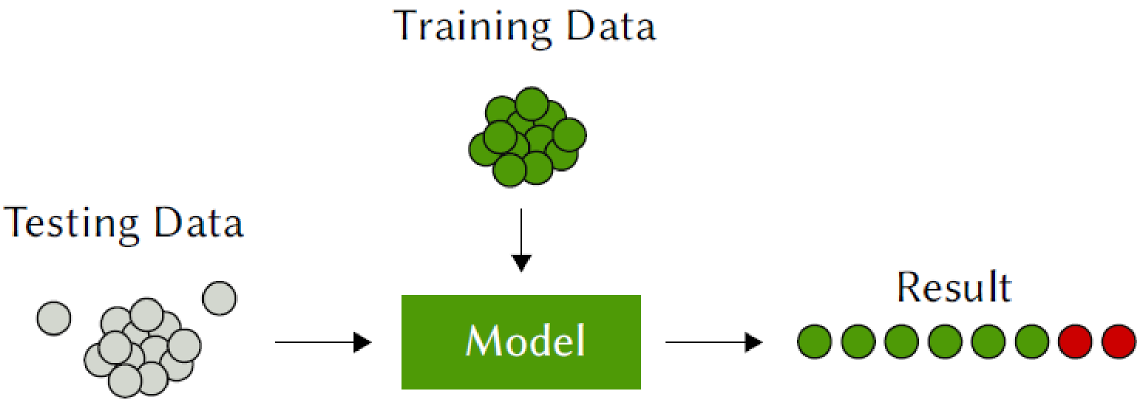


Figure 3.5 Semi-Supervised Anomaly Detection (Goldstein and Uchida, 2016)

c) *Unsupervised*

In this category, it is assumed that the available data is not labelled, and no training data is required. It is further assumed that the majority of the data points are normal and therefore, the technique groups or clusters the data points into cluster and any isolated points are considered anomalies. This is illustrated in *Figure 3.6*

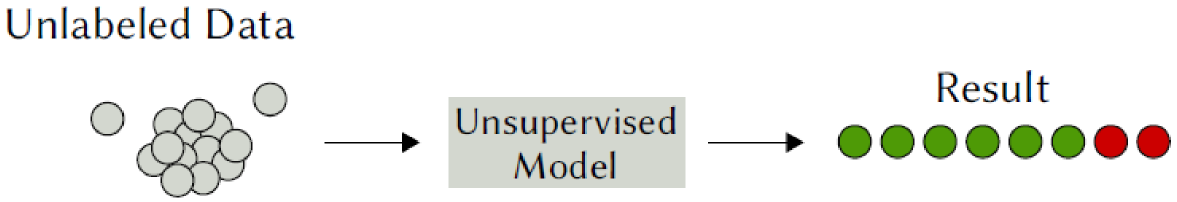


Figure 3.6 Unsupervised Anomaly Detection (Goldstein and Uchida, 2016)

3.8 Anomaly Detection Techniques

The discussion in this section will be about the several approaches and techniques used to detect anomalies. The categorisation factor used will be based on the fundamental technique employed by each approach, namely, classification, nearest neighbours, clustering, statistical, and spectral.

i) *Classification*

The techniques used in classification need labelled data from which the system can learn. The logic for classification is to train a classifier on the normal data points and thereafter an evaluation of the accuracy of the model on unidentified data points, called the testing data points. Depending on how many classes can be learned, the classification technique can be further divided into two categories namely: ***one-class*** and ***multi-class*** anomaly detection techniques.

The one-class technique takes the assumption that the training data points are all normal. Therefore, any data point that falls outside of that normal class will be classified by the model as an anomaly. In *Figure 3.7*, the one-class model groups all normal data points as a one big class and any other points residing outside this class are flagged as anomalies.

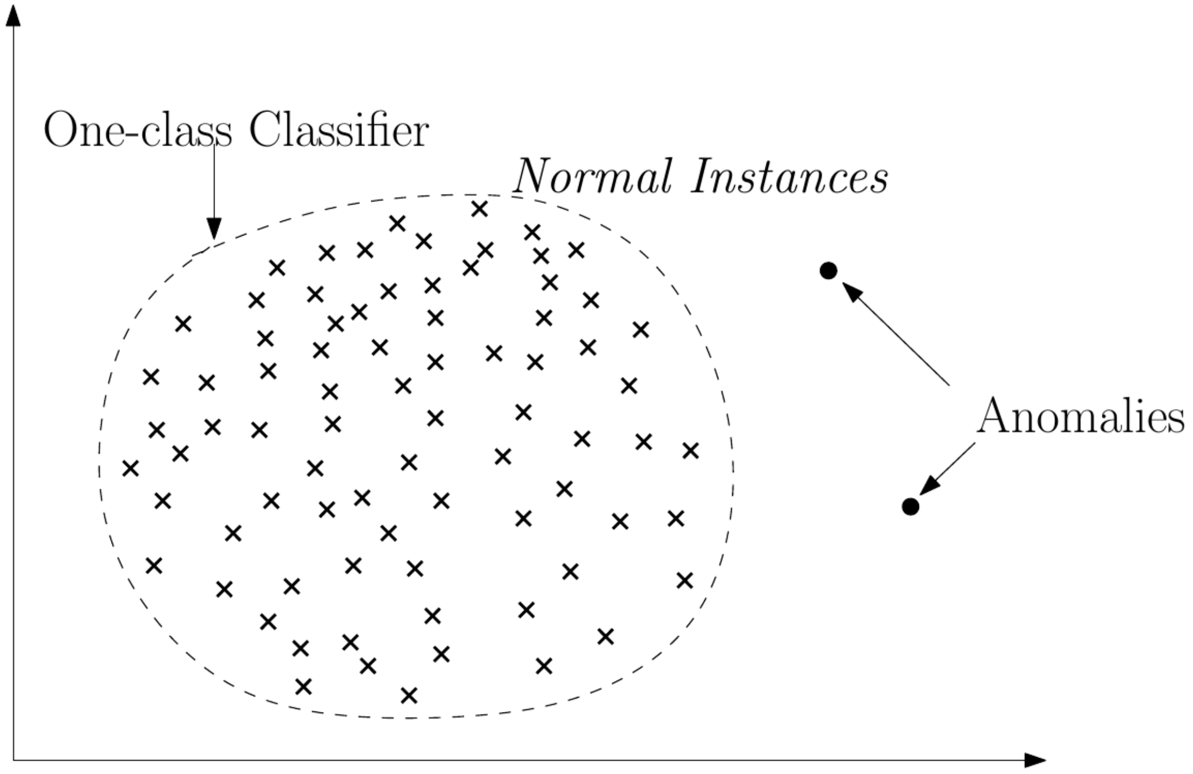


Figure 3.7 One-class Classification (Chandola, Banerjee and Kumar, 2009)

The multi-class category is similar to the one-class category except that multi-class can learn multiple regions of the normal data as illustrated in *Figure 3.8*.

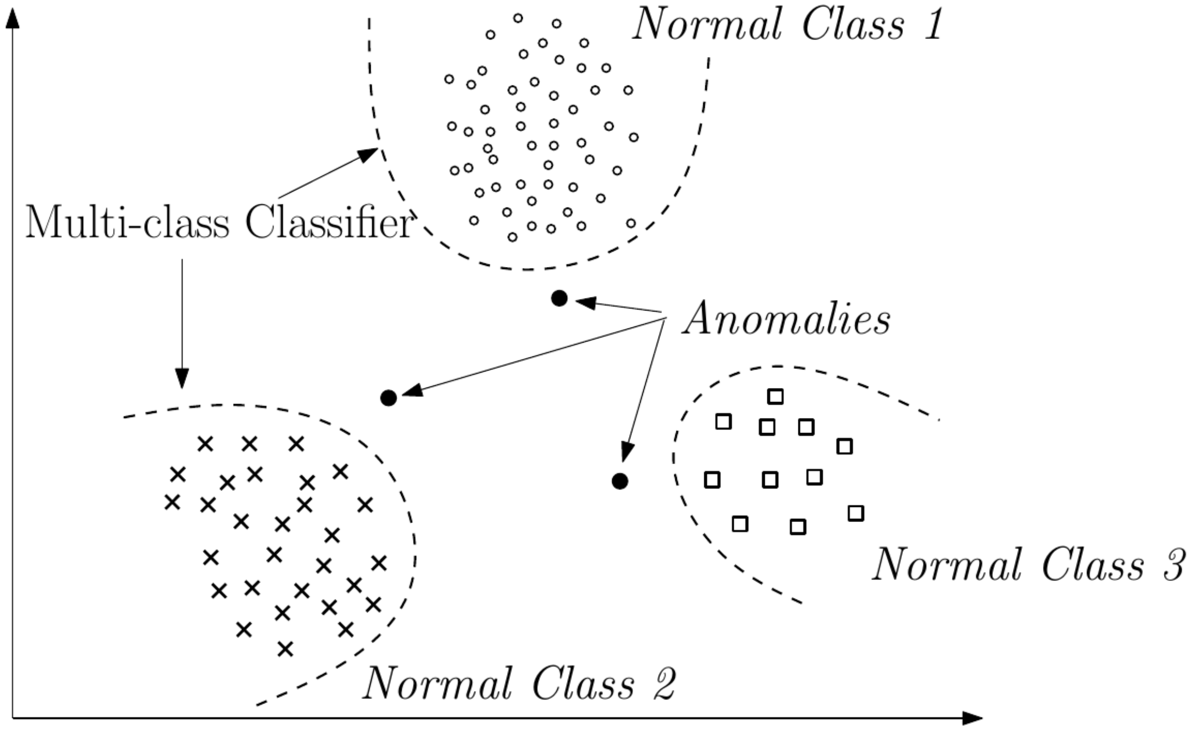


Figure 3.8 Multi-class Classification (Chandola, Banerjee and Kumar, 2009)

The classification techniques can be organised into several categories based on the algorithm used by the model. These are Support Vector Machines (SVM), Artificial Neural Networks (ANN), Bayesian Networks, and rules.

The SVM is capable of learning the normal region from the training data samples and with kernel trick, the model is able to learn non-linear regions in hyperplanes (Vapnik, 2000) and (Boser, Guyon and Vapnik, 1992).

The ANN can be applied to one-class and multi-class classification problems. ANN trains a neural network model on a portion of the dataset. The testing data points are then fed to the network and when the network rejects the data point, it is flagged as an anomalous data point (De Stefano, Sansone and Vento, 2000).

The Bayesian Networks based techniques can be applied in multi-class classifications. The idea behind the use of the Bayesian Networks is to estimate the prior probabilities whereby a testing set of data points will be fed to the network and for each data point, the data point with the biggest posterior score will be the class or the label. Intrusion in computer networks have been detected by use of Bayesian Networks (Sebyala, Olukemi and Sacks, 2002), (Barbará, Wu and Jajodia, 2001), and (Valdes and Skinner, 2000).

Anomaly detection techniques that are based on rules, learn from normal data points and any data point that does not conform to any learned rule is consequently regarded as an anomaly. The main idea in this technique is to train a model on a training set so the model can derive rules. Every rule is assigned a confidence score, a ratio of the correctly classified data points by this rule. During the testing phase, the model will search for the best matching rule for each testing data point. The anomaly score for each testing data point will be the inverse of the confidence score of the best matching rule.

From these discussions on the classification techniques, several advantages have been brought out and more so is the abundance of models of classification that give this research the flexibility to choose an appropriate model to solve the problem.

On the contrary, these approaches also do possess several shortcomings. One of the shortfalls is the difficulty to find (in some domains) training datasets that are correctly annotated.

ii) Nearest Neighbours

The techniques for anomaly detection in this category rely on distance measure between two data points which are computed and based on the distance, the data points are organised in neighbourhoods to get an understanding of the structure of the dataset. Methods based on distance calculate an anomaly score by assessing the distance between data points and their neighbouring points (Cai *et al.*, 2023). The distance measures used depend on the type of the variables or the feature space. For numerical data points, the Euclidean distance is usually used (Tan and Steinbach, 2006), whereas for categorical features, Jaccard distance can be used in addition to other available methods (Küffner *et al.*, 2010). The algorithms within the nearest neighbour category can be divided further into two sub-categories: K^{th} nearest neighbour algorithm and density of the data points algorithm.

The K^{th} nearest neighbour distance can be used as an anomaly score for a collection of data points. As an illustration, k-nearest neighbours (k-NN) determine the anomaly score of an instance by considering the distance to its K^{th} nearest neighbour. Meanwhile, distance-to-measure (DTM) introduces a fresh distance metric derived from the distances of the initial k-nearest neighbours (Gu, Akoglu and Rinaldo, 2019). Additionally, the local outlier factor (LOF) calculates the anomaly score by assessing the deviation of the instance from the local density of its neighbouring data points.

The density-based techniques measure the density of the data points neighbourhoods. A data point is flagged as an anomaly if it resides in a low-density neighbourhood. The techniques in this category rely on having close to uniform densities for the data points.

iii) Clustering

Anomaly techniques that are clustering based carry the same similarities to the nearest neighbours' techniques. The nearest neighbours' techniques perform the calculation between a data point and its local nearest neighbour while the clustering-based techniques perform the calculations between each data point and the group or the cluster that it belongs to based on the similarity measure. The clustering-based techniques can employ both the unsupervised and semi-supervised methods. The clustering family assumes that the normal data has a cluster, therefore, if any data point is outside of this cluster is flagged as an anomalous data point. The clustering algorithms' main objective is to find structures in datasets. For this reason, they are used for performing exploratory analysis of datasets and are used in recommender systems. Of these clustering techniques, some allow for data points to reside outside the cluster they build. The algorithms that allow for this condition can be used for anomaly detection (Alshammari *et al.*, 2018a). Examples of these techniques are Density-Based Spatial Clustering of Applications with Noise (DBSCAN) proposed by (Ester *et al.*, 1996), Spiking Neural Network (SNN) by (Ertöz, Steinbach and Kumar, 2004), and ROCK by (Guha, Rastogi and Shim, 2000). The output of these models is binary because of the nature of how these algorithms work.

Another family of the clustering techniques performs by assuming that normal data points are organised around the centre of the cluster (centroid). In this case, the anomalous data points are those that are far away from the centre. The general procedure of these techniques starts by using a clustering algorithm to group the data points. Then for every data point, the distance from the data point to the cluster centroid is defined as the anomaly score. The algorithms that have successfully achieved this goal are for example; K-means and Expectation Maximisation (EM) algorithms by (Smith *et al.*, 2002), and Self-Organising Maps (SOMs) by (Kohonen, 1995). Recent advancements in this family include incorporating deep learning architectures into centroid-based anomaly detection. Studies like Chen *et al.* (2020) proposed a Convolutional Variational Autoencoder (CVAE) for anomaly detection, achieving superior performance on complex, high-dimensional data.

The third family of the clustering techniques employs an assumption that in the dense clusters is where the normal data resides while the low-density cluster groups are where the anomalous data points reside. They are algorithms that follow this assumption, and these are for example; the Cluster- Based Local Outlier Factor (CBLOF) which was proposed by (Pires and Santos-Pereira, 2005), (He, Xu and Deng, 2003), and (Jiang, Tseng and Su, 2001). The improvement to this algorithm has been seen in literature where extensions to the algorithms have been developed (Sun and Chawla, 2004).

Current research on density-based anomaly detection focuses on handling complex data structures and improving interpretability. For instance, Scitovski and Sabo (2020) proposed a method called DBSCAN* that efficiently handles data with high dimensionality and noise.

Like the nearest neighbours' techniques, clustering based techniques have advantages such as being able to operate in both unsupervised and semi-supervised fashions. Another advantage is that the clustering-based techniques can work with many data types. As in nearest neighbours' techniques, the performance of clustering-based techniques relies on the performance of the clustering algorithm used. Consequently, the computation complexity of this technique could be high. It is also noted that some of the clustering techniques are not mainly geared towards anomaly detection but they are extended to perform anomaly detection which can make the performance of these approaches less than optimal (Alshammari, 2018b).

iv) Statistical

Statistical techniques try to fit a statistical model on the normal data point distribution and thereafter, to identify whether a data point is normal or anomalous, statistical tests are used. There are two divisions that categorise the statistical techniques for anomaly detection problems, these are; parametric and non-parametric techniques (Alshammari, 2018b).

Parametric techniques assume the existence of a distribution and its parameters can be learned from the data points. The distribution parameter is estimated from the training data points and the probability density function for any given data point. The anomaly score is defined as the inverse of the probability density function. This category can be classified based on the distribution model used. One of the popular distributions is the Gaussian model. The parameter for this model can be calculated by the Maximum Likelihood Estimates (MLE) and the anomaly score is defined as the distance of the data point from the distribution mean. When the model is defined, simple thresholds can be applied to filter out the normal data from the anomalous data.

Another category of parametric techniques uses a regression model to detect anomalies. These techniques are usually used in time-series analysis (Abraham and Chuang, 1989). Based on the training data points, these parametric techniques fit a regression model. The anomaly score for a data point is calculated by how far this testing data point is from the regression model. A robust regression model proposed by Rousseeuw and Leroy (2005) deals with anomalies present in training dataset.

In non-parametric techniques, there is no assumption of the existence of distribution of data points, rather, the distribution of the data is derived from the data points. These techniques can be divided into histogram and kernel-based function techniques. For the histogram approaches, a histogram is generated from the training data points. Then for the testing data points, a test is performed to determine if the data point lies in one of the histogram bins or not. If it does, then it is flagged as a normal data point. Otherwise, it is flagged as an anomalous data point.

Kernel based techniques use kernel functions to estimate the density of the dataset. The techniques seem to resemble the parametric techniques ones, only that there is a difference of how the density function is calculated.

In general, statistical approaches have several advantages (Alshammari, 2018b). Once the dataset distribution is known, it gives room for a number of available algorithms and options to perform anomaly detection. As the output of the anomaly detection models is a scalar value, it allows for more sophisticated approaches to be carried out based on the score. The techniques here can operate in both supervised and semi-supervised fashions. There is also a possibility of operation in unsupervised fashion for those techniques that can deal with anomalies in the training datasets.

The main disadvantage of statistical approaches is the reliance on the existence of a distribution of the datasets which is often not the case in real-world data. Finding the appropriate statistical test can be a challenge, even if there is a known distribution of the data points. Multivariate datasets can be a difficult task especially for histogram-based techniques.

v) Spectral

Spectral techniques or subspace anomaly detection techniques try to capture a meaningful representation of the data points by reducing the dimensions of the datasets to lower dimensions that could reveal structures not visible in the original form of the dataset. The reduction step is also referred to as embedding or projecting the data point to lower dimensions. These techniques work in an unsupervised fashion and can be used in conjunction with other models. They can also be used to perform pre-processing for the data points before feeding the data to the model.

One of the main advantages of using spectral techniques is their ability to handle high dimensional datasets. Reducing the feature space to lower dimensions makes it easier for the model to learn the characteristics of the data. Thus, spectral techniques can be used with other models that cannot handle high dimensionality. The ability for the spectral techniques to operate in unsupervised fashion is also another advantage point. Conversely, these techniques can only perform well if the data is separable when projected to lower dimensions. Another disadvantage is that they can be computationally costly especially when dealing with big datasets.

3.9 Unsupervised Anomaly Detection Algorithms

As explained by Alshammari *et al.* (2017), the main objective of unsupervised anomaly detection is to organise the data point into clusters or groups in a manner that enables the algorithms to detect data points that deviate from the normal clusters. The reason why the operation is unsupervised is that the data points fed to the model have no target labels from which to learn and draw associations. Therefore, the models are completely driven by the data points they receive (Alshammari, 2018b).

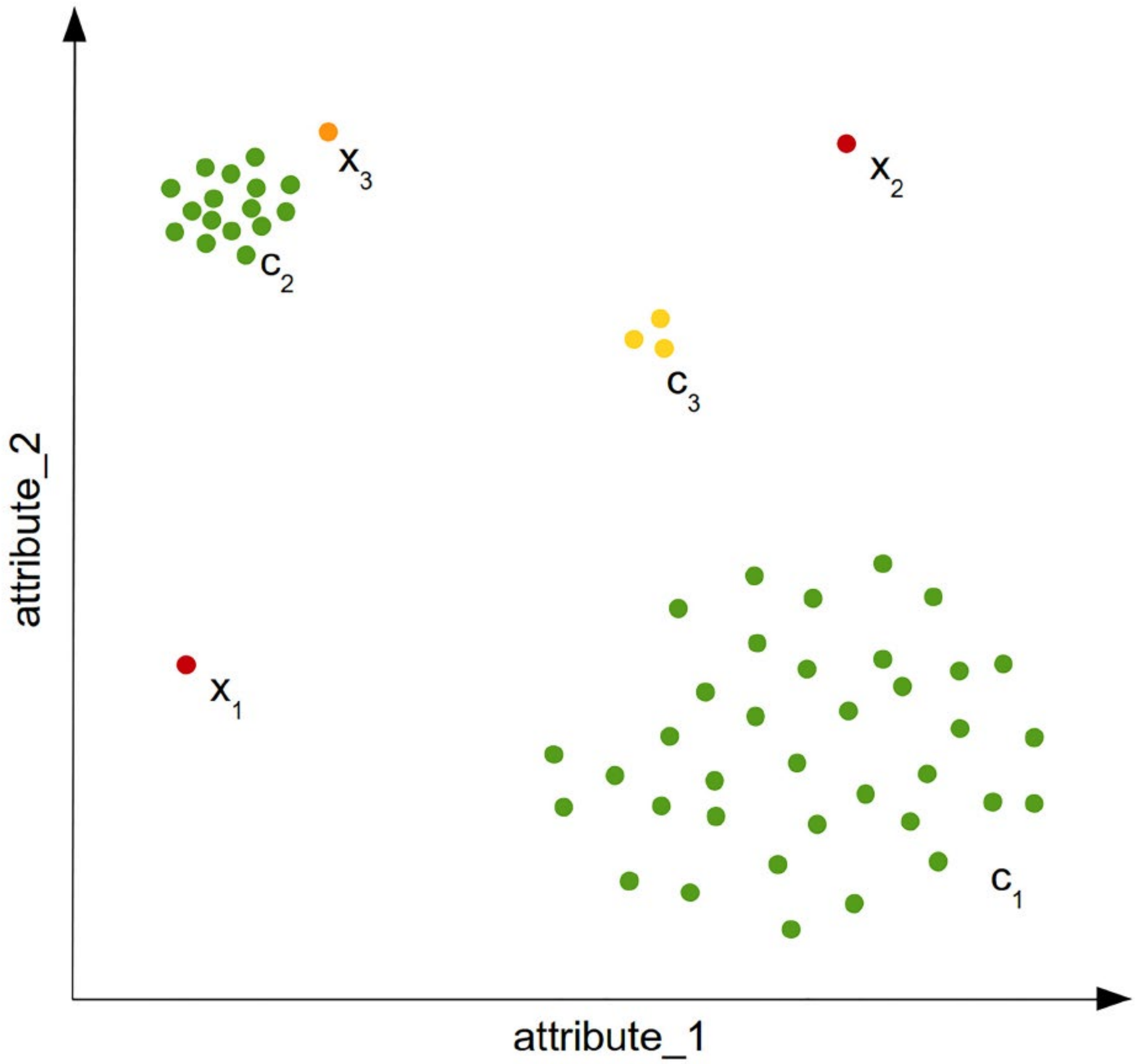


Figure 3.9 The global anomalies x1, x2 and the local anomaly x3 (Goldstein and Uchida, 2016)

Figure 3.9 illustrates and explores the notion of global and local anomalies. The clear anomalies are x_1 and x_2. Techniques for global anomaly detection should be able to identify these points as such. However, the data point $x_{3,}$ being too close to the cluster c_2 cluster, will cause an issue to global techniques as it is likely to be mislabelled as a normal data point. Therefore, the algorithm is said to be a global anomaly detection algorithm if it approaches the anomalies in a dataset in a global manner. In the event that the algorithm approaches the clusters individually, the data point x_3 is likely to be flagged as an anomaly and hence the algorithm will be referred to as a local anomaly detection algorithm. Something to take note though is that data point c_3 lies in a grey area and could likely pose difficulties to anomaly detection algorithms. Should it be flagged as an anomaly or a normal cluster? This is part of the challenge that anomaly detection faces and usually, to determine to which class this cluster belongs, a domain human expert opinion is needed (Alshammari, 2018b).

The unsupervised group of algorithms can be categorised as shown in *Figure 3.10*.

Nearest Neighbour		Clusters		Statistical	Spectral
Global	Local	Global	Local	HBOS	rPCA CMGOS
k-NN	LOF	CBLOF	LDCOF		
	COF	uCBLOF	CMGOS		
	INFLO				
	LoOP				
	LOCI				
	aLOCI				

Figure 3.10 Categorisation of unsupervised anomaly detection algorithms

i) k-Nearest Neighbour (k-NN)

The global anomaly detection algorithm is the k-NN, however, as exhibited in *Figure 3.9*, the k-NN algorithm could face difficulties detecting local anomalies. The k-NN algorithm group can be split into two classifications: *k*th-nearest neighbours and *k*-nearest neighbours. The *k*th-nearest neighbours defines an anomaly score by calculating for each data point, the distance to the *k*th-nearest neighbours (Ramaswamy, Rastogi and Shim, 2000). The *k*-nearest neighbours on the other hand, defines an anomaly score by calculating the average distance of the *k*-nearest neighbours (Angiulli and Pizzuti, 2002).

Figure 3.11 displays the results when k-nearest neighbour algorithm is applied on a sample artificial dataset. The anomalous data points are represented by the red data points, and their radius corresponds to the anomaly score they got. The *k* parameter must be set before running the algorithm. In this case (the example used in *Figure 3.11*), *k*=10. For data points closer to the green clusters, it can evidently be seen how the algorithm assigns low anomaly scores to them.

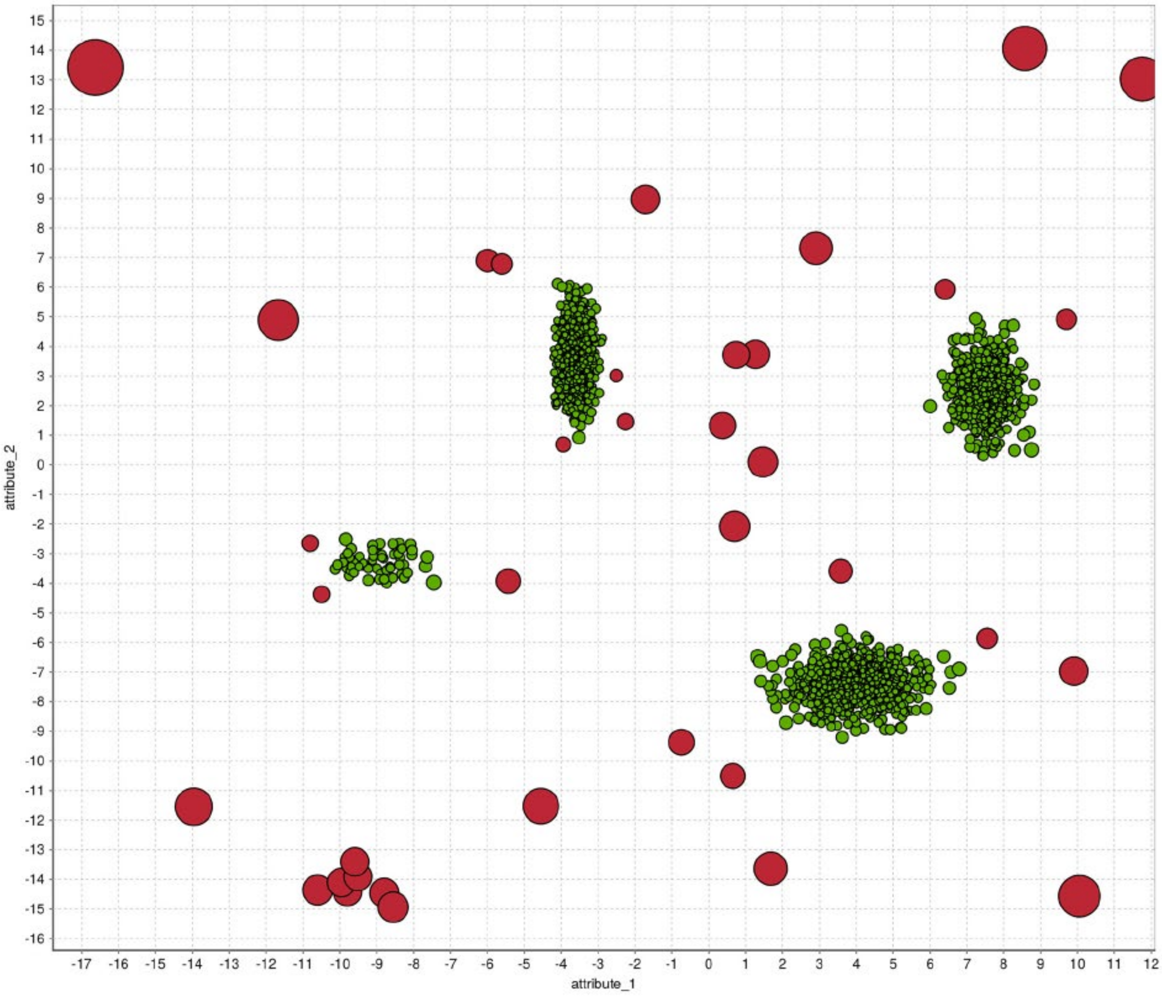


Figure 3.11 The k-nearest neighbour anomaly scoring of an artificial sample dataset (Goldstein and Uchida, 2016)

ii) Local Outlier Factor

The Local Outlier Factor (LOF) has been described by (Breuniq *et al.*, 2000) as one of the popular local anomaly detection techniques from which many other improvements and extensions have emerged. The LOF algorithm can be summarised in the following three steps:

1. Calculating the *k*-NN for every data point
2. Calculating the local density based on the previous *k*-NN scores (N_k) by using the local reachability density (LRD) function for a data point *x* and an object *o*:

$$LRD_k(x) = 1/\left(\frac{\sum_{0 \in N_k(x)} d_k(x,o)}{|N_k(x)|}\right)$$ *Equation 3.1 LRD Function*

3. Calculating the LOF score by comparing the LRD function of a data point with the LRD of its *k*-nearest neighbours.

$$LOF\ (x) = \frac{\sum_{0 \in N_k(x)} \frac{LRD_k(o)}{LRD_k(x)}}{|N_k(x)|}$$ *Equation 3.2 LOF score*

In simple terms, the ratio of the local densities is the LOF score. Therefore, normal data points will have densities similar to their local densities and the calculated anomaly score will be 1.0. The anomalous data points will get much larger score depending on how different the data point density is from its neighbours.

iii) Connectivity-Based Outlier Factor

The authors Tang *et al.* (2002) state that Connectivity-Based Outlier Factor (COF) is similar to LOF only that there is a difference in how the density is calculated. Whereas LOF calculates Euclidean distances using hypersphere centred on a data point, COF calculates the distance in an incremental manner by finding the shortest paths between data points.

iv) Influenced Outlierness

The INFluenced Outlierness (INFLO) was proposed by Jin *et al.* (2006) and is an extension to the LOF, it (INFLO) solves a challenge of LOF where there are two clusters of different density close to each other. The data points are mislabelled by LOF at the edges of the adjacent clusters. This issue is overcome by INFLO by incorporating a reverse nearest neighbours set of data points. *Figure 3.12* illustrates this by showing two clusters of varying densities. The LOF has flagged the red data points as anomaly because in the hypersphere (the grey circle) there are 5 nearest neighbours which have high local density. INFLO will incorporate the reverse nearest neighbours set of data points (the blue data points) which will make it less likely for INFLO to consider the red point as an anomaly.

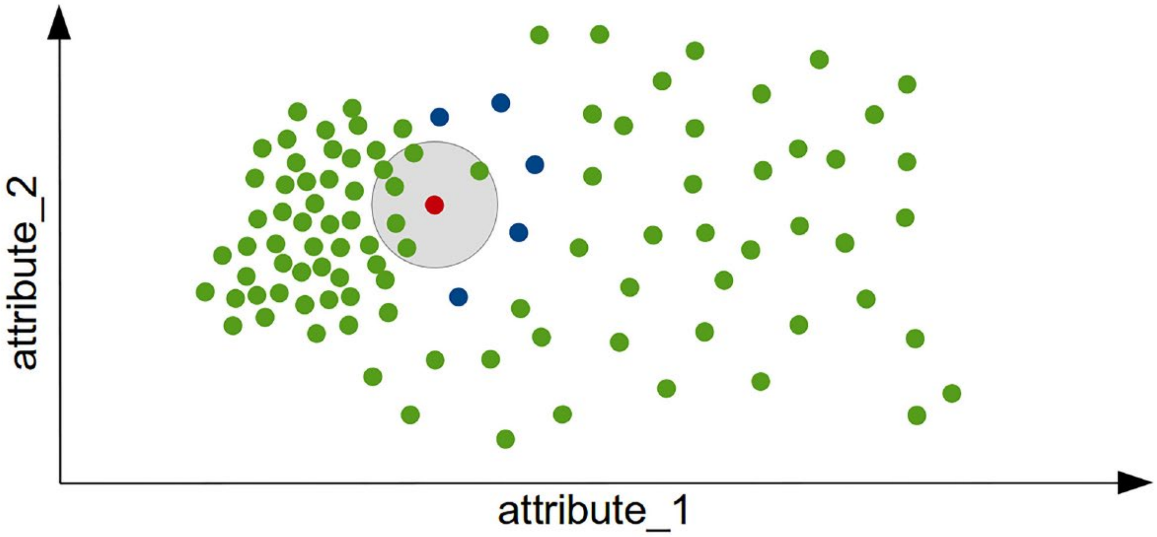


Figure 3.12 The INFLO algorithm compared to the LOF algorithm (Goldstein and Uchida, 2016)

i) Local Outlier Probability

Kriegel *et al.* (2009) proposed the Local Outlier Probability (LoOP) to tackle the interpretation of the anomaly score of the prior algorithms. As has been evident in the literature, some anomaly detection algorithms have binary output which could be a limiting factor in some applications. Other algorithms have scalar outputs value which measures how anomalous a data point is. This anomaly score can take arbitrary values depending on the data points of the dataset. This can values can make it hard to interpret the output of the algorithm (Alshammari, 2018b). LoOP tries to solve this problem by producing a probability score of how anomalous a data point is.

ii) Local Correlation Integral

The Local Correlation Integral (LOCI) is an improvement of the previous algorithms discussed so far. Developed by Papadimitriou *et al.* (2003), its main improvement is provision of a way to estimate a good value for the key parameter *k*. The algorithm delivers the best *k* score by iterating varying values of *k* for each data point and the maximum score is taken for the corresponding *k.* However, this approach is deemed computationally expensive (Alshammari, 2018b). Whereas in normal situations the *k*-NN approaches have $O(N^2)$ complexity, the complexity in LOCI can reach $O(N^3)$.

iii) Approximate Local Correlation Integral

The approximate Local Correlation Integral (aLOCI) is an extension of the LOCI algorithm which addresses the complexity problems. The aLOCI algorithm speeds up the LOCI operation by incorporating quad trees.

iv) Cluster-Based Local Outlier Factor

The prior discussed algorithms have a common feature which is that they rely on nearest-neighbours approaches. The Cluster-Based Local Outlier Factor (CBLOF) which was proposed by He, Xu and Deng (2003) identifies anomalies by relying on using clustering approaches. Due to its low computational complexity, it is common to use *k*-means. The clusters from the clustering algorithm are then grouped into small and big clusters by CBLOF. The distance of each data point to the cluster centroid times the number of data points in that cluster is how the anomaly score is calculated. The unweighted CBLOF (uCBLOF) proposed by Amer and Goldstein (2012) is an extension of the CBLOF. The uCBLOF, the authors claim that it excludes the scaling factor from the calculations because it introduces issues when calculating the densities. *Figure 3.13* shows the results of applying uCBLOF on a dataset. The different colours correspond to the clusters identified by the clustering algorithm used and the radius of the data points corresponds to the anomaly score assigned by uCBLOF to each data point.

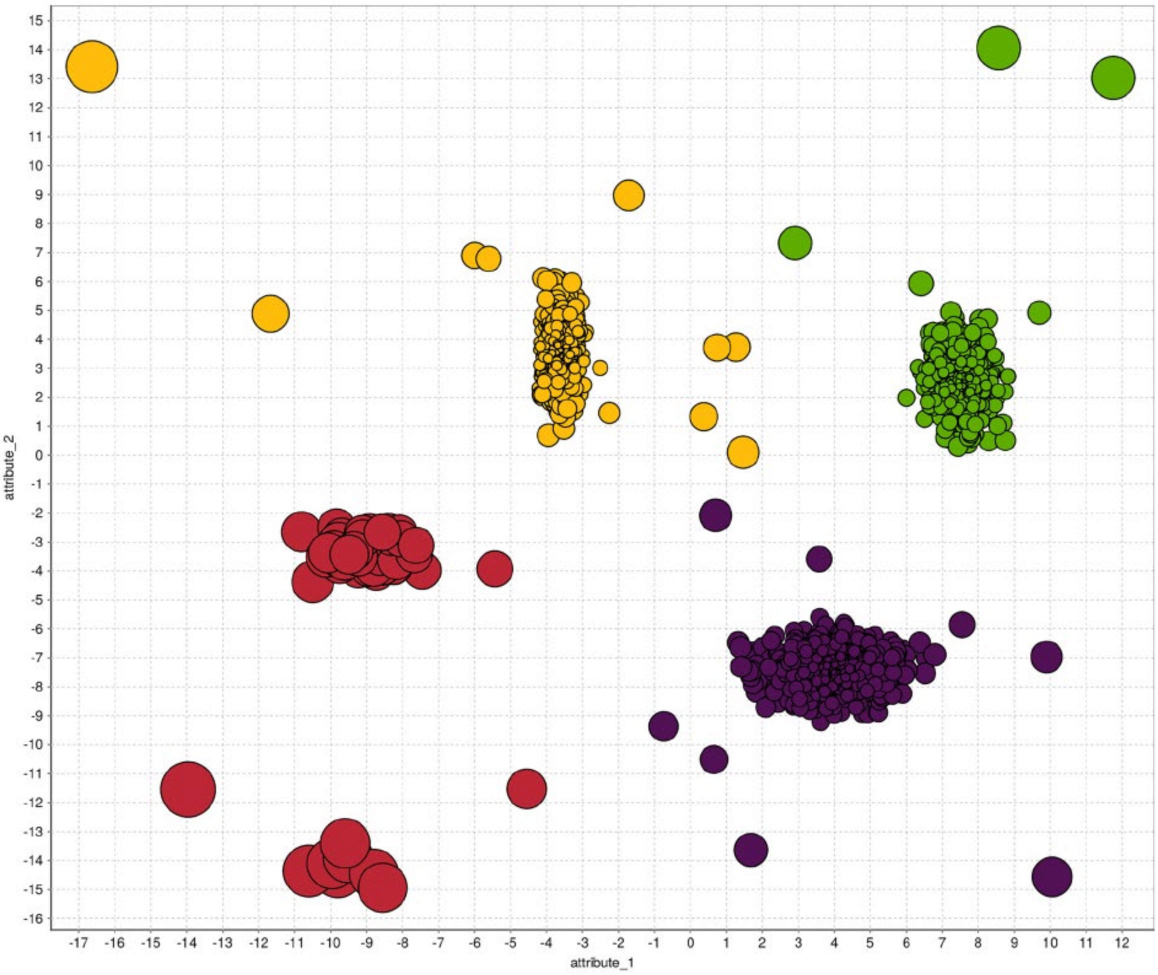


Figure 3.13 The unweighted Cluster-Based Local Outlier Factor (uCBLOF) algorithm (Goldstein and Uchida, 2016)

v) Local Density Cluster-Based Outlier Factor

One of the drawbacks of the CBLOF is its use of the number of a cluster data points as a density measure. Proposed by Amer and Goldstein (2012), the Local Density Cluster-Based Outlier Factor (LDCOF) uses a density measure of the identified clusters. Following the same approach of the CBLOF, it uses any clustering algorithm as a first step. This is followed by the calculations of the average distances from a cluster centroid to the data points that belong to it. The LDCOF calculates the anomaly score by dividing a data point distance to the centroid by the cluster average.

vi) Clustering-Based Multivariate Gaussian Outlier Score

The Clustering-Based Multivariate Gaussian Outlier Score (CMGOS), as its name suggests, depends on using a clustering algorithm as a first step (Goldstein and Uchida, 2016). The cluster density in CMGOS is calculated using a multivariate

Guassian model with Mohalanobis distance as the measurement function. A covariance matrix of each cluster is calculated after identifying the clusters by a clustering algorithm. The anomaly score is then defined by dividing the Mahalonobis distance of a data point by the X^2 distribution of the confidence interval.

vii) Histogram-Based Outlier Score

Proposed by Goldstein and Dengel (2012), the Histogram-Based Outlier Score (HBOS) is a statistical anomaly detection algorithm that assumes that the feature space is independent. The algorithm works with the idea of developing a histogram for every variable (feature or dimension) in the dataset. The height of the bin represents a density estimator for every data point. The multiplication of the inverse of estimated densities is the final score. Although the assumption that the features are independent is limiting, this assumption gives HBOS an advantage when dealing with high dimensional dataset as the algorithm complexity is linear in relation to the input size.

3.10 A Review of the State-of-the-Art Anomaly Detection Algorithms

This section is to review the available state-of-the-art unsupervised anomaly detection algorithms. The authors Falcão *et al.* (2019) have compared unsupervised anomaly detection algorithms. In their research, they have selected the algorithms, datasets, and metrics based on well-defined criteria.

Selection of the Algorithms

The following criteria was used to select the algorithms:

- The method should be fully unsupervised therefore, semi supervised learning algorithms are omitted.
- The methods should cover the main categories of unsupervised methods: neighbour-based, clustering, classification, statistical, angle, and density based.
- The methods should have been applied successfully for an anomaly detection.

3.10.1 Robust Principal Component Analysis (rPCA)

Successfully applied by Kwitt and Hofmann (2007), the rPCA is based on the Principal Component Analysis (PCA) which is used to reduce dimensionality of datasets. A stated by the authors, PCA, when used, detects subspaces in datasets by identifying the deviations from the expected subspaces for anomaly detection. The eigenvectors of the covariance matrix are the principal components of PCA and are therefore computed twice to improve the robustness.

3.10.2 Angle-Based Outlier Detection (ABOD)

The ABOD relates data to high dimensional spaces. This is implemented by using the variance in the angles between a data point to the other points as anomaly score (Lippmann *et al.*, 2000). In a polygonal chain (*p1, p2, p3*), each point in the dataset is used as the middle point *p2,* of the chain, whereas the points *p1 and p3* are regarded as any two different data points of the dataset, *p1 ≠ p2 ≠ p3.* Thereafter, all the angles of *p1p2p3* are measured and their variance is used to calculate the ABOF (Angle-Based Outlier Factor). Typically, the anomalies result in very small variance in the angles from a couple of points.

3.10.3 Fast Angle-Based Outlier Detection (FastABOD)

Like the ABOD, in FastABOD, the anomalous data points are detected depending on the variance of the angles between pairs of distance vectors to other points (Lazarevic *et al.*, 2003). The only angles considered are the pairs between the neighbours thereby working in a sub-quadratic time. For each data point, the algorithm first calculates the ABOF to its k-nearest neighbour as the normalised scalar product of the difference vectors of any pair of neighbours. Then, FastABOD ranks the data points according to their ABOF. The smaller the ABOF, the bigger the probability that the data point is anomalous.

3.10.4 One-class Support Vector Machine (one-class SVM)

This algorithm aims at learning a decision boundary for data points grouping (Schölkopf *et al.*, 2001). Despite this algorithm being first used for supervised support vector machines for semi-supervised anomaly detections, it can very well be used for unsupervised anomaly detection (Prasad, Almanza-Garcia and Lu, 2009). The one-class SVM is trained with the dataset and then each data point is classified considering the normalised distance of the data point from the determined decision boundary (Amer, Goldstein and Abdennadher, 2013).

3.10.5 Isolation Forest (IForest)

Structures data points as nodes of an isolation tree, assuming that anomalies are rare events with feature values that differ a lot from expected data points. Therefore, anomalies are more susceptible to isolation than the expected datapoints since they are isolated closer to the root of the tree instead of the leaves. It follows that a data point can be isolated and then classified according to its distance from the root of the tree (Liu, Ting and Zhou, 2008).

3.11 Selection of the Metrics

The metrics used by Falcão *et al.* (2019) have been used by the survey studies of (Powers (2011). The metrics are based on Boolean anomaly/expected labels assigned to a given data point.

The authors Falcão *et al.* (2019) defined their thresholds based on interquartile range. According to Wan *et al.* (2014), this range (interquartile range) is the difference between the two quartiles *Q3* and *Q1.*

- **Precision (P)**

True Positives (TP) were considered as the anomalies detected corresponding to the manifestation of the attacks, while the False Positive (FP) as the detected anomalies that did not. Precision is defined by Falcão *et al.* (2019) as the fraction of True Positives (TP) among the union of False Positives (FP) and True Positives (TP).

- **Recall (R)**

According to Falcão *et al.* (2019), Recall is mostly presented together with positives (P). its definition is the ratio of TP over the union of TP and the False Negatives (FN) which are the undetected anomalies.

- **F-Score (F_β) and F-Measure**

The F-Score (β) metric combines both Precision (P) and Recall (R) by use of a parameter β. Therefore, when $\beta > 1$, R is weighted more than P. According to Powers (2011), the F-Measure (F_1) is defined as the balanced mean of P and R, and this is adopted when it is deemed that FPs and FNs are equally undesired.

- **Accuracy (ACC)**

The accuracy is usually defined as the ratio of correct detections where you get both the True Positives (TP) and True Negatives (TN) among all the data points examined.

This then allows for the aggregation of the positive and negative scores into a unique metric (Falcão *et al.*, 2019).

- **Area Under ROC curve (AUC)**

The Receiver Operating Characteristic (ROC) curve is a graphical plot representing the performance of binary classifiers when their discrimination thresholds vary: the r is depicted by plotting R against a False positive rate. When there is a high value of the area underlying the ROC curve, it signifies that the identified algorithm is suitable for the dataset targeted (Falcão *et al.*, 2019).

3.12 Performance of Algorithms

In the research study conducted by Falcão *et al.* (2019), the authors have proposed a question: is there an algorithm (or a family) that performs better than the others? In their experiments and analysis, the authors obtained results by running 12 algorithms on 5 datasets. The results were ranked by F1 scores (see *Table 3.1* below). The median and standard deviation score for each metric were reported.

The observations made were that the first two algorithms belong to the classification family. Both Isolation Forest and One-Class SVM showed good scores for anomaly detection whereby Precision, Recall, and Accuracy scores were above 96%. This contrasted with the other classification algorithms angle-based which showed poor results for the F1 score.

Another observation that was interesting was that when the authors aggregated and plotted the results related to each family (see *Table 3.1* below). The results from this also showed that the classification family was the most effective even though statistical and density-based families showed similar results. The neighbour-based scores showed a bit lower score than the other families, although they (neighbour-based) have a higher standard deviation. It is observed that of the two neighbour-based algorithms (KNN and ODIN, KNN is significantly depicted as the worst, however, it shows a higher recall score. This according to Falcão *et al.* (2019), is explained by reason of ODIN being based on the KNN graph with some ‘indegree score’. As anticipated by Zhang *et al.* (2004) the semi-density score added on top of the KNN query provides a decisive support which improves the detection scores. A similar result was also exhibited in the clustering family whereby the K-Means algorithm was used as a baseline for the LDCOF, this showed better scores.

On the Accuracy scores observations, the authors Falcão *et al.* (2019) noted that even though the angle-based algorithms showed worst F1 scores overall, they (angle-based) have higher accuracy values than the clustering and neighbour-based families. The authors explain that this motivated by the fact that F1 score is based on Precision and Recall, which do not account for true negatives. As a result, higher Accuracy scores for angle-based algorithms compared to the corresponding F1 scores highlight that the percentage of true negatives is higher than the others.

Algorithm	**# Combinations**	**Family**	**AUC**	**Precision**	**Recall**	**Accuracy**	**F1**
Isolation Forest	8	Classification	37.2 ± 0.4	99.9 ± 0.3	99.3 ± 0.4	99.7 ± 0.3	99.6 ± 0.3
One-Class SVM	1	Classification	53.4 ± 2.9	96.6 ± 3.2	99.3 ± 0.0	96.2 ± 3.2	98.0 ± 1.9
COF	8	Density-Based	48.8 ± 1.7	93.6 ± 3.4	97.8 ± 0.1	91.7 ± 3.1	95.7 ± 2.0
ODIN	8	Neighbour-Based	49.9 ± 1.7	96.6 ± 2.4	99.9 ± 0.4	89.8 ± 1.6	94.6 ± 1.1
HBOS	1	Statistical	57.8 ± 5.5	92.6 ± 5.8	99.5 ± 4.3	89.2 ± 4.7	94.3 ± 4.8
rPCA	1	Statistical	55.0 ± 4.0	97.5 ± 3.4	95.0 ± 1.0	83.1 ± 3.2	90.6 ± 2.0
LOF	8	Density-Based	50.0 ± 1.3	96.6 ± 3.5	88.0 ± 1.1	81.3 ± 3.1	89.6 ± 2.1
LDCOF	8	Clustering	49.9 ± 2.3	82.4 ± 1.8	94.4 ± 0.2	77.9 ± 1.5	87.4 ± 0.7
KNN	8	Neighbour-Based	35.9 ± 6.7	91.9 ± 5.8	75.1 ± 3.4	71.4 ± 4.0	82.8 ± 4.3
K-Means	8	Clustering	54.4 ± 8.9	95.7 ± 5.3	68.5 ± 2.8	65.6 ± 3.4	78.3 ± 3.5
ABOD	8	Angle-Based	90.5 ± 7.8	69.2 ± 8.1	92.4 ± 8.3	90.0 ± 1.8	75.5 ± 10.2
FastABOD	15	Angle-Based	86.4 ± 9.2	90.6 ± 7.8	77.4 ± 5.3	67.6 ± 3.2	74.7 ± 6.1

Table 3.1 Metric scores (median ± std) for the 12 algorithms, ordered by F1score (Falcão et al., 2019)

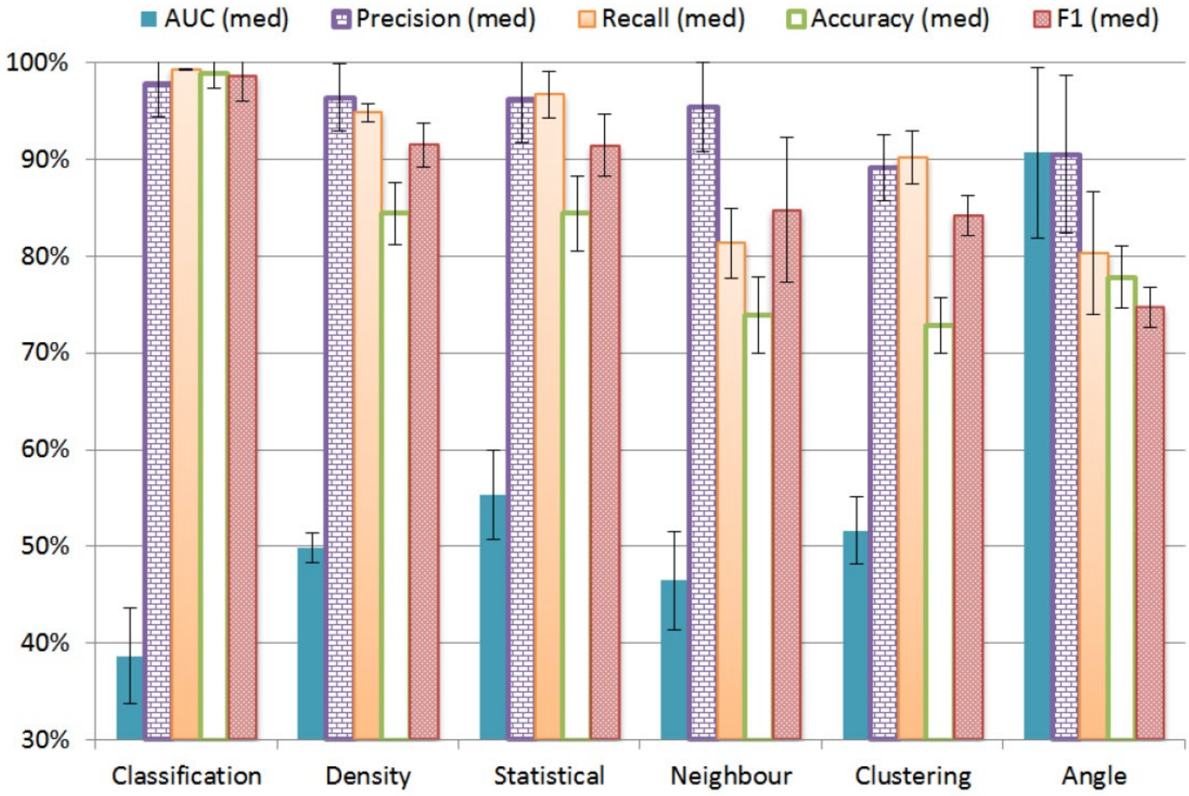


Figure 3.14 Results on all the datasets and all the attacks, grouped by algorithms families. Columns report median scores, while error bars depict the standard deviation (Falcão et al., 2019)

3.13 Chosen Algorithms

The results from *Table 3.1* above show that the recommended algorithms for better results in anomaly detection are Isolation Forest, One Class Support Vector Machine (OCSVM), Connectivity-Based Outlier Factor (COF), and Out-of-Distribution Detector for Neural Networks (ODIN).

The section below will focus on an in-depth discussion of these algorithms and how they have been implemented by other authors.

3.13.1 Isolation Forest

The Isolation Forest algorithm (also known as iForest) was proposed by the authors Liu, Ting and Zhou (2008). According to the authors, this algorithm isolates anomalies instead of profiling normal instances. To achieve this, the authors state that Isolation Forest takes advantage of two anomalies' quantitative properties (Liu, Ting and Zhou, 2008). These are that the anomalies:

a) Are the minority consisting of fewer instances and

b) The anomalies have attribute-values which are different from the attribute-values of the normal instances.

This shows that anomalies can be said to be ‘few and different’, making them more vulnerable to be isolated than the normal points. This vulnerability to isolation is the reason why anomalies are isolated closer to the root of the tree. The authors then called this tree as Isolation Tree or iTree. With this in mind, the authors then stated that the Isolation Forest creates an ensemble of iTrees for a given dataset, then the anomalies are defined as those instances that have a shorter average path length on the iTrees (Liu, Ting and Zhou, 2008).

This method only has two variables:

i) the number of trees to build and,

ii) the sub-sampling size.

The figures below (Figure 3.15) show the isolation of an anomalous point against a normal point.

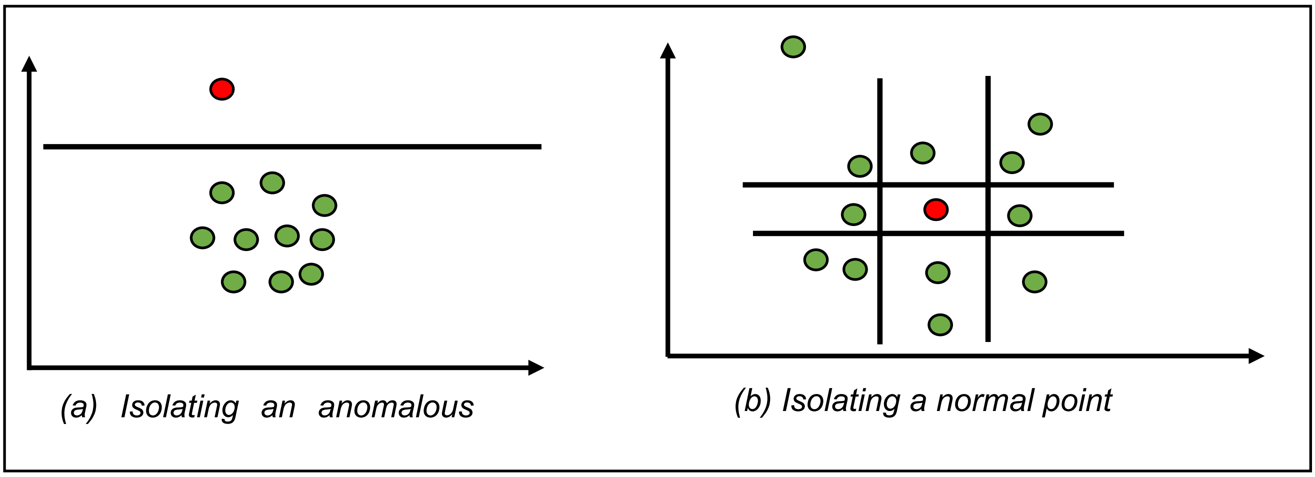


Figure 3.15 Isolating an anomalous point against a normal point

In Figure 3.15, it can be seen that the isolation of an anomalous point form normal points only uses one line (a), unlike the isolation of the normal points in (b) which requires four lines for complete isolation.

The iForest Algorithm

For a given sample of data points *X* in a dataset, the iForest algorithm builds an Isolation Tree (iTree), *T*. This is done using the following steps:

1. Random selection of an attribute *q* and a split value *p*.
2. *X* is divided into two subnets using the rule $q < p$. These subnets correspond to the left and right subtree in *T*.
3. The steps 1 and 2 are repeated recursively until the current node only has one sample or the current node has all the values that are the same.

All these steps are repeated severally to build Isolation Trees which in turn produce an Isolation Forest. With the understanding of how Isolation Trees are produced and the characteristics of the anomalous points, it can be said that most of the anomalous points will be located nearer to the root of the tree, reason being that they are easier to isolate as opposed to the normal points. An iTree is a proper binary tree, in that, every node in the tree has exactly zero or two daughter nodes. Assuming that all the nodes are distinct, each instance is isolated to an external node when iTree is fully grown, in this case, the number of external nodes is *n* and the number of internal nodes is $n – 1$; this therefore means that the total number of an iTree is $2n – 1$ (Liu, Ting and Zhou, 2008).

The main aim of anomaly detection is to give rise to a ranking that reflects the degree of anomaly. Therefore, Liu, Ting and Zhou (2008) state that another way of detecting anomalies is by sorting the data points according to their path lengths or anomaly scores, in which case, the anomalies are points which are ranked at the top of the list. The authors (Liu, Ting and Zhou, 2008) defined path length and anomaly score as follows:

Path Length

The Path Length *h(x)* of a point *x* is measured by the number of edges *x* traverses an iTree from the root node until the traversal is terminated at an external node (Liu, Ting and Zhou, 2008).

Anomaly Score

For any anomaly detection method, an anomaly score is required. The problem of deriving such a score from the Path Length *h(x)* is that while the maximum possible height of iTree grows in the order of *n*, the author Wilkes (1974) shows that the average height grows in the order of *log n*. Due to this, Liu, Ting and Zhou (2008) state that, normalisation of *h(x)* by any of the above terms is either not bounded or cannot be directly compared.

The iForest algorithm authors (Liu, Ting and Zhou, 2008) have compared the structure of iTrees to Binary Search Tree (BST) and concluded that both iTrees and BST have an equivalent structure. The authors have therefore borrowed the analysis of BST to estimate the average path length of iTree. This is because the termination of a node in an iTree is like an unsuccessful search in a BST in relation to the path length. Therefore, for a given dataset of *n* instances, the average path length of unsuccessful search in BST is given as:

$$c(n) = 2H\,(n-1) - (2(n-1)\,/n),$$

Equation 3.3 Path Length

where *H(i)* is the harmonic number and it can be estimated by the Euler's constant, *ln(i) + 0.5772156649.* As *c(n)* is the average of *h(x)* given *n,* the authors Liu, Ting and Zhou (2008) used it to normalise *h(x).*

The authors went further and defined the anomaly score *s* of an instance *x* as:

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

Equation 3.4 Anomaly Score

where, *h(x)* represents the path length of the data point *x* in each Isolation Tree; *E(h(x))* represents the expected or the average value across all the Isolation Trees; *c(n)* represents the average value of the path length *h(x)* given a sample size of *n.*

After the computation of the anomaly score *s(x, n)* for a given data point, the following criteria could be used to detect the anomalies

1. If *s(x, n)* is close to 1, then *x* is very likely to be an anomaly
2. If *s(x, n)* is less than 0.5, then *x* is likely a normal point
3. If *s(x, n)* is close to 0.5 for all of the points in the dataset, then it is likely that the data does not contain any anomalies.

N/B: *A reminder that the anomaly score will always be greater than zero but less than 1 for all the points, making it similar to a probability score.*

Figure 3.16 illustrates the relationship between *E(h(x))* and *s* when the following conditions are applied where *0 < s ≤ 1* for *0 < h(x) ≤ n – 1.*

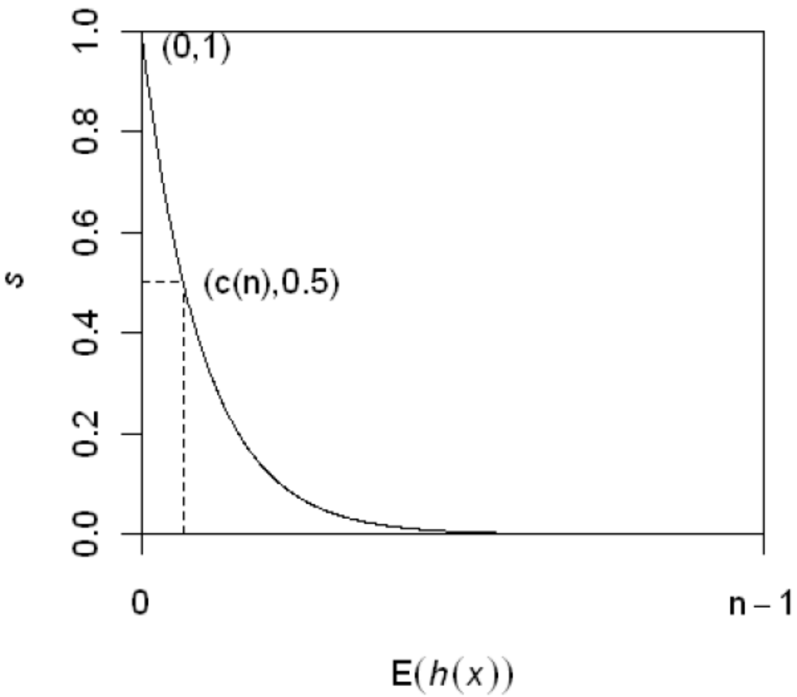


Figure 3.16 The relationship of expected path length E(h(x)) and anomaly score (Liu, Ting and Zhou, 2008)

From *Figure 3.16* above, the relationship of expected path length E(h(x)) and anomaly score s, c(n) is the average path length as defined in *Equation 3.3*. If the expected path length E(h(x)) is equal to the average path length c(n), then s = 0.5, regardless of the value of n (Liu, Ting and Zhou, 2008).

Anomaly Detection using iForest

The use of iForest for anomaly detection is in a two-stage process. The first stage is training which usually builds isolation trees by use of the sub-samples of the training set. The second stage - testing, passes the test instances through isolation trees to obtain an anomaly score for each instance.

Training stage

In the training stage, iTrees are constructed by recursively partitioning the given training set until instances are isolated or a specific tree height is reached of which results a partial model. Note that the tree height limit l is automatically set by the sub-sampling size $\psi: l = ceiling(log_2 \psi)$, which is approximately the average tree heigh. The rationale of growing trees up to the average tree height is that we are only interested in data points that have shorter-than-average path lengths, as those points are more likely to be anomalies (Liu, Ting and Zhou, 2008).

The details of this training stage are illustrated in the Algorithm 3.1 and Algorithm 3.2 below.

Algorithm 3.1: $iForest(X, t, \psi)$

Inputs: *X* – input data, *t* – number of trees, ψ – sub-sampling size

Output: a set of *t iTrees*

```
Initialise Forest
set height limit $l = ceiling(log_2 \psi)$
for i=1 to t do
    $X' \leftarrow sample(X, \psi)$
    $Forest \leftarrow Forest \cup iTree(X', 0,1)$
end for
return Forest
```

Algorithm 3.2: *iTree(X,e,l)*

Inputs: *X* – Input data, *e* – current tree height, *l* – height limit

Output: an iTree

```
if e≥ $l \; or \; |X| \leq 1$ then
    return $exNode\{Size \leftarrow [X]\}$
else
    let Q be a list of attributes in X
    randomly select an attribute q ∈ $Q$
    randomly select a split point p from max and min values of attribute q in X
    $X_l \leftarrow filter(X, q < p)$
    $X_r \leftarrow filter(X, q \geq p)$
    return inNode$\{Left \leftarrow iTree(X_l, e+1, l),$
                  $Right \leftarrow iTree(X_r, e+1, l),$
                  $splitAtt \leftarrow q,$
                  $SplitValue \leftarrow p\}$
end if
```

In the iForest algorithm above (Algorithm 3.1), there are two input parameters (sub-sampling size ψ and the number of trees t). The **sub-sampling size ψ** controls the training data size. It is found that when ψ *is* increased to a desired value, the detection of the iForest is reliable thereby eliminating the need for further increase of ψ as this increase only increases the processing time and memory size without any meaningful gain in the performance of the detection (Liu, Ting and Zhou, 2008).

The ensemble size is controlled by the **number of tree** t. After the training process, a collection of trees is returned making it ready for the evaluation stage, the authors Liu, Ting and Zhou (2008) state that the complexity of training an iForest is $O(t\psi \; log \; \psi)$.

Evaluating Stage

In this stage, an anomaly score S is derived from the expected path length $E(h(x))$ for each test instance. When instances are passed through each iTree in an iForest, the path length *E*(*h*(*x*)) is derived. By the use of the *PathLength* function, a single path length $h(x)$ is derived by counting the number of edges e from the root node to a terminating node as instance x traverses through an iTree. The termination of x at an external node where *Size>1,* the return value is e plus an adjustment $c(Size)$. The anomaly score is found by computing $s(x, \psi)$ in *Equation 3.4* when $h(x)$ is obtained for each tree of the ensemble. The *PathLength* details are illustrated in Algorithm 3.3 below.

Algorithm 3.3: *PathLength*(*x, T, e*)

```
Inputs : x - an instance, T - an iTree, e - current path length;
to be initialised to zero when first called
Output: path length of x
if T is an external node then
    return $e + c(T.size)$ {$c(.)$ is defined in Equation 3.3}
end if
$a \leftarrow T.splitAtt$
if $x_a < T.splitValue$ then
    return PathLength(x, T.left, e + 1)
else {$x_a \geq T.splitValue$}
    return PathLength(x, T.right, e + 1)
end if
```

The top m anomalies are found by simply sorting the data in descending order using s. The first m instances are the top *m* anomalies.

3.13.2 One Class SVM (OCSVM)

In one class classification, the constitution of the problem is covered by a single target sample of the same class represented by a training set usually separated from the any novel samples that do not belong to the same class (i.e., outlier samples). This algorithm (OCSVM) has been applied successfully in the various detection and classification tasks such as communication network performance, wireless sensor network, forensic science, detection of handwritten digits and objection detection among others (Noumir, Honeine and Richard, 2012).

According to Hempstalk and Frank (2008), the OCSVM algorithms have been extended to both binary and multiclass classification tasks through the application of a single one-class classifier to each class and subsequently combining the decision rules. The task of one class classification consists of identifying a sphere of minimum volume that englobes all (or most of) the training data by jointly estimating the centre and its radius.

Through the concept of reproducing kernels explored by Aronszajn (1950), a kernel function $k(.,.)$ defines a nonlinear transformation $\Phi(\cdot)$ of the input space into some feature space. What is required in the nonlinear characteristics is the inner product which can be evaluated using a kernel function $\langle\Phi(x_i), \Phi(x_j)\rangle = k(x_i, x_j)$ for any x_i, x_j from the input space *X.*

The main idea of OCSVM is to find a sphere of minimum volume that contains all the training samples. Therefore, the sphere, being described by its centre ***C*** and its radius *r,* is obtained by solving the constrained optimisation problem.

$$\begin{matrix} min \\ r, c \end{matrix} \qquad r^2$$

Equation 3.5

Subject to $\|\Phi(x_i) - c\|^2 \leq r^2$ for *i= 1,2,…….,n*

The above constraint may be deemed restrictive by tolerating a small fraction of the samples to be outside the sphere. In so doing, robustness is yielded such that there is less sensitivity to the presence of outliers in the training dataset. Due to this, Noumir, Honeine and Richard (2012) specified *V* to be a positive parameter for trade-off between the sphere volume and the number of outliers. The problem then becomes the estimation of *C, r,* and a set of non-negative slack variables $\zeta_{1,}\zeta_2 \ldots.., \zeta_n$:

$$\begin{matrix} min \\ r, c, \zeta \end{matrix} r^2 + \frac{1}{vn}\sum_{i=1}^{n}\zeta_i$$

Equation 3.6

subject to $\|\Phi(x_i) - c\|^2 \leq r^2 + \zeta_i$ for all *i = 1,2,…….,n*

By the introduction of the Kurush-Kuhn-Tucker (KKT) optimality conditions, the following equation is derived:

$$c = \sum_{i=1}^{n} \propto_i \Phi(x_i), \qquad \textit{Equation 3.7}$$

where the $\propto_i$ solves the optimisation problem.

Kittidachanan *et al.* (2020) have recently discussed the OCSVM algorithm. They differentiated SVM and OCSVM by stating that whereas both algorithms (SVM and OCSVM) work in unsupervised learning where only one class is considered, OCSVM tries to find hyperplane to separate the outlier from normal data during the training process (Kittidachanan *et al.*, 2020).

In the training process, the first step is to transform input data into kernel function, the data is then mapped from input space onto high-dimensional space (feature space). The algorithm then finds the best separating hyperplane from training data by maximising the margin.

The margin in this case is represented as:

$$Margin = \ b/\|\boldsymbol{w}\|$$

The objective function is represented as:

$$minimise\ f(\boldsymbol{w}) = \frac{\|\boldsymbol{w}\|^2}{2} + \frac{1}{vn}\sum_{i+1}^{n} \zeta_i - b \qquad \textit{Equation 3.8}$$

subject to : $\left(w\ \cdot \varPhi(x_i)\right) \geq b\ -\ \zeta_i, \zeta_i\ \geq 0, \forall i = 1, \ldots., n$

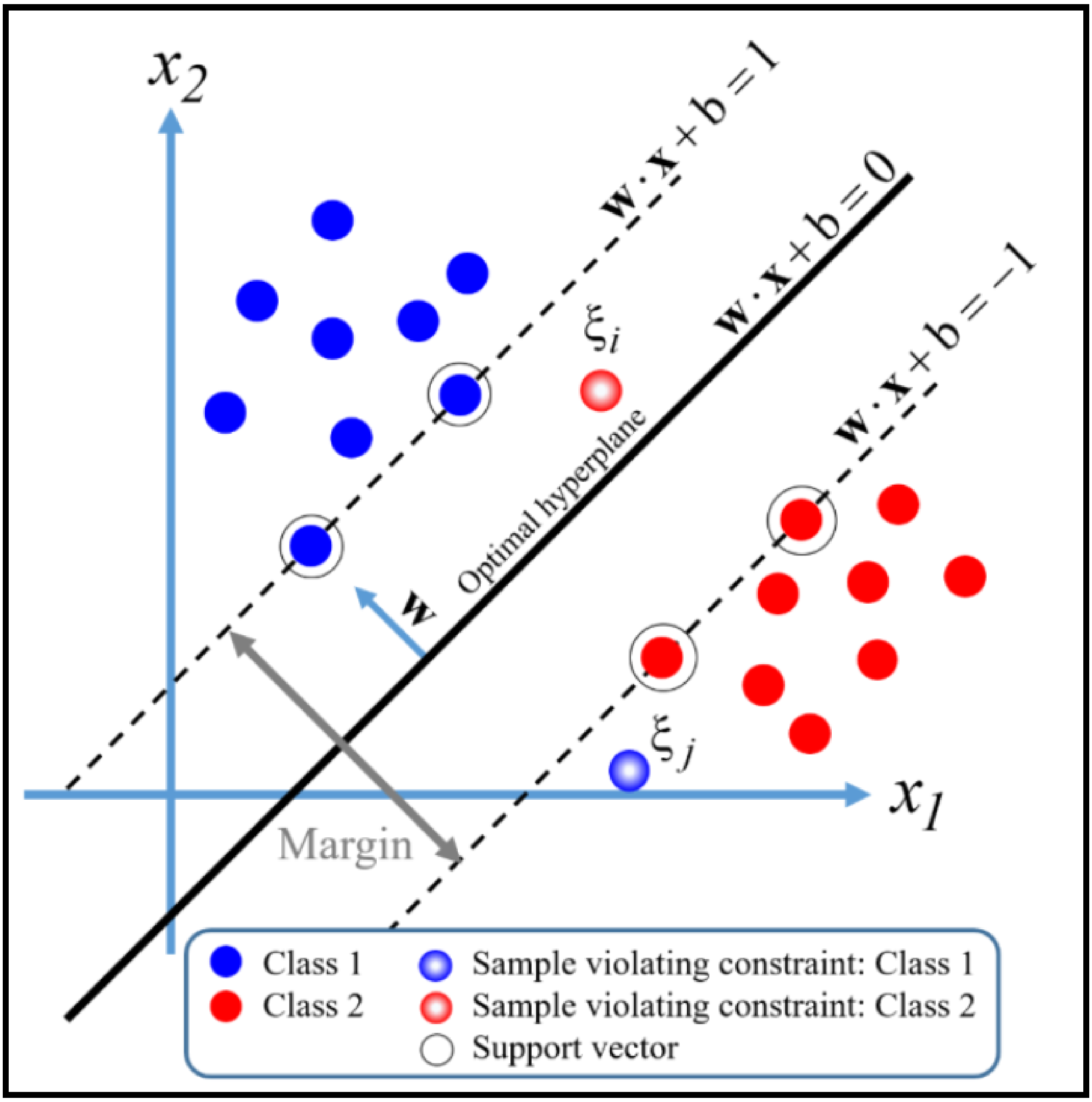
where *V* is the regularisation coefficient, or the parameter-controlled crossing of the data or the proportion of outliers. This parameter *v* is ranged from 0 to 1. The optimisation problem in the objective function above is usually solved by its dual form as below:

minimise: $-\sum_{i=1}^{n}\sum_{j=1}^{n} \propto_i \propto_j\ K\left(x_i, x_j\right)$

subject to: $\sum_{i=1}^{n} \propto_i\ = 1, 0\ \leq \propto_i \leq \frac{1}{vn}$

where the kernel function is $K\left(x_{i,}x_j\right) = \Phi(x_i)^T \cdot \Phi\left(x_j\right)$

The difference between two class and one class SVM is shown in *Figure 3.17*.



x_2
x_1
$\mathbf{w}\cdot\mathbf{x}+b=1$
$\mathbf{w}\cdot\mathbf{x}+b=0$
$\mathbf{w}\cdot\mathbf{x}+b=-1$
ξ_i
ξ_j
Optimal hyperplane
$\mathbf{w}$
Margin
● Class 1
● Class 2
● Sample violating constraint: Class 1
● Sample violating constraint: Class 2
○ Support vector

Figure 3.17 Two class SVM (Kittidachanan et al., 2020)

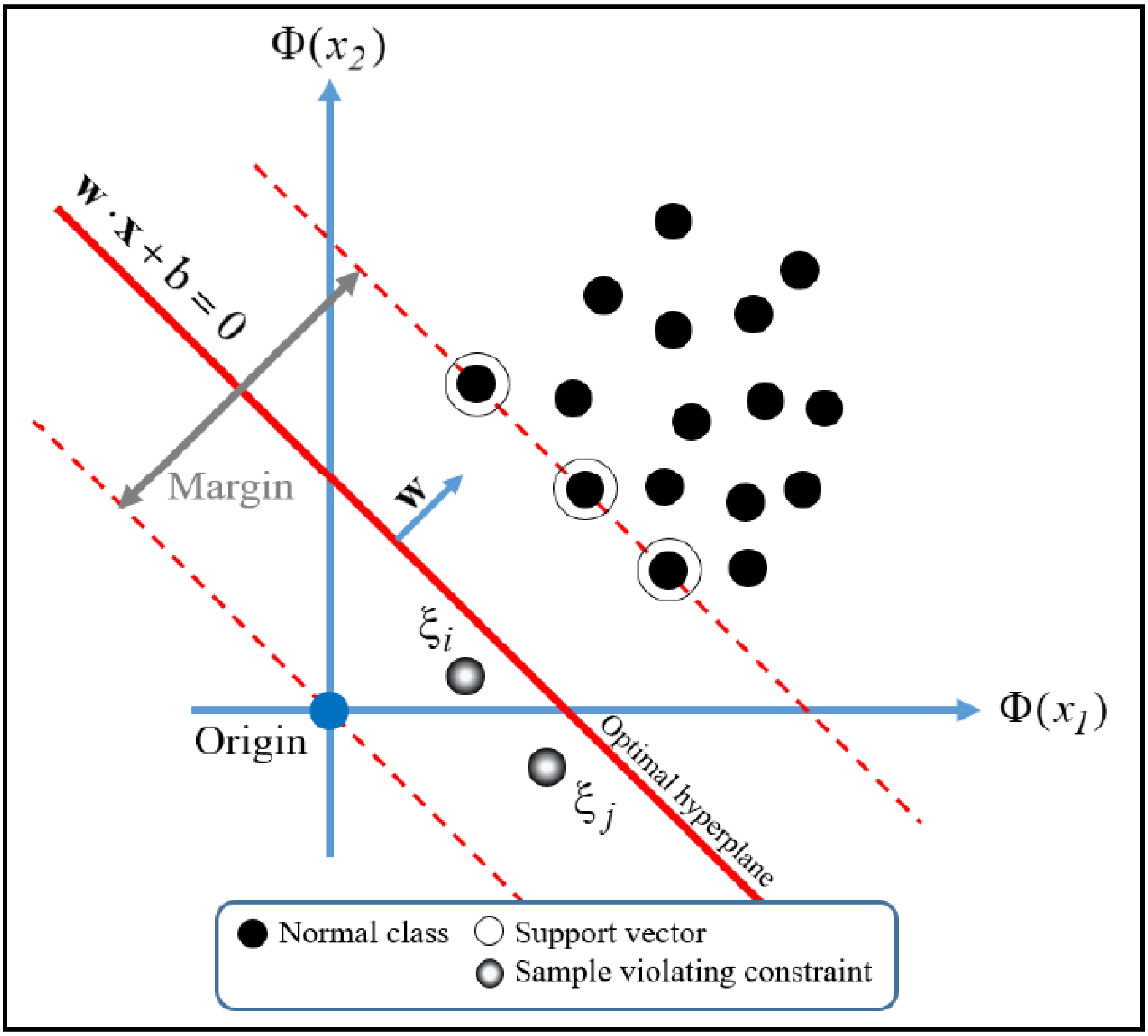


Figure 3.18 One Class SVM (Kittidachanan et al., 2020)

According to the authors Liu and Xu (2014), Lin and Lin (2003), and Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin (2008), the popular kernel functions that are frequently applied to OCSVMs are:

Linear:

$$K(x_i, x_j) = x_i x_j$$ *Equation 3.9*

Polynomial:

$$K(x_i, x_j) = (\gamma x_i x_j + r)^d, \gamma > 0,$$ *Equation 3.10*

where d is the degree of the polynomial kernel function (Chang and Lin, 2011).

Radial Basis Function (RBF):

$$K(x_i, x_j) = e^{-\gamma \llbracket x_i - x_j \rrbracket^2}, \gamma > 0$$

Equation 3.11

Sigmoid:

$$K(x_i, x_j) = \tanh(\gamma x_i x_j + r),$$

Equation 3.12

where i, j ∈ N. Parameters γ serve as mapping threshold coefficients that define boundary characteristics, and r is shifting parameter that controls the threshold of mapping.

Using grid search for hyperparameter tuning, the authors Kittidachanan *et al.* (2020) focussed on finding the best value of *V* (regularisation coefficient/proportion of outliers), and γ parameter in the kernel function.

SVM models are based on the hyperparameter values, therefore, grid search is used as a hyperparameter tuning process to the purpose of estimating the optimal values of parameters for an SVM model. In OCSVM, the two parameters considered are γ and *V.*

A summary of the grid search algorithm of OCSVM is as follows in Algorithm 3-4:

Algorithm 3.4: Grid Search OCSVM Hyperparameter Selection Algorithm.

```
Input: Target dataset $\boldsymbol{X}_{target}$ and $\boldsymbol{Y}_{target}$ ,
       Hyperparameter $\gamma_{range}$ and $\nu_{range}$
Output: Optimal hyperparameter combination
       $(\gamma_{opt}, \nu_{opt})$
prepare training data without negative class
    $(\boldsymbol{X}_{train}, \boldsymbol{Y}_{train})$
prepare testing data with negative class and positive class
    $(\boldsymbol{X}_{train}, \boldsymbol{Y}_{train})$
set minimum of AUC: $AUC_{best} = 0$ ;
for each hyperparameter combination $(\gamma, \nu)$ from
    $(\gamma_{range}, \nu_{range})$
    do
    train an OCSVM model with hyperparameter $(\gamma, \nu)$ by $\boldsymbol{X}_{train}$ ;
    predict $Y_{target}$ with train model by $\boldsymbol{X}_{test}$ ;
    calculate the AUC by $\boldsymbol{Y}_{test}$ and $\boldsymbol{Y}_{target}$ ;
    if $\boldsymbol{AUC_{best}} < \boldsymbol{AUC}$ then
        $(\gamma_{opt}, \nu_{opt})$ = $(\gamma, \nu)$;
return $(\gamma_{opt}, \nu_{opt})$;
```

3.13.3 Connectivity-Base Outlier Factor (COF)

COF was introduced by Tang *et al.* (2002) to improve the effectiveness of Local Outlier Factor (LOF) in patterns that have the same neighbourhood density as an outlier. Whereas LOF calculates Euclidean distances using hypersphere centred on a data point, COF calculates the distance in an incremental manner by finding the shortest paths between data points.

The authors' (Tang *et al.*, 2002) idea of COF is based on the idea of differentiating 'low density' from 'isolativity'. Low density is referred to as the understanding that the number of objects in the 'close' neighbourhood of an object is (relatively) small, while isolativity is referred to as the degree to which an object is connected to other objects. So, generally, a low density outlier results from a deviation from a pattern of high density, whereas an isolated outlier results from a deviation from a pattern that is connected (Tang *et al.*, 2002). Therefore, an outlier detector should take both cases into consideration.

The following ae the definitions for the formulation of the connectivity-based outlier factor (COF).

Definition 1:

Let $P, Q \subseteq \mathcal{D}, P \cap Q = \emptyset$ *and,*

P, Q$\neq \emptyset$. $dist(P,Q) = min\{dist(x,y) : x \in P \;\&\; y \in Q\}$,

Call $dist(P,Q)$ *the distance between P and Q.*

For any given $q \in Q$,, *q is the nearest neighbour to P in Q if there is a* $p \in P$ *such that* $dist(p,q) = dist(P,Q)$.

Definition 2:

Let $G = \{p_1, p_2, \ldots\ldots p_r\}$ *be the subset of* $\mathcal{D}$.

A set based nearest path (SBN-path) from p_1 *on* G *is a sequence* $\langle p_1, p_2, \ldots\ldots, p_r \rangle$ *such that for all* $1 \leq i \leq r - 1, p_{i+1}$ *is the nearest neighbour of set* $\{p_1, \ldots\ldots, p_i\}$ *in* $\{p_i + 1, \ldots\ldots, p_r\}$.

The SBN-path is used to indicate the order of presenting the nearest objects (Tang *et al.*, 2002).

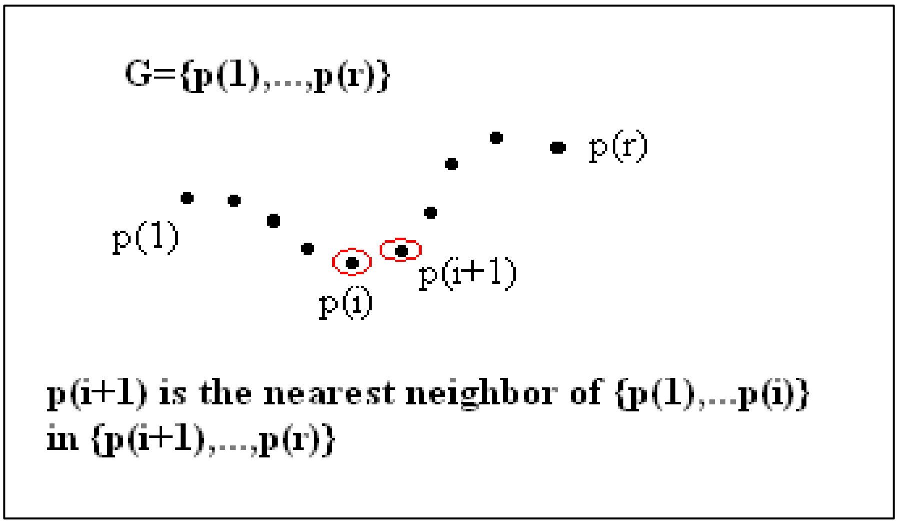


Figure 3.19 The Set Based Nearest (SBN) Path (Tang et al., 2002)

<u>Definition 3:</u>

Let S = $\langle p_1, p_2, \ldots\ldots, p_r \rangle$ *be an SBN-path.*

A set based nearest trail, or SBN-trail, with respect to **S** *is a sequence* $\langle e_1, \ldots, e_{r-1} \rangle$ *such that for all* $1 \leq i \leq r - 1, e_i = (0_i, p_i + 1),$

where $0_i \in \{p_1, \ldots, p_i\},$ *and*

$$dist\ (e_1) = \ dist\ (0_i, p_i + 1) = dist\ (\{p_1, \ldots, p_i\}, \{p_i + 1, \ldots, p_r\}).$$

Each e_1 *is called an edge.*

The sequence is $\langle dist(e_1), \ldots, dist(e_{r-1}) \rangle$ *is the cost description of* $\langle e_1, \ldots, e_{r-1} \rangle$.

In the event that 0_i is not uniquely determined, the tie is broken by a pre-defined order. This therefore means that SBN-trail is unique for any SBN-path.

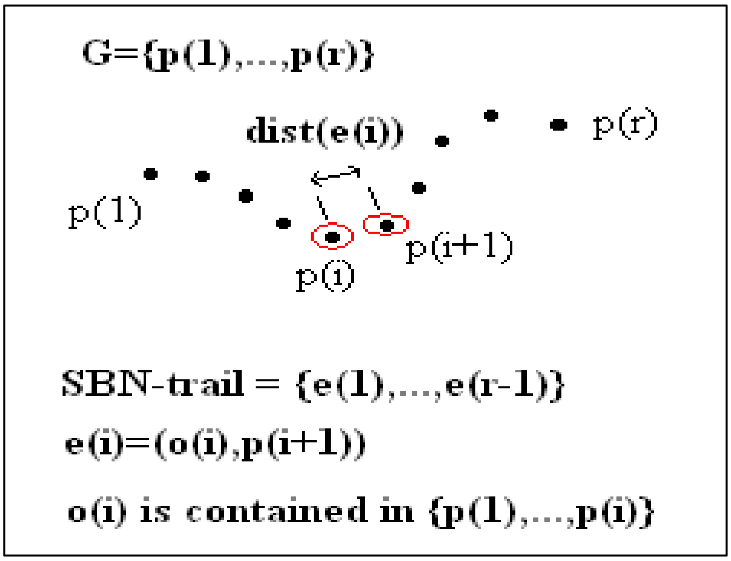

G={p(1),...,p(r)}
dist(e(i))
p(r)
p(1)
p(i+1)
p(i)
SBN-trail = {e(1),...,e(r-1)}
e(i)=(o(i),p(i+1))
o(i) is contained in {p(1),...,p(i)}

Figure 3.20 SBN-trail (Tang *et al.*, 2002)

Definition 4:

Let $s = \langle p_1, p_2, \ldots\ldots, p_r \rangle$ *be an SBN-path from* p_1 *and* $e = \langle e_1, \ldots, e_{r-1} \rangle$ *be the SBN-trail with respect to s. The average chaining distance from* p_1 *to* $G - \{p_1\}$*, denoted by* $ac - dist_G(p_1)$*, is defined as:*

$$ac - dist_G(p_1) = \sum_{i=1}^{r-1} \frac{2(r-1)}{r(r-1)} \cdot dist(e_i) \cdot$$

The weighted sum of the cost description of the SBN-trail for some SBN-path from p_1 is the average chaining distance from p_1 to $G - \{p_1\}$. As the cost description is unique for p_1, this definition is rewritten as:

$$ac - dist_G(p_1) = \frac{1}{r-1} \cdot \sum_{i=1}^{r-1} \frac{2(r-1)}{r(r-1)} \cdot dist(e_i)$$

When the fraction preceding the summation sign as the weight is viewed, the average chaining distance can then be viewed as the average of the weighted distances in the cost description of the SBN-trail. This means that if edges close to p_i are larger than those further away, then they contribute more to the average chaining distance as illustrated in *Figure 3.21* below.

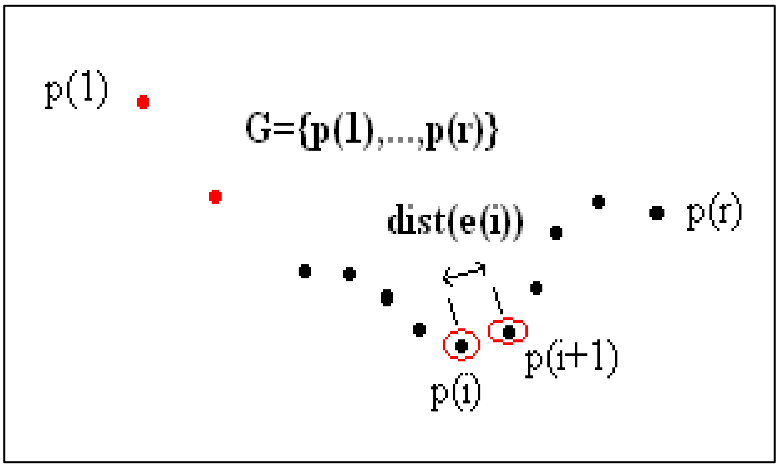


Figure 3.21 Average chaining distance (Tang *et al.*, 2002)

Definition 5:

Let $p \in \mathcal{D}$ and k be a positive integer. The connectivity-based outlier factor (COF) at p with respect to its k-neighbourhood is defined as:

$$COF_k(p) = \frac{|N_k(p)| \cdot ac - dist_{N_k(p)}(p)}{\sum_{0 \in N_k(p)} ac - dist_{N_k(0)}(0)}.$$

The COF at p is the ratio of the average chaining distance from p to $N_k(p)$ and the average chaining distance from p's *k*-distance neighbours to their own *k*-distance neighbours. It indicates how far away a point shifts from a pattern. It further compares the point to the points around it to influence the outlier factor as illustrated in *Figure 3.22*.

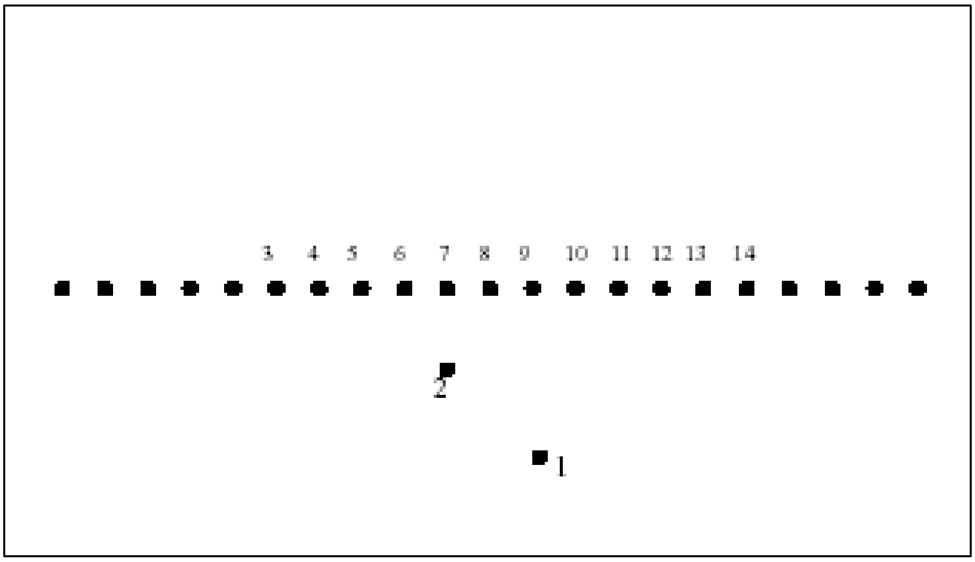


Figure 3.22 An example of calculating COF (Tang *et al.*, 2002)

3.13.4 Out-Of-Distribution Detector (ODIN)

ODIN was proposed by Liang, Li and Srikant (2017). The author stated that ODIN is a simple and effective method that does not require any change on a pre-trained neural network. The experimental methodology is based on observation that state that through the use of temperature scaling and adding small perturbations to the input can separate the SoftMax score distributions between in- and out-of-distribution images, which in turn allows for more effective detection (Liang, Li and Srikant, 2018). The authors' experiments also show that ODIN can be used for diverse network architectures and datasets.

The authors (Liang, Li and Srikant, 2018) have considered a problem to distinguish in- and out-of-distribution images on a pretrained neural network. A variety of variables have been defined: P_X and Q_X have been used to denote two distinct data distributions defined on the image space X. A neural network f is assumed have been trained on a dataset drawn from the distribution P_X. The P_X has been called the in-distribution and Q_X has been called the out-distribution.

The ODIN Detector is built on two components which are the temperature scaling and input preprocessing. These components are described as below:

Temperature Scaling

For a neural network $f = (f_1, \ldots., f_N)$, it is assumed that it has been trained to classify N classes. For an input x, the neural network assigns a label $\hat{y}(x)=arg\ max_i\ S_i(x;T)$ by computing the softmax output for each class.

$$S_i(x;T) = \frac{exp\ (f_i(x)/T)}{\sum_{j=1}^{N} exp(f_j(x)/T)}, \qquad \textit{Equation 3.13 Softmax Score}$$

where $T \in R^+$ is the temperature scaling parameter and is set to 1 during the training. For each input *x,* the maximum softmax probability $S\hat{y}(x;T) = max_i\ S_i(x;T)$ is called the softmax score. In their work, the authors Liang, Li and Srikant (2017) have mentioned that prior research has shown that the temperature scaling can be used to distil the knowledge in neural networks (Hinton, Vinyals and Dean, 2015) and could also be used to calibrate the prediction confidence in classification tasks (Guo *et al.*, 2017).

Input Preprocessing

Further to the temperature scaling, Liang, Li and Srikant (2017) pre-processed the input by adding small perturbations:

$$\hat{x} = x - \in sign(-\nabla_x \log S_{\hat{y}}(x;T)), \qquad \text{Equation 3.14}$$

where the parameter $\in$ is the perturbation magnitude. This method, as stated by the authors (Liang, Li and Srikant, 2018) is inspired by Goodfellow, Shlens and Szegedy (2015)'s idea of adversarial examples in which small perturbations are added to decrease the softmax score for the true label thereby forcing the neural network to make predictions that are wrong. For this, the ODIN authors Liang, Li and Srikant (2017) state that their goal and setting is the opposite of the adversarial examples. The authors' aim is to increase the softmax score of each input without needing a class label at all. It is also noted that the computation of the perturbations can be done easily by back-propagation of the gradient of the cross-entropy loss with respect to the input (Liang, Li and Srikant, 2018).

Out-of-distribution Detector

The ODIN detector combines the two components (temperature scaling and input preprocessing). For every input point ***X**,* a calculation for the pre-processed input point $\hat{x}$ is performed according to *Equation 3.4*. The pre-processed input point $\hat{x}$ is fed into the neural network then its calibrated softmax score $S(\hat{x};T)$ is calculated and this score is compared to the to the threshold δ. When the softmax score is greater than the threshold, the input point $\boldsymbol{X}$ is classified as in-distribution and vice versa.

Therefore, the out-of-distribution detector can be mathematically described as:

$$g(x;\delta,T,\in) = \begin{cases} 1\ if\ max_i\, p(\hat{x};\ T)\ \leq\ \delta \\ 0\ if max_i\, p(\hat{x};\ T)\ >\ \delta \end{cases}$$ Equation 3.15 Out-of-Distribution

The choice of the parameters $T,\ \in,$ *and* δ is so that the true positive rate is 95%. The true positive rate in this case is the fraction of in-distribution input points that are correctly classified as in-distribution images.

3.14 Conclusion

The integration of automated processes, AI, and ML into digital forensics has brought significant advancements to the field. These technologies have facilitated the examination and analysis of digital evidence, leading to faster results and improved efficiency in handling cybercrime cases. However, challenges remain, as automated tools still require human oversight and expertise to ensure accuracy and reliability. Ongoing development and training are necessary to enhance the capabilities of these intelligent systems. Despite these challenges, the use of automated processes in digital forensics holds great potential for future risk mitigation and the effective handling of digital evidence. As technology continues to advance, it is crucial for forensic investigators to stay updated and leverage intelligent tools to address the evolving landscape of cybercrimes and attacks. By embracing AI and automation, the field of digital forensics can better serve the demands of the digital age and contribute to the successful resolution of criminal investigations.

CHAPTER 4. PROPOSED THEORETICAL IoT FORENSIC FRAMEWORK

This chapter delves into the theoretical framework for conducting IoT forensic investigations, aiming to provide investigators with a structured and systematic approach to navigate the complexities of this unique landscape.

The chapter begins by acknowledging the gaps and limitations in existing research and highlights the need for a comprehensive framework that addresses the challenges specific to IoT environments. It emphasises the importance of timely and dependable data extraction and analysis, considering the constant connectivity and vast amounts of data generated by IoT devices.

The proposed IoT forensic framework is presented as a step-by-step guide, encompassing four distinct phases: Preparation, Live Investigation, Offline Investigation, and Presentation. Each phase is meticulously designed to cater to the specific requirements and circumstances of IoT investigations, taking into account factors such as live crime scenes, offline devices, evidence preservation, analysis techniques, and reporting.

Throughout the chapter, we explore the intricacies of each phase, discussing the key stages, sub-stages, and their significance in the investigative process. The framework recognises the critical need for securing crime scenes, documenting details meticulously, and maintaining a chain of custody to ensure the integrity of the investigation.

Additionally, this chapter underlines the importance of adapting the framework to suit individual investigations, as each case may present its own unique challenges and requirements. Practical implementation, case studies, and validation are recommended to assess the effectiveness of the framework in real-world IoT crime scenarios.

By offering a structured approach, this theoretical framework aims to enhance the effectiveness and efficiency of IoT forensic investigations, aiding investigators in the extraction of crucial evidence in a timely and reliable manner. Ultimately, this chapter seeks to contribute to the field of digital forensics by providing valuable insights and *guidance for investigators, helping to ensure justice is served and IoT-related crimes are prevented in this rapidly evolving landscape of interconnected devices.*

4.1 Introduction

A survey conducted by Adjei, Babu and Yakubu (2018) examined the existing approaches to digital forensics in the context of IoT. The survey identified various gaps and limitations in the sampled papers, highlighting the necessity for an enhanced proactive model to effectively address IoT crime scenarios. The survey concluded that very few of the frameworks and models proposed in the sampled papers could extract data in a timely and dependable manner. Over time, several IoT forensic processes have been proposed, encompassing methodologies, models, and frameworks, which have collectively advanced research in this field.

In this chapter, a step-by-step process of an IoT forensic process is proposed and presented to aid in the investigation of this research. A Theoretical IoT Forensic Investigation Framework is proposed and developed and utilises some processes from a network forensic framework developed by Hikmatyar, Prayudi and Riadi (2017). The framework has been developed with the emphasis being placed solely on IoT.

The framework is shown in Figure 4.1.

The framework follows the traditional digital forensic investigation process; however, it has been modified to accommodate the challenging scenarios in an IoT investigation scene. It emphasises on the critical activities that underscore the integrity and efficacy of the investigative process through ***authorisation***, ***maintaining of a chain of custody***, and ***documentation***. It stresses the crucial need for authorisation, highlighting the legal and ethical gateway to accessing and scrutinising sensitive data at any level of the forensic process. Additionally, the framework emphasises the strict need to maintain a chain of custody, recognising it as indispensable for ensuring the reliability and admissibility of evidence in a court of law. Further, the practice of documentation, particularly contemporaneous notes, is underscored for its role in providing a real-time, accurate record, serving as a transparent and accountable guide throughout the investigation.

The framework has four distinct phases, namely, preparation, live investigation, offline investigation, and presentation phase.

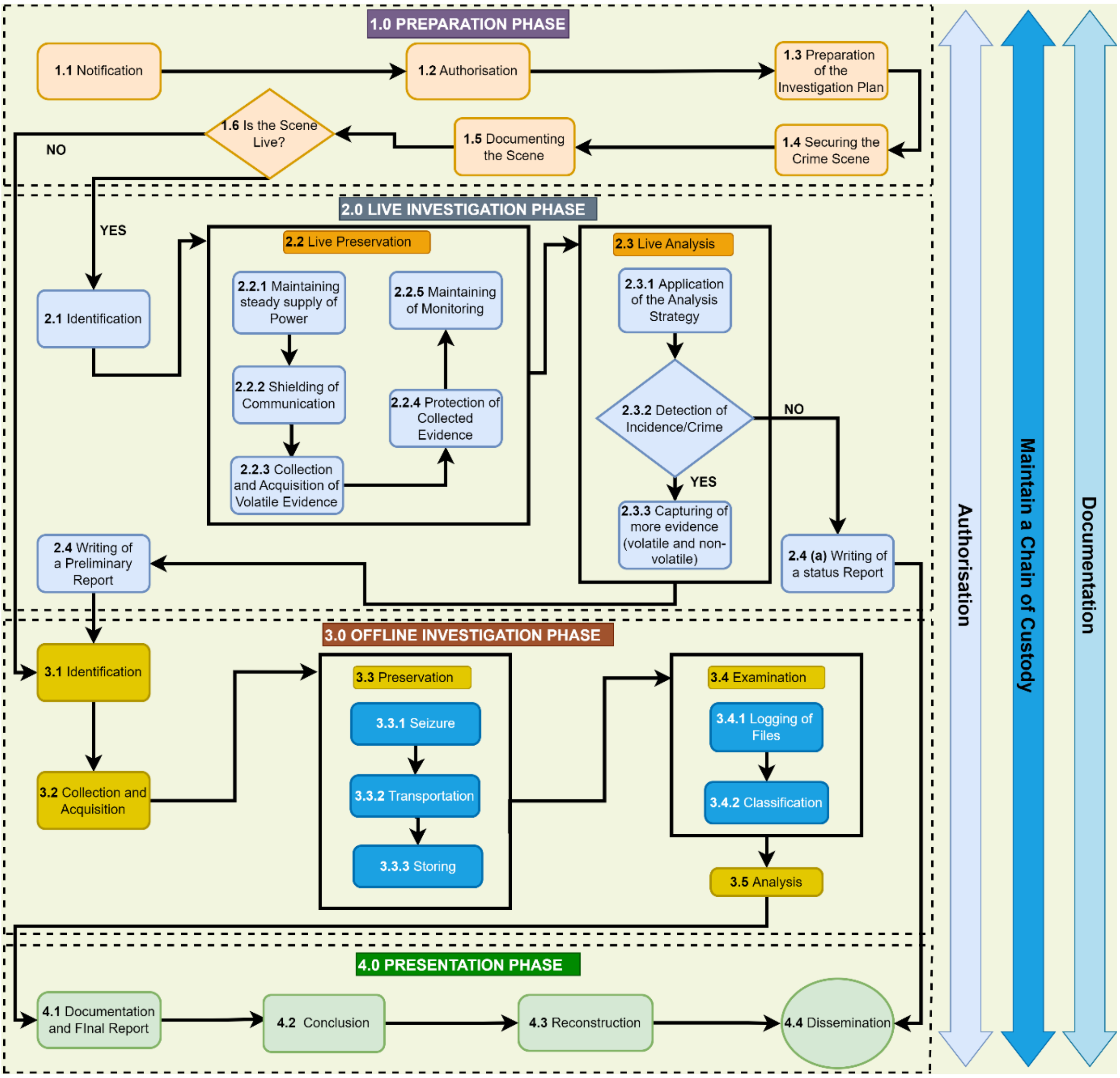

1.0 PREPARATION PHASE
1.1 Notification
1.2 Authorisation
1.3 Preparation of the Investigation Plan
1.4 Securing the Crime Scene
1.5 Documenting the Scene
1.6 Is the Scene Live?
NO
YES
2.0 LIVE INVESTIGATION PHASE
2.1 Identification
2.2 Live Preservation
2.2.1 Maintaining steady supply of Power
2.2.2 Shielding of Communication
2.2.3 Collection and Acquisition of Volatile Evidence
2.2.4 Protection of Collected Evidence
2.2.5 Maintaining of Monitoring
2.3 Live Analysis
2.3.1 Application of the Analysis Strategy
2.3.2 Detection of Incidence/Crime
NO
YES
2.3.3 Capturing of more evidence (volatile and non-volatile)
2.4 Writing of a Preliminary Report
2.4 (a) Writing of a status Report
3.0 OFFLINE INVESTIGATION PHASE
3.1 Identification
3.2 Collection and Acquisition
3.3 Preservation
3.3.1 Seizure
3.3.2 Transportation
3.3.3 Storing
3.4 Examination
3.4.1 Logging of Files
3.4.2 Classification
3.5 Analysis
4.0 PRESENTATION PHASE
4.1 Documentation and FInal Report
4.2 Conclusion
4.3 Reconstruction
4.4 Dissemination
Authorisation
Maintain a Chain of Custody
Documentation

Figure 4.1 Proposed IoT Forensic Investigation Framework

Each of these phases and steps have been explained below.

4.2 Preparation Phase (1.0)

For any investigation process, thorough preparation is crucial to ensure its successful completion. The stages involved in the preparation phase go beyond initial notification and include key steps that lay the foundation for a comprehensive investigation. These stages encompass notification, authorisation, preparation of the investigation plan, securing the crime scene, documenting the scene, and determining whether the scene is still active or offline.

Notification (1.1)

Notification serves as the initial step in an IoT forensic investigation, where law enforcement agencies or forensic investigators are informed of a suspected violation of the law. This notification takes the form of a detailed report that highlights the potential occurrence of a crime. It is essential to provide accurate and comprehensive information in the report to assist investigators in assessing the gravity of the situation and determining the necessary actions to be taken.

Authorisation (1.2)

Upon receipt of a report indicating a suspected crime, obtaining proper authorisation becomes crucial to enable investigators to access the relevant data and evidence. This authorisation is typically obtained through legal means, such as a warrant of arrest or a search and seizure warrant issued by the police. The authorisation should clearly outline the areas and scope within which the investigator has been granted access, ensuring compliance with legal procedures, and protecting the rights of all parties involved.

Preparation of the Investigation Plan (1.3)

Once authorisation has been obtained, the investigative team responsible for conducting the inquiry must meticulously plan and organise the necessary resources to effectively address the case. This includes acquiring appropriate software, hardware, and personnel with expertise in IoT forensic investigations. Pertinent training and briefing sessions should also be conducted for first responders, equipping them with the requisite knowledge, skills, and protocols to be followed upon arriving at the crime scene.

Securing the Crime Scene (1.4)

Preserving the integrity of the crime scene is of utmost importance in an IoT forensic investigation. The crime scene must be secured promptly to prevent any unauthorised access or tampering, which could compromise the validity and reliability of the evidence. Measures should be taken to seal off the area, control access, and prevent contamination, ensuring that the scene remains undisturbed until the investigation is complete. This involves establishing strict protocols and guidelines to be followed by all personnel involved in the investigation.

Documenting the Scene (1.5)

Thorough documentation of the crime scene is an essential step in an IoT forensic investigation. Investigators must conduct a meticulous examination of the scene, taking note of the various connections, types of communication, and hardware components present. Detailed photographs, sketches, and written descriptions should be captured to accurately record the condition and layout of the scene. It is imperative to document each step undertaken during the investigation process, creating a clear and traceable chain of custody that ensures the integrity of the evidence and establishes a foundation for subsequent analysis.

Is the Scene Live? (1.6)

As part of the preparation phase, investigators must determine whether the crime scene is still active or has transitioned to an offline state. This assessment helps in understanding the current state of the IoT environment and the potential risks and challenges involved in collecting and preserving evidence. By verifying the scene's activity status, investigators can tailor their approach accordingly, whether it requires immediate proactive measures for preserving live evidence or focuses on subsequent offline investigation procedures.

The meticulous execution of each stage within the preparation phase sets the groundwork for a systematic and effective IoT forensic investigation. These initial steps play a pivotal role in establishing a solid framework for subsequent phases, enabling investigators to proceed with confidence and maximise their chances of uncovering valuable evidence and insights in the pursuit of justice.

4.3 Live Investigation Phase (2.0)

Due to the inherent characteristics of IoT environments, where devices are constantly connected, it is possible that the crime scene remains active and live. In such cases, a proactive approach must be adopted to prevent the loss of crucial evidence. The Live Investigation Phase of IoT forensic investigations comprises of the identification, live preservation, live analysis, and writing of wither a preliminary or status report. Each of these stages play a crucial role in effectively managing and analysing the live crime scene.

Identification (2.1)

During this stage, the first responder(s) must diligently identify any readily available devices within the IoT environment that could potentially contain evidential value. This involves analysing the process events leading up to the incident and gathering information from witnesses or users present during the occurrence. By piecing together this information, investigators can gain insights into the triggers and context of the incident, forming an initial understanding of what transpired.

Live Preservation (2.2)

Live preservation involves immediate actions taken to maintain the integrity of the evidence within the live crime scene. This stage consists of several sub-stages that ensure crucial elements are safeguarded:

Steady Supply of Power (2.2.1)

In a live crime scene, maintaining a stable power supply is vital to prevent the loss of valuable evidence. To mitigate the risk of power disruptions, an uninterrupted power supply (UPS) should be employed whenever available, ensuring that devices and systems remain powered during the investigation process.

Shielding of Communication (2.2.2)

To prevent contamination and preserve the integrity of the investigation, it is imperative to isolate the network and communication mechanisms within the IoT environment. By disconnecting the live crime scene from external networks, investigators can eliminate the possibility of unauthorised access or interference.

Collection and Acquisition of Volatile Data (2.2.3)

Volatile data, which can be easily lost or altered, must be swiftly collected, and acquired to capture real-time evidence. This includes data stored in memory, temporary files, and active network connections. Specialised tools and techniques are employed to ensure the proper extraction and preservation of volatile data, which can provide crucial insights into the immediate events surrounding the incident.

Protecting Collected Evidence (2.2.4)

Once evidence is collected, it must be carefully protected to prevent loss, contamination, or theft. Strict protocols should be followed to securely store and transport the collected evidence, ensuring that its integrity is maintained throughout the investigation process. Proper labelling, sealing, and documentation are essential to establish a clear chain of custody and facilitate its admissibility in legal proceedings, if necessary.

Continuous Monitoring (2.2.5)

Continuous monitoring of the live crime scene and its communication channels is critical to detect any anomalies, leaks, or potential intruders. Investigators must closely observe the network traffic, system logs, and device behaviour to identify any unauthorised activities or unusual patterns. This ongoing monitoring provides valuable insights and helps in determining the scope and impact of the incident.

Live Analysis (2.3)

To establish an initial hypothesis and gain preliminary insights into the incident, a live analysis is conducted during this stage. The sub-stages involved in this process include:

Application of the Analysis Strategy (2.3.1)

Based on the nature of the investigation, an appropriate analysis technique or method is selected. This could involve examining network traffic, analysing system logs, conducting memory forensics, or utilising specialised tools and algorithms designed for IoT environments. The chosen strategy guides investigators in collecting relevant data and performing the subsequent analysis effectively.

Detection of a Crime (2.3.2)

Through the application of the chosen analysis strategy, investigators aim to determine whether a violation of the law has occurred, constituting a crime that requires further investigation and analysis. This stage involves examining the collected evidence and identifying any suspicious activities, unauthorised access, data breaches, or other malicious actions.

Capturing of more Evidence (2.3.4)

Upon confirming that a crime has taken place, the investigation proceeds to capture additional evidence, both volatile and non-volatile, that could strengthen the case. This includes expanding the scope of data collection beyond the initial observations and actively seeking out relevant information within the live crime scene. Investigators may deploy specialised tools and techniques to extract data from IoT devices, network logs, cloud services, and other sources, ensuring a comprehensive and thorough examination.

Report Writing (2.4)

Following the completion of the Live Investigation Phase, it is essential to document all actions taken and findings within a comprehensive report. This report serves as a record of the investigation process, providing a detailed account of the steps followed, evidence collected, analysis conducted, and initial conclusions reached. The report should adhere to proper documentation practices, including a clear chain of custody for all evidence, accurate timestamps, and a thorough description of the crime scene.

In instances where no significant evidence or indications of a crime are found during the live investigation, a status report should be prepared to document the outcome of the steps taken. This report serves to officially communicate that the investigation has concluded and no further action is required at that stage. It is important to maintain thorough documentation even in cases where no criminal activity is detected, as it ensures transparency and accountability in the investigative process.

By following the systematic framework outlined in the Live Investigation Phase, investigators can effectively manage and analyse live crime scenes within IoT environments. The emphasis on proactive preservation, continuous monitoring, and careful evidence collection enables the preservation of vital data and enhances the chances of successfully identifying and prosecuting perpetrators in IoT-related crimes.

4.4 Offline Investigation Phase (3.0)

The offline investigation phase in an IoT environment, also known as reactive investigation, occurs when the IoT devices are disconnected or offline, often due to being collected from the crime scene and secured at a different location. This phase may also be conducted as a follow-up to the live investigation, aiming to optimise the overall investigation process. The following stages are involved in this phase:

Identification (3.1)

In the identification stage, investigators must carefully identify and locate potential evidence, including IoT devices that may contain data of evidential value. Thoroughly search all relevant areas where evidence may be concealed. It is crucial to maintain an open mind, employ keen investigative skills, and uncover any potential evidence without making assumptions prematurely.

Collection and Acquisition (3.2)

During the collection and acquisition stage, all identified evidence must be gathered and the data contained within extracted for further analysis. This process involves employing appropriate forensic techniques to ensure the preservation and extraction of valuable evidence stored within the IoT devices.

Preservation (3.3)

To maintain the integrity of the investigation, preserving the collected evidence is of utmost importance. This stage involves three sub-stages:

Seize (3.3.1)

Under the authority of a valid search and seizure authorisation, any evidence collected from the crime scene should be properly seized. It is essential to correctly label and secure all seized items, ensuring that authorised personnel are the only ones with access until after the investigation is completed.

Transport (3.3.2)

Once seized, all collected evidence must be securely transported from the crime scene to a forensic laboratory or a designated secure storage facility. Proper packaging and protection are necessary to prevent any damage or tampering during transportation.

Store (3.3.3)

The acquired digital evidence should be stored in an appropriate and secure database, while any other seized devices should be stored in a controlled environment.

Maintaining a detailed chain of custody is vital, documenting all individuals who accessed the evidence, along with the date and time of access.

Examination (3.4)

The examination stage involves a thorough analysis of the collected evidence to uncover the nature of the committed criminal offense. This stage comprises two sub-stages:

Logging of Files (3.4.1)

Due to the potentially large volumes of IoT data that may have been collected, all digital evidence is provisioned in a database for efficient management and retrieval. Proper logging and organisation of files facilitates subsequent analysis and examination.

Classification (3.4.2)

During the classification sub-stage, the evidence is categorised and classified based on criteria established by the investigative team. This classification aids in structuring the subsequent analysis and ensures a systematic approach to identifying relevant patterns or anomalies.

Analyse (3.5)

The analysis stage involves conducting a detailed and in-depth investigation of the evidence to determine any violations of laws or regulations. Using appropriate forensic tools and methodologies, the analysis aims to reveal meaningful findings and establish a clear course of action for further legal proceedings.

In summary, the offline investigation phase in IoT forensic investigations plays a crucial role in uncovering and analysing evidence that may have been collected from the crime scene or obtained after the live investigation. This phase involves meticulous identification, collection, preservation, examination, analysis, and documentation of digital evidence. By following a systematic and well-defined process, investigators can maximise the potential of the collected evidence to reveal the nature of the criminal offense committed and provide valuable insights for legal proceedings.

During the identification stage, investigators must employ their expertise to identify potential evidence, including IoT devices, and conduct thorough searches to uncover hidden data. The collection and acquisition stage requires the use of specialised forensic techniques to gather evidence and extract valuable data from IoT devices, ensuring the preservation of evidential integrity.

Preservation is a critical aspect of the offline investigation phase, encompassing the proper seizure, transportation, and storage of collected evidence. Maintaining a robust chain of custody and employing hash functions helps safeguard the integrity of the evidence and ensures its admissibility in legal proceedings.

The examination stage involves meticulous logging and classification of digital evidence, allowing for efficient management and subsequent analysis. By conducting a detailed analysis of the evidence, investigators can uncover patterns, anomalies, and indications of criminal activity, providing a solid foundation for legal action.

By following a systematic and rigorous offline investigation process, forensic investigators can effectively navigate the complexities of IoT environments and leverage the available evidence to uncover the truth. The successful completion of this phase lays the groundwork for the subsequent stages of the investigation, ensuring that justice is served, and the integrity of the legal system is upheld.

4.5 Presentation Phase (4.0)

The presentation phase is the culmination of the IoT forensic investigation process, where the findings and results are presented in a comprehensive report. This final phase comprises several key stages that contribute to the overall effectiveness and impact of the investigation. This final phase has these stages:

Documentation and Final Report (4.1)

During the investigation process, it is crucial to have thorough documentation. This involves creating a detailed report that captures all the activities conducted throughout the various stages of the investigation. The report should offer a comprehensive overview of the methods utilised, evidence gathered, analysis carried out, and the final findings and conclusions. Thorough documentation is vital as it ensures transparency, promotes collaboration with other stakeholders, and acts as a reliable record for possible legal proceedings.

Conclusion (4.2)

The conclusion stage of the offline investigation phase serves as the closing remarks of the investigation report. It involves summarising the key findings, evidence, and analysis conducted throughout the investigation process. This section provides a concise overview of the investigation, highlighting the main outcomes and conclusions drawn from the evidence collected and analysed. The conclusion should be written in clear and simple language, ensuring that it can be easily understood by both professionals and non-professionals involved in the case.

Reconstruction (4.3)

Reconstruction is a crucial stage of the presentation phase that focuses on piecing together the sequence of events that occurred during and after the crime. Through careful analysis of the collected evidence, including digital data, witness testimonies, and expert opinions, investigators aim to reconstruct the series of actions and events that led to the incident. This process provides a coherent narrative and helps establish a timeline of events, aiding in understanding the circumstances surrounding the crime. The reconstruction is typically documented in the investigation report, which details the investigative steps taken, the evidence reviewed, and the rationale behind the determined sequence of events.

Dissemination (4.4)

The dissemination stage involves making the investigative report available within the forensics community and relevant stakeholders. By sharing the findings, methodologies, and insights gained from the investigation, the wider forensic community can benefit and use this information to enhance their own practices and investigations. Dissemination may occur through various channels, such as conferences, workshops, research papers, or specialised forums dedicated to forensic science and IoT investigations. This stage contributes to the collective knowledge and expertise in the field, fostering continuous improvement and promoting best practices for future similar cases.

The presentation phase serves as the final step in the IoT forensic investigation process, where the results and conclusions of the investigation are compiled, summarised, and shared. By effectively communicating the findings and sharing the investigative process with the relevant stakeholders, the presentation phase ensures transparency, facilitates collaboration, and contributes to the advancement of forensic science in the context of IoT investigations.

4.6 Automation of the Proposed Forensic Framework

Part of the objectives of this research is the semi-automation of the IoT forensic processes. This study therefore attempts to apply Machine Learning techniques to semi-automate the Examination step within the proposed IoT forensic framework as shown in Figure *4.2*.

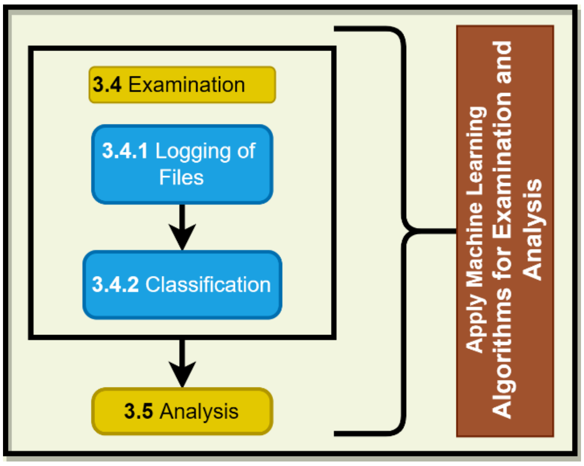


Figure 4.2 Steps Proposed for Automation

This semi-automation of IoT forensic processes, is aimed at leveraging the capabilities of machine learning techniques to enhance the efficiency and effectiveness of analysing big IoT data. Recognising the growing complexity and scale of IoT environments, the study endeavours to harness the power of machine learning to semi-automate the Examination step within the proposed IoT forensic framework. Through the application of carefully selected machine learning algorithms – Isolation Forest and One Class Support Vector Machines (detailed in Section 3.13), this research seeks to advance the analysis of IoT big data, thereby augmenting the investigative capabilities of forensic practitioners. Specific attention is geared towards the tasks within the Examination step, such as data classification, pattern recognition, and anomaly detection, which lend themselves to semi-automation through machine learning as further explored in Section 6.6. Anticipated outcomes of this endeavour include notable improvements in analysis speed and accuracy, as well as the ability to discern complex patterns and anomalies in IoT data streams. While recognising the challenges inherent in semi-automation, including the need for high-quality training data and algorithmic interpretability, the research aims to mitigate these obstacles through rigorous data preprocessing, model validation, and human oversight mechanisms as described in Sections 6.5 and 6.6 with a detailed illustration in Figure 6.2. Ultimately, the proposed semi-automation endeavours to contribute to the advancement of IoT forensic methodologies, filling a critical gap in the literature and equipping forensic investigators with innovative techniques to address the evolving landscape of huge data encountered IoT digital forensics investigations.

4.7 Conclusion

The proposed theoretical IoT forensic framework presented in this chapter offers a comprehensive and structured approach to conducting investigations in the context of the IoT. By incorporating the insights gained from the different phases of the investigation process, the framework addresses the unique challenges posed by IoT environments, such as constant connectivity and vast data volumes.

During the Preparation Phase, the framework emphasises the importance of thorough planning, notification, and authorisation to ensure a solid foundation for the investigation. The Live Investigation Phase recognises the need for proactive measures to preserve and analyse evidence in real-time, considering the dynamic and interconnected nature of IoT systems. The Offline Investigation Phase focuses on the meticulous collection, preservation, examination, and analysis of offline or collected IoT devices, ensuring the integrity of the investigation is maintained.

The framework also highlights the significance of reconstruction, where the sequence of events is pieced together to provide a coherent narrative of the crime. This stage aids investigators in understanding the circumstances surrounding the incident and helps establish a timeline of events. Furthermore, the Presentation Phase facilitates the dissemination of investigation findings to the forensics community, promoting knowledge sharing, collaboration, and continuous improvement in the field.

The framework acknowledges the need for strong security measures to secure crime scenes, prevent contamination, and maintain the chain of custody of evidence. It also emphasises the importance of detailed documentation throughout the investigation process to ensure transparency, traceability, and accountability.

While the proposed framework offers a structured and systematic approach, it recognises the need for flexibility and adaptation to suit the specific requirements of each investigation. Real-world implementation and case studies are essential to validate and refine the framework, considering the evolving nature of IoT technologies and associated forensic challenges.

In conclusion, the proposed theoretical IoT forensic framework contributes to the field of digital forensics by providing guidance and structure for investigators in navigating the complexities of IoT investigations. By following this framework, investigators can enhance the effectiveness and efficiency of their investigations, aiding in the pursuit of justice and the prevention of IoT-related crimes in the ever-expanding world of connected devices.

CHAPTER 5. SIMULATION AND DATASET GENERATION

This chapter provides a comprehensive overview of forensic analysis in smart homes as an IoT environment and explores the challenges and opportunities inherent in investigating smart home environments.

The chapter begins by introducing the concept of smart homes and their increasing prevalence in modern society. It highlights the diverse range of interconnected devices and sensors found within these environments and emphasises the significance of forensic analysis in uncovering evidence related to crimes, anomalies, or security breaches. One of the key topics discussed in this chapter is the use of simulation for generating data in smart home forensic analysis. It delves into the benefits and limitations of simulation and introduces the OpenSHS simulator as a powerful tool for simulating smart home environments. By utilising OpenSHS, researchers can generate realistic data and scenarios to analyse various forensic aspects. The chapter explores different aspects of simulation, including the generation of sensor data and activity patterns. It discusses how simulation can aid in understanding the behaviour of smart home systems and assist in identifying deviations or anomalies that may indicate suspicious activities.

Furthermore, the chapter outlines the preparatory steps for conducting forensic analysis in smart homes using simulation. It provides insights into setting up the OpenSHS simulator, configuring simulated devices, and designing realistic scenarios to mimic actual smart home environments. These steps ensure that the generated data aligns with real-world situations, enhancing the accuracy and reliability of forensic investigations.

5.1 Introduction

In recent years, researchers have shown a growing interest in analysing data obtained from IoT environments. Alongside the collection of this data, there has been a surge in research focused on developing intelligent machine learning algorithms and techniques to enhance service provision for smart home residents (Alshammari *et al.*, 2017). For instance, the creation of intelligent services requires effective methodologies to classify and recognise Activities of Daily Living (ADLs) as well as detect anomalies in ADLs. However, reliable datasets are necessary to test and validate the results of such research projects (Helal *et al.*, 2011). The medical field particularly recognises the importance of analysing ADLs due to their effectiveness in detecting patients' medical conditions (Tapia, Intille and Larson, 2004). Therefore, researchers need to construct datasets that accurately represent the data collected from smart home scenarios (Alshammari *et al.*, 2017).

However, building real smart homes to gather datasets from such scenarios is often costly and impractical for many projects, as noted by Helal *et al.* (2011), Armac and Retkowitz (2007), Lei *et al.* (2010), Synnott, Nugent and Jeffers (2015) and Mendez-Vazquez, Helal and Cook (2009). These authors also highlight several issues researchers encounter when constructing real smart home scenarios, such as the challenge of determining optimal sensor placement, lack of flexibility, difficulty in finding suitable participants, and concerns regarding privacy and ethical implications posed by smart homes (Fu *et al.*, 2011).

Although smart home datasets do exist, as mentioned by Alemdar *et al.* (2013) and Cook *et al.* (2003), they often do not fully meet the requirements of the research being conducted. These datasets may not allow the incorporation of additional sensors or grant the researcher control over the generated scenarios. Moreover, creating real datasets can be a labour-intensive task, and if not executed carefully, it may lead to inaccurate outcomes.

Building real smart home test beds presents challenges in preparing the datasets, as highlighted by Alshammari *et al*, (2017). One of these challenges is ensuring the continuous and robust capture of sensor data. Additionally, there is a need for a suitable documentation method to accurately record all the activities of the smart home inhabitants.

5.2 Simulation Tools

The challenges associated with constructing real smart homes to generate authentic datasets can be effectively addressed through the utilisation of dataset simulation tools. These tools offer a rapid means of generating datasets and provide a robust mechanism for capturing sensor data. By employing simulation tools, researchers can overcome the limitations and difficulties of gathering data from real smart home environments.

Simulation tools also prove beneficial in accurately annotating activities ADL by offering features like pausing or fast-forwarding the simulation process. When developing machine learning algorithms targeted at specific functionalities, researchers heavily rely on the availability of representative datasets. Typically, these datasets are divided into two groups: training and testing. The model creation process involves initialising parameters and training the model on a portion of the dataset. Subsequently, the model is tested on a different subset of the same dataset, and the results are evaluated. These results help identify whether modifications need to be made to the smart home setup (such as adding or removing smart devices) or if the generated scenarios should be adjusted. Simulation provides the flexibility to make such changes, which would be costly and unfeasible in a real smart home scenario (Alshammari *et al.*, 2017).

Figure *5.1* illustrates the degree to which a researcher can tweak and modify scenarios in both real and simulated smart home environments. It clearly demonstrates that such modifications are more easily accomplished in a simulated environment, as researchers can revisit and modify the smart home design as needed.

As previously mentioned, it is evident that virtually simulated smart home offers much more flexibility and is less costly compared to a real smart home simulation (Synnott, Nugent and Jeffers, 2015). The simulated environments can be augmented by the advancement of computer graphics for example, the virtual reality technologies which can provide immersive and semi-realistic experiences that could imitate real-life experiences.

By employing dataset simulation tools, researchers can mitigate the challenges posed by physical smart home construction and leverage the advantages of flexibility and adaptability offered by simulations. This enables them to refine their models and experiments more efficiently, leading to improved results and insights in the field of smart home research.

5.3 Smart Home Simulation Tools

Smart home simulators are developed to serve many different purposes, however, the main functions of these simulators are mostly to collect data or to visualise a smart home scenario (Renoux and Klügl, 2019).

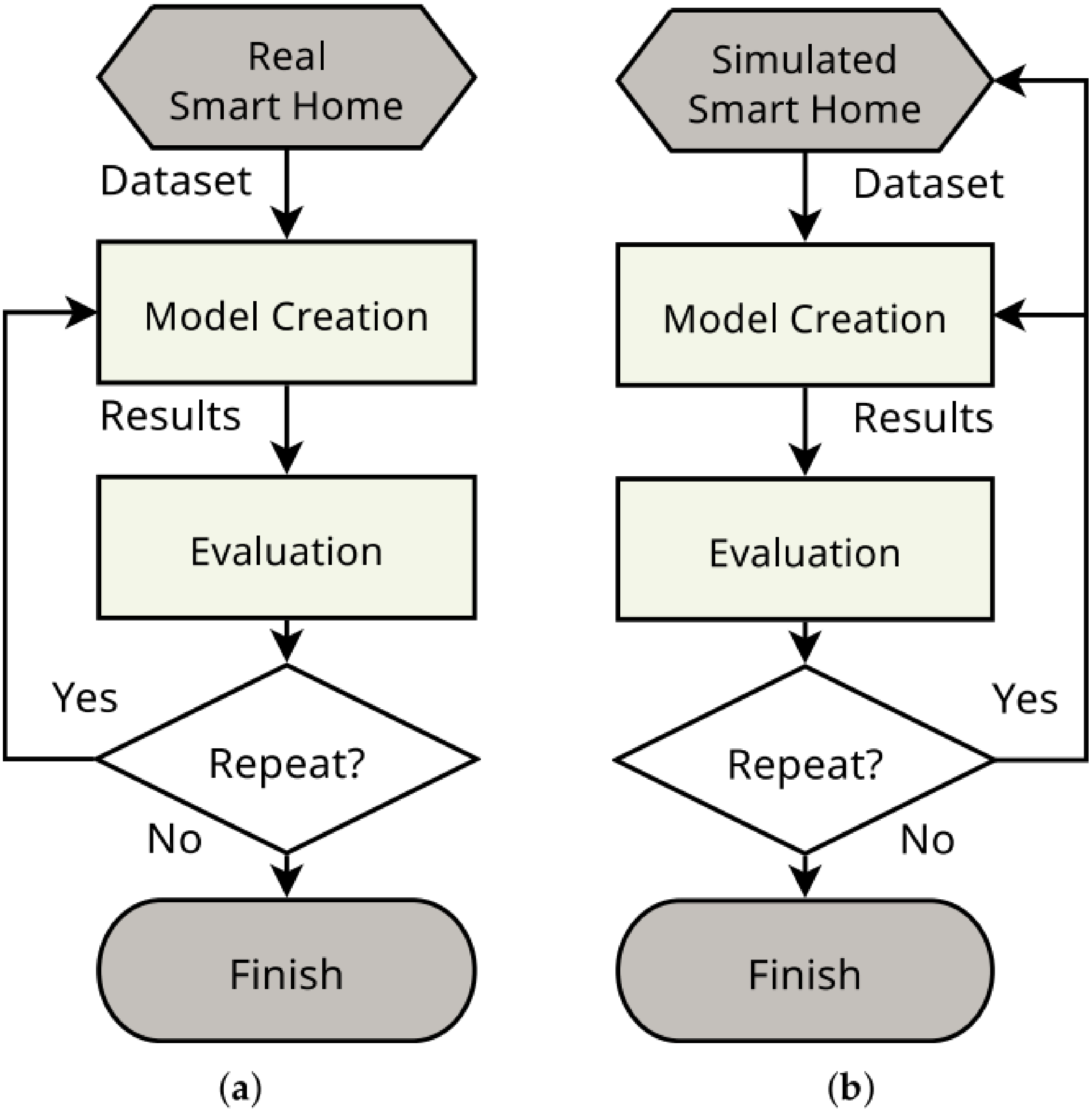


Figure 5.1 The workflow of smart home test beds (a) Real; (b) Simulated (Alshammari et al., 2017)

According to Synnott, Nugent and Jeffers (2015), there are two main approaches that categorise the smart home simulation tools;

i) Model-based

ii) Interactive based approaches.

A third hybrid approach that combines both model-based and interactive simulation is proposed by Alshammari *et al.* (2017). This approach combines interactive data generation for short periods of time which are in turn aggregated into full days with model-based approach.

5.3.1 Model-based Approach

The model-based approach employs pre-defined models of activities in the generation of synthetic data. The order of events, the probability with which they occur and the duration of time for activity is explicitly specified by this model. These events can be real time or not. The main advantage of this approach is that it enables a researcher to generate huge amounts of datasets within a short time. The quality of the generated data is dependent majorly on the quality of the modelling technique used. However, a drawback of this approach is that it does not allow for capturing of sophisticated interactions or unforeseeable accidents which are quite usual in real smart homes (Mendez-Vazquez, Helal and Cook, 2009). It is also claimed by Renoux and Klügl (2019) that because model based simulators mostly use large granularities of event and activities, it may be difficult to develop (model) fine-grained events and activities. Finally, for the very reason that these approaches are in most cases scripted entirely, it means that if a researcher wishes to simulate many days of data, he/she will need each day scripted independently. These drawbacks make is unsuitable for this approach to be used in application areas like e-health which require data for several days/weeks so as to detect patterns that are long-term.

Some of the simulators that use model-based approach are discussed below:

PerSim 3D

Developed by Lee *et al.* (2015), PerSim 3D is a smart home simulation tool whose aim is to generate datasets that are realistic of the inhabitants' activities in complex scenarios. A Graphical User Interface (GUI) is provided by this tool to visualise inhabitants' activities in a 3-Dimensional (3D) view. Users of the tool are able to define contexts of acceptable values per given set ranges that suit their projects. This tool, however, is not available freely for public use.

SIMACT

This tool is also a 3-Dimensional simulator designed to recognise activities of a smart home inhabitant. Developed by Bouchard *et al.* (2010), SIMACT contains scenarios that are pre-recorded and captured from experiments to aid in the generation of datasets to recognise ADLs. The tool is an open source and has 3D capabilities developed with Java Monkey Engine (JME).

DiaSim

A tool created by Bruneau, Jouve and Consel (2012), DiaSim is developed using Java for ubiquitous systems of computing which can work with varied smart homes. The tool has an editor that enables a researcher to create different scenarios in virtual environments.

Context-Aware Simulation System (CASS)

The aim of the CASS tool is to generate context information and testing context awareness applications in virtually built smart homes. Developed by Park *et al.* (2007), CASS allows a researcher to create different contexts with set rules. An example of the rules that could be set is like turning the heating on once the temperatures go down to a particular range. Conflict of rules are able to be spotted by this tool and is able to determine the best position to place the sensors. The GUI provided by this tool offers a 2D visualisation of the virtual smart home.

Context-Awareness Simulation Toolkit (CAST)

CAST is a simulation tool developed by Kim *et al.* (2006), it is designed for testing context-awareness applications, it also offers visualisation of different contexts. There is generation of context-awareness information of ADLs of smart homes. This tool is not publicly available and is under proprietary technology.

5.3.2 Interactive Approach

Unlike the model-based approach, the interactive approach has the capability of capturing interactions of ADLs in an interesting and in finer details (Alshammari *et al.*, 2017). The approach employs almost an avatar which can be controlled by the researcher, simulated or human participants. The avatar (which has sensors and/or actuators) is able to move and interact with the simulated virtual smart home environment. The interactions could be carried out both in an active or passive way. An example of a passive interaction is where you install a pressure sensor on the floor such that when the avatar walks on it, the sensor detects it and emits a signal. In the case for active interactions, good examples could be an action to turn the lights on/off or sensors when a door is opened. A drawback for this approach, however, is that it takes a lot of time to generate adequate datasets, this is because all interactions must be captured in real time. As further claimed by Renoux and Klügl (2019), an interactive approach is majorly restricted to in situations where data generation is only done for a short period of time. This means that the focus is only placed on specific activities.

Some of the tools that use interactive approach are highlighted below:

Intelligent Environment Simulation (IE Sim)

IE Sim tool developed by Synnott *et al.* (2014), is used for generation of datasets by capturing both normal and abnormal ADLs of smart home inhabitants. It provides a 2D GUI for researchers to design their virtual smart home environments. The tool allows the researcher to add more sensors to the virtual smart home. An avatar is then used for simulation to capture the ADLs data. This tool is not publicly available for use.

Residential environment and Ambient sensor simulator

The authors Ariani *et al.* (2013) developed a smart home simulation tool which captures the inhabitant's activities/interactions through ambient sensors. A researcher designs the virtual smart home using a map editor that is incorporated in the tool for 2D outputs. Ambient sensors are then added to the virtual smart home by the researcher.

UbiREAL

This java based simulation tool designed by Nishikawa *et al.* (2006) enables the development of pervasive (IoT) applications in a 3D virtual smart home. A researcher is allowed to simulated smart devices' communications and operations at the network level.

V-PlaceSims

This simulation tool by Lertlakkhanakul, Choi and Kim (2008) allows a researcher to design a smart home from a floor plan. Through a web interface, multiple users are able to interact with the virtual smart home environment. The focus of this tool is the improvement of the designs and management of the smart home.

5.4 Selected Tool

A survey conducted by Synnott, Nugent and Jeffers (2015) to analyse simulation tools that are in existence for the purposes of generating data in a smart home environment revealed that; because of the cost of the technology of the sensors, limitations for the availability of the smart home devices, considerations for time, and configurations of the sensors for optimum results, smart home simulation tools serve a valuable role for smart home research. It is further identified by the authors that many of the available simulation tools emphasise more on the applications that are context-awareness capabilities rather than to generate representative datasets. Additionally, the available simulation tools lack a feature that provides support for multiple inhabitants.

This research adopted the Open Smart Home Simulator (OpenSHS) simulation tool for the design of the virtual smart home and generation of datasets. OpenSHS was designed by Alshammari *et al.* (2017).

5.4.1 Reasons for OpenSHS

This simulation tool has been selected because it is an open-source application and therefore offers the flexibility for modification and scalability depending on the needs of the researcher. The tool also enables a researcher to simulate the Activities of Daily Living (ADLs) in a 3-dimensional virtual environment.

OpenSHS incorporates a hybrid approach in generating datasets by employing both the model-based and interactive approaches. The tool enables quick and large generation of datasets through a mechanism offered by tool where recorded ADLs are replicated. The replications have fine-grained details as the activities are captured in real-time, similar to the interactive approaches. OpenSHS has the flexibility to add different activity labels that can be customised by the researcher and tailored to their needs. It also has a fast-forwarding feature which facilitates the simulation of long inactivity periods (Alshammari *et al.*, 2018b).

A summary of the five main advantages of OpenSHS are:

i) Scalability: the tool can be easily extended to accommodate additional kinds of new sensors and smart devices through a provided smart devices library.

ii) Accessibility: the tool is designed to work on any platform which makes it easier for researchers to access and use the tool on any operating system.

iii) Interactivity: the tool offers a type of real-time style of how the interactions between the participants and the smart home are captured. This enables generation of datasets that are richer.

iv) Reproducibility: OpenSHS being an open-source tool, it allows researchers to reproduce datasets which can be used to validate activities of other research studies.

v) Flexibility: the tool allows researchers to simulate different smart home scenarios that relate to their needs. The ability to modify and customise the tools adding or removing sensors and smart devices to meet the research desires.

5.5 Data Acquisition in a Simulated Environment

A variety of data exists, this raw data may be acquired directly or may be observed from different systems. In a methodological way, the aim of data science approaches is to be able to process and clean the raw data for preparation for further analysis.

As noted earlier, it is not always feasible to acquire data from real world scenarios due to limited access to the systems that we may be interested in. It is due to the factor that simulation comes in handy in the creation of artificial systems that mimic the real-world scenarios. As stated by Lorig and Timm (2020), in simulated systems, the execution and investigation can be done in a speed that could be slow or accelerated depending on what the researcher needs. The simulated systems pose no restrictions for access and can be easily restored to the original state without ant expenses encountered.

In the real-world systems, the conventional acquisition of data process is dependent on the data being collected and exported for example from a data warehouse. The acquired data then undergoes a step-by-step methodology of processing, cleaning, exploration, and analysed in a quantitative way to derive qualitative insights which aid in the decision-making.

In the contrary to the conventional way, the data acquisition approach in a simulated environment only targets some small data of a specific set as stated by Kuhl *et al.* (2006). This data may include the system/device information needed for the creation of the process model. The knowledge or experience of the participants to be incorporated in the creation of the simulation model. The simulation experiments are carried out to acquire the data that is desired. Renoux and Klügl (2019) states that this data can be used for testing augmented living algorithms and/or for identifying patterns for learning rules on activities of inhabitants of smart homes.

Figure 5.2 is an illustration of data acquisition in simulated and real-world scenarios.

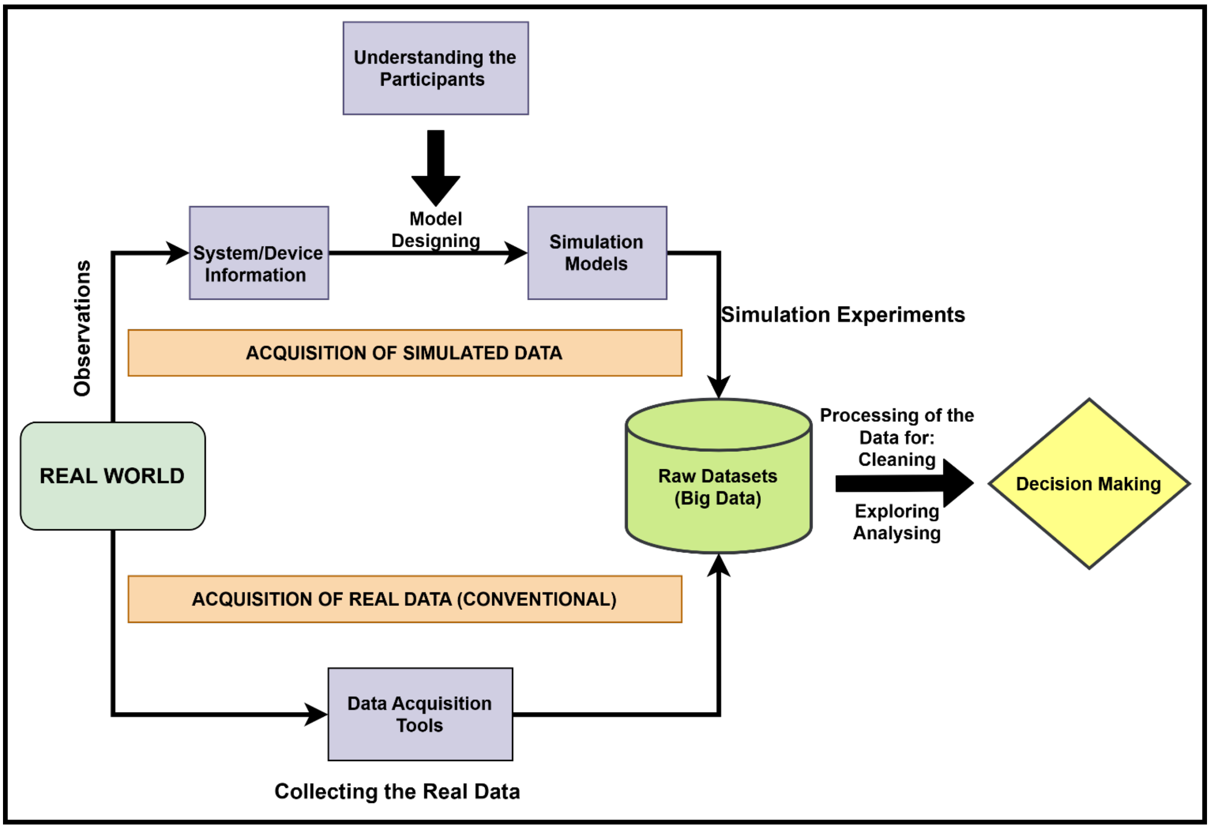


Figure 5.2 Conventional and simulated data acquisition process

5.6 The Design of Open Smart Home Simulator (OpenSHS)

There are three main phases involved in the design of the OpenSHS.

They are:

i) Design Phase

ii) Simulation Phase

iii) Aggregation Phase

These phases are described as follows:

i) Design Phase

A researcher designs a virtual smart home environment by importing smart devices, assigns labels to the activities, and designs the contexts.

The floor design is done based on the needs of the researcher in relation to the experiments being carried out. Smart devices and sensors are then imported into the smart home from a library provided by OpenSHS simulator. The devices and sensors provided by the tool are:

- Light controllers
- Door sensors
- Appliances (Television, Fridge, etc.)
- Sensor Devices (floor sensors on carpets, couch and/or bed)
- Lock Devices

Labels are attached to the activities, for example, these labels could be; eat, sleep, work, personal or other, this depends entirely on the requirements of the researcher.

The final part of the design phase is the design of the contexts to be simulated. The context in this case would be in relation to the time frames that a researcher wants to use in the simulation; these time frames would be like morning, afternoon, or evening. The initial states of the devices are specified for every context.

This is illustrated in Figure 5.3:

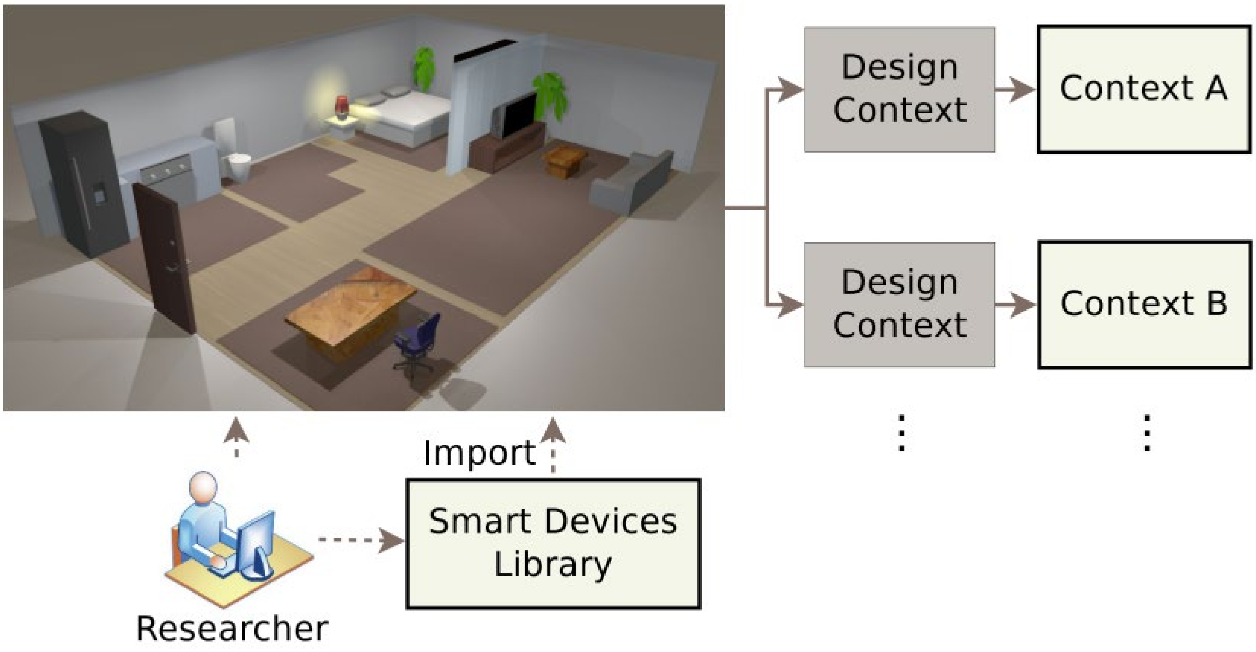


Figure 5.3 The Design Phase of OpenSHS

ii) Simulation Phase

Contexts

The Open SHS has a total of four contexts: a participant has two contexts in the day (morning and evening). and the other two contexts are in the week (weekday and weekend) for simulation purposes.

The OpenSHS module is started, and the researcher specifies which context (morning, afternoon, or evening) they wish to simulate. The tool has a default start time; however, this can be changed according to the needs. The default state of the sensors and the 3D positioning of the avatar is set. The participants then simulate the ADLs in the context set by themselves. The outputs of the sensors and different devices' states during the time of the simulation are captured and deposited in a temporary dataset for storage. The sampling rate is set to one second by default but can be reconfigured to desired rate. At the completion of a simulation by a participant, the application control is sent back to the main module to start the simulation of another context.

The purpose of this simulation phase is for capturing granularity of the realistic interactions of the participants.

The flow of the simulation phase is as shown in Figure 5.4:

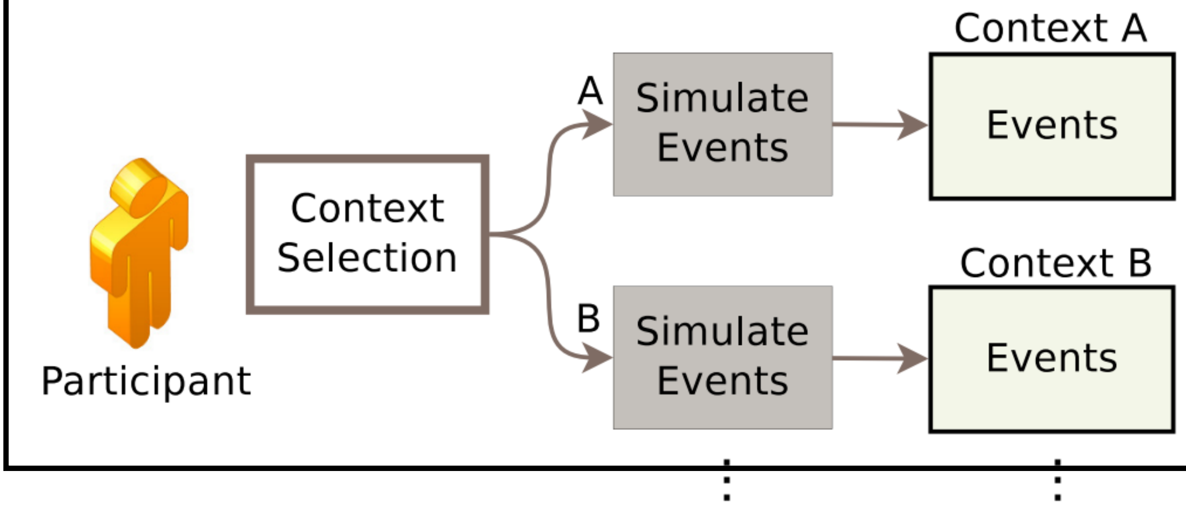
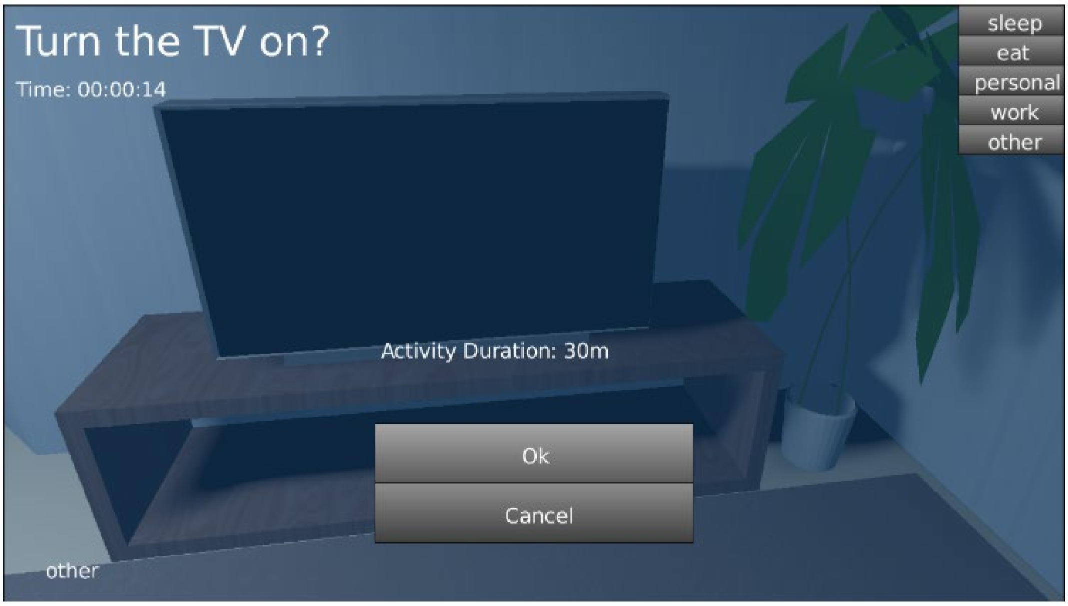


Figure 5.4 The Simulation Phase of OpenSHS

The process of capturing these fine-grained activities in a prolonged set of time may be difficult, to solve this, OpenSHS offers a solution to solve this problem by employing a mechanism named fast-forwarding. This allows for a participant to have control of the activities in relation to how long the activity should last (time span). In this case, say a participant wanted to watch television for a period of 30 minutes. If the participant did not want to go about the whole activity in real-time, he can therefore set a time span of 30 minutes. The tool then copies and repeats the sensors and devices' existing states in the specified time span.

Figure 5.5 illustrates this:



Turn the TV on?

Time: 00:00:14

Activity Duration: 30m

Figure 5.5 Fast Forwarding for an Activity in OpenSHS

iii) Aggregation Phase

At the end of the simulation phase, the generated events (sample activities for every context) are aggregated to produce the final dataset. As shown in Figure 5.6; the purpose for aggregation is for the production/generation of datasets that are large for a short time of simulation. An algorithm for replication of the simulation phase output is developed by drawing appropriate samples for every designated context.

As it is not feasible for a participant to sit down the whole day simulating their ADLs, OpenSHS has implemented an Events Replication mechanism. In this case, the participant only simulates real time sample activities in a given context and these events are replicated to produce rich datasets.

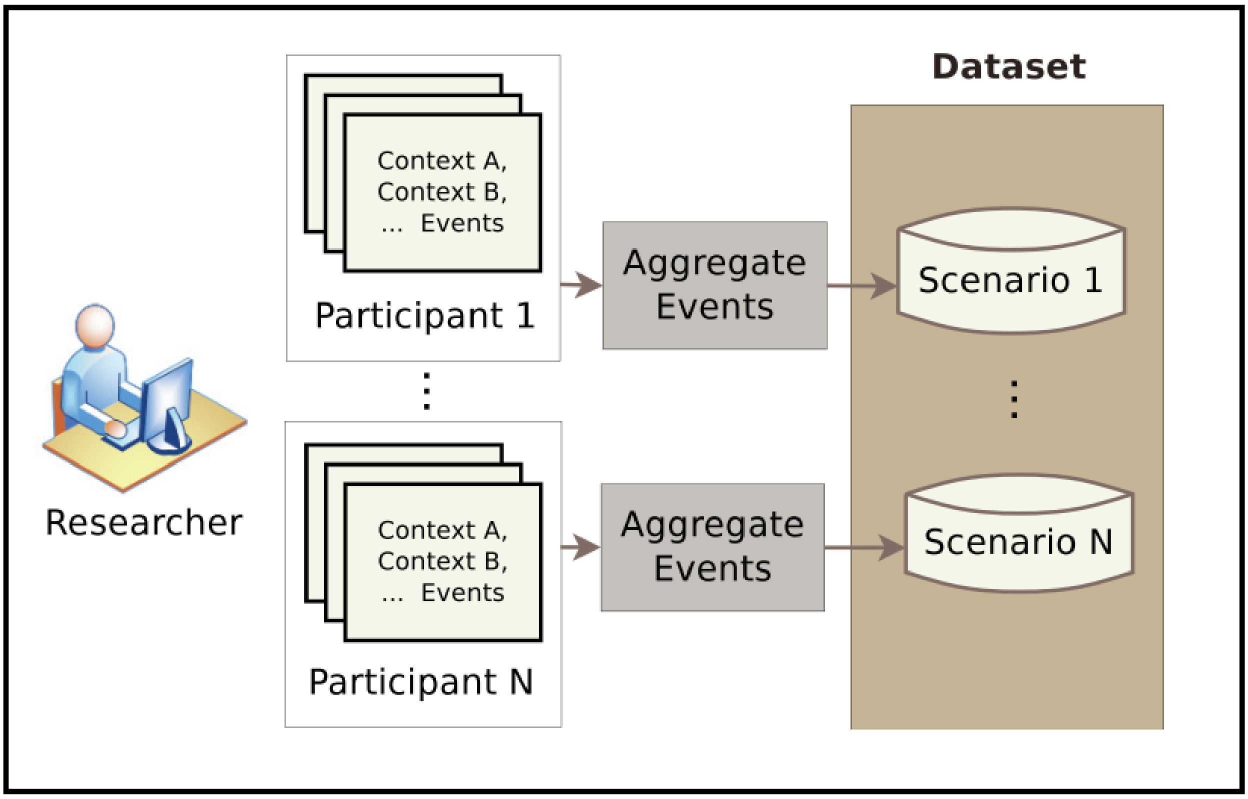


Figure 5.6 Aggregation Phase of OpenSHS

Execution of Experiments

The designed smart home in the OpenSHS consists of a bedroom, kitchen, bathroom, living room and an office.

The smart home has been fitted with binary sensors as seen in Table *5.1* on page 163. These sensors are in the state of either ON (1) or OFF (0) and are divided into two groups of either being passive or active. The passive sensors do not need a participant to explicitly interact with them. However, they respond to the position and/or the movement of the participant, for example, a floor sensor is activated when the participant steps on it.

The active sensors are the ones that change their state when the participant explicitly interacts with them. These kinds of interactions are for example turning on the TV, switching the light on, or opening the door.

The activity labels included in the datasets are sleep, eat, personal, work, leisure, and other.

This is as shown in Figure 5.7:

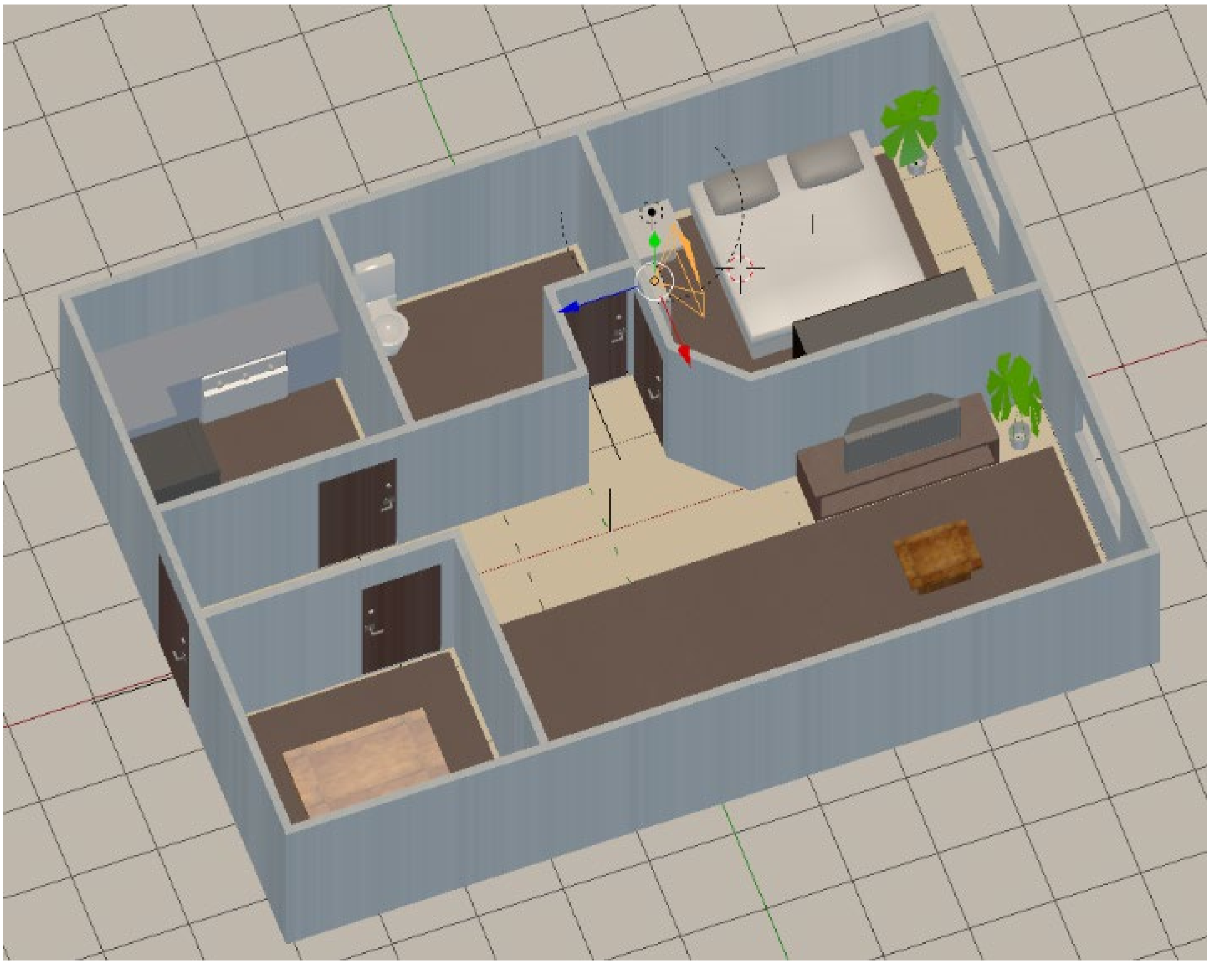


Figure 5.7 Aerial View of the OpenSHS

The majority of the existing smart devices' state is in binary by nature (Alshammari *et al.*, 2017), this consists of things that can be opened and closed or rather anything which has the capability of detecting the presence or absence of an object. It is noted that the binary simplification of the device state cannot fully cover all the states that smart devices can have, however, to aid the usefulness of anomaly detection, this simplicity serves the purpose. An example can be said of the state of a television; it could be ON playing a particular channel station, it could as well be on a sleep mode to preserve power, or it could actually be turned OFF completely. All these states help in detecting the daily patterns of an inhabitant of a smart home.

In the event that there is a need for implementing a middleware to add a service for detecting anomaly, the threshold can be set manually or by a technical to set the threshold on every set of devices. The threshold set turns the output of the sensors into a binary form.

5.7 Anomaly Activities for Forensic Analysis

The definition of anomaly, in some context, can be deemed clear and can be easily quantified. The easier example is the tracking of a patient's heart rate where the heartbeat counts can either be quantified as normal or abnormal. However, in the context of a smart home, it is difficult to define and quantify the anomaly behaviour of the inhabitant. In trying to define and quantify these behaviours may be tedious and prove subjective because inhabitants vary from each other (Alshammari *et al.*, 2018b).

5.8 What is considered an Anomaly, Attack or Error?

The author, Oriwoh (2015) has defined and distinguished these three aspects. The author defines an ***anomaly*** in a smart home as an occurrence which does not follow the expected or an already-known outcome or pattern. The author further describes an anomaly as an element or a set of elements within a group of similar or related elements that does not conform to the normal pattern of occurrence. In a setting of a smart home which only has one occupant, an anomaly can arise when two people are present when there should only be one person at that particular time.

Typical examples given for a smart home anomaly are:

- A persistent network probing
- A member of the smart home occupants coming back home in unusual hours

An ***attack*** is defined by Oriwoh (2015) as an intentional (caused on purpose) malicious anomaly. A good example of this is the occurrence of persistence network probing that denies the Smart Home occupants' access to the network, i.e Denial of Service attack. Consequently, if a sensor detects a broken water pipe caused by freezing water, then this can be defined as an anomaly, however, if there is a physical destruction by an intruder, then that becomes an attack.

An ***error*** is defined as a failure or fault.

Other than the broad classification of anomalies into errors and attacks, Kumar *et al.* (1994) classifies anomalies into four classes:

- An Intrusion that is not an Anomaly i.e. **False Negatives**
- Non-Intrusion that is an Anomaly i.e. **False Positives**

- Non-Intrusion that is not an Anomaly i.e. **True Negatives**
- Intrusion that is an Anomaly i.e. **True Positives**

It is challenging to differential between attacks and errors in a smart home setup because the nature of humans is so unpredictable. The change of human behaviour may have a huge impact on how the Machine Learning algorithms learn the patterns. To therefore minimise the wastage of resources, Smart Homes' anomaly detection systems should be developed in a manner that minimises the responses triggered by False Positives. In the same breath, it should be noted that these anomaly detection systems should not be designed/tweaked extremely in such a way that in their attempt to reduce the False Positives, they allow for True Positives to pass through in the end.

5.9 Hypothetical Case Scenarios for Forensic Analysis

In relation to forensic scenarios, however, there are some activities that reveal anomalies that may lead to a forensic investigation in a smart home.

The focus is placed on the data that is generated through motion and/or sensors that are placed within the smart home setup.

This data therefore can be from

a) Devices

As there are a variety of smart home devices that can be implemented, it might prove difficult to extract meaningful data for forensic purposes. To this end, it is imperative that the forensic investigators exploit the configuration information of the environment (smart home) for the purposes of identifying the devices in place. This configuration information could give a clue on the name of the smart home, the setup of the rooms and the devices deployed in those rooms. This clue will enable the investigator to choose their points of target in relation to data and devices (Kim *et al.*, 2020).

As this research involves simulation, the simulated devices are already known, and their deployment is also explicitly done. Therefore, the forensic data sought will not necessarily be that from the devices. However, identification of devices deployed within a smart home setting is important in a digital forensic case as it helps the investigation.

b) *Motion/Movement*

The motion/movement data can be used in determining an invasion/intrusion into the smart home. The timestamps for these intrusion events can help distinguish allegations of theft and determine whether false claims were made by the smart homeowner.

The sensors placed on doors and carpets will reveal their states (open/closed) whenever they get attached to by other objects (in this case, intruders).

In the OpenSHS, the sensor data will be stored with all the activities that happened in the form of the sensor acceleration (active/inactive), sensor status (open/closed)

A scenario is envisioned whereby the sensor data is able to shed light on claims on whether a person (participant/intruder) of the smart home was indeed at a specific place at that very specific time. This data will be obtained from the sensors that are deployed within the OpenSHS simulator.

The combination of motion and sensor data can easily determine whether someone has entered or left the smart home.

c) *Other activities*

These other activities include those that appear abnormal to the daily happenings of that smart home.

These are as below:

- Main door (front) open; in case of a burglary investigation, the forensic investigators can establish whether the inhabitants left the door open, or the robbers hacked the door system to gain access.
- When the fridge door is left open; leading to food being spoilt.
- When the television is left on.
- When the lights are left on (bathroom and bedroom).
- When the wardrobe doors are left open.
- When the oven is left on for long time.

5.10 Normal Day to Day Activities

These are the daily activities happening in the smart home as captured by the OpenSHS simulator.

The smart homeowner has a daily routine for the seven days of the week (Monday to Sunday). He goes to work on Monday to Friday (he leaves the house at around 08:30 in the morning and returns at around 17:30 in the evening) and is in the house on weekends (Saturday and Sunday). The smart homeowner has a domestic worker who comes in the house Monday to Friday. The domestic worker arrives when the smart homeowner leaves for work and leaves when the owner arrives from work.

On a normal workday, the smart homeowner would start the day by waking up at around 07:00 in the morning. The following are the activities that he does before he leaves for work:

a) Get up from bed

b) Turn on the bedroom light (turn off on exit)

c) Open the bedroom door (close when exiting)

d) Open the bathroom door (close after using bathroom)

e) Turn on the bathroom light (turn off after use)

f) Use the bathroom (toilet and shower)

g) Back to the bedroom room after shower

h) Open the wardrobes to choose clothes (close it afterwards)

i) Head to the kitchen area

j) May use any kitchen facilities to prepare his breakfast

k) May head to the living room to have his breakfast there (may also just have it in the kitchen)

l) While in the living room having breakfast, he may watch the television

m) The smart homeowner leaves the house, and the domestic worker enters the house

Once in the smart home (this is normally between 08:30 and 17:30), the domestic worker is expected to do the normal house chores, these activities can be the following:

a) Tidy up all the rooms

b) Use the kitchen to cook (fridge and oven)

c) Arrange clothes in the wardrobes

On coming back from work, the homeowner relieves the domestic worker. The owner embarks on the normal evening activities namely:

a) Inspects the house to ensure the domestic worker has done the daily house chores

b) Sit in the living room watching TV

c) Have dinner (go to the kitchen, may open fridge to take food and use oven)

d) May use home office

e) Go to bathroom for shower

f) Go to bedroom for sleep

On weekends, the smart homeowner spends the time in house and may use his home office to work. The owner may also leave the house for his own social activities. The domestic worker does not come on weekends.

5.11 Proposed Forensic Case Scenarios

From the daily patterns created by the normal activities within the smart home, abnormal activities may arise that will require a forensic audit.

These abnormal activities will be those that deviate from the normal daily activities captured. These activities may include:

a) Smart homeowner not waking up at all on a workday (died in bed?)

b) The owner wakes up, goes to the bathroom but never comes out (fell/fainted in the bathroom?)

c) The owner goes to work but never returns home

d) The owner accuses the domestic worker of stealing his valuables in the home office or bedroom

e) The domestic worker gets in the house and just watches television the whole day

f) The domestic worker is accused of misusing the privilege, (ie, going to sleep in the owners bed the whole day, turning on all the lights and not turning them off thereby wasting electricity)

g) The domestic worker leaving the house and returning a few hours before the owner returns

All these claims constitute a forensic case which can be investigated to prove or disprove the claims.

5.12 Forensic Preparations for Smart Home Dataset Generation

By use of data aggregation mechanisms, logs and data which are potentially useful for the forensic investigation can be collected.

In this preparation stage, the following should be identified.

- All the events or activities making up the ADLs.
- All core activities and events of the smart home inhabitant.
- All the activities and events of the devices within the smart home.
- Location of storage for logs (establish whether they are local or in the cloud).

An attack on a smart home may not necessarily occur from a single source and may be target to many different areas within the smart home. Adequate preparation is a key strategy for first responders to address these types of attacks and to ensure that forensic investigators are able to collect as much relevant data/evidence as possible (Oriwoh, 2015).

In a real time, live investigation of a forensic case, information gathering is commenced immediately after the investigators receive an alert. As is the case, more information should be sought from victims (smart home inhabitants) regarding what might have happened or/and what might have triggered the event.

Once the nature of the investigation has been established, the scope of the investigation must now be established so that the investigation is narrowed down to the scope. The scope is determined by identifying the physical and the logical perimeters of the investigation. The identification of the scope helps in determining if the incident that occurred involved physical devices located within the smart home or if the devices were connected to externally. External parties may also be established in the scope; this especially in the event that external family members are involved and may therefore need to be contacted as far as the investigation is concerned. A perfect scenario is depicted as for example, a smart home inhabitant received an order of groceries from his/her supplier without his/her knowledge. The inhabitant may then contact the supplier denying the order only to find out that one of their friends or external family member ordered the groceries without informing them. It is therefore important that all the aspects (humans and devices) are explicitly identified. The identified scope also helps understand which laws are applied in the investigation (international or local).

The following should be identified:

- The type and number of incidents involved; establish their relationship and how they overlap between themselves.
- Any locations where aggregation and storage of the smart home data logs is done.
- The affected devices.
- The affected areas of the smart home (Kitchen, Living Room, Bathroom, etc).
- Identify the devices that can be disconnected (analysis can be done offsite if possible) and those that need to be analysed on site.
- For each affected device, distinguish the physical and logical evidence held on it.
- Identify the volatile evidence so that it is acquired first and soonest as possible during the investigation process.
- Consideration of the law to be applied, this should include the ethical and moral boundaries in place.

5.12.1 Dataset Generation

The OpenSHS was used in the acquisition of the IoT forensic dataset based on the described scenarios in Section 5.11 following the IoT forensic process developed in IoT Forensic Framework in Figure 4.1.

The dataset from the sensors is in binary form. The reasons for the binary form was given by Alshammari *et al.* (2017) as:

- Many sensors are designed with binary states. This means that it is either ON or OFF. For this thesis experiment, a sensor in standby mode is considered OFF.
- Binary state makes it easier for sensors to be implemented in available middleware.
- Machine Learning encoders work closely with binary form.

The OpenSHS has 29 sensors which form part of the part of the headers of dataset. The sensors are placed on carpets, doors, lights, bed, couch, fridge, oven, tv, wardrobes, among others as seen in Table *5.1*.

In addition to these sensors, the dataset also contains the activity column that describes what activity (eat, sleep, work, personal, other, or anomaly) being simulated.

Another column included in the dataset is the timestamps. This captures the time the sensor was activated and aggregated accordingly during the aggregation phase of the simulation process.

	Name	Type	Description	State
1	bathroomCarp	Binary	Bathroom carpet sensor	Passive
2	bathroomDoor	Binary	Bathroom door sensor	Active
3	bathroomDoorLock	Binary	Bathroom door lock sensor	Active
4	bathroomLight	Binary	Bathroom ceiling light	Active
5	bed	Binary	Bed contact sensor	Passive
6	bedTableLamp	Binary	Bedroom table lamp	Active
7	bedroomCarp	Binary	Bedroom carpet sensor	Passive
8	bedroomDoor	Binary	Bedroom door sensor	Active
9	bedroomDoorLock	Binary	Bedroom door lock sensor	Active
10	bedroomLight	Binary	Bedroom ceiling light	Active
11	couch	Binary	Living room couch	Passive
12	fridge	Binary	Kitchen fridge	Active
13	hallwayLight	Binary	Hallway ceiling light	Active
14	kitchenCarp	Binary	Kitchen carpet sensor	Passive
15	kitchenDoor	Binary	Kitchen door sensor	Active
16	kitchenDoorLock	Binary	Kitchen door lock sensor	Active
17	kitchenLight	Binary	Kitchen ceiling light	Active
18	livingCarp	Binary	Living room carpet sensor	Passive
19	livingLight	Binary	Living room ceiling light	Active
20	mainDoor	Binary	Main door sensor	Active
21	mainDoorLock	Binary	Main door lock sensor	Active
22	office	Binary	Office room desk sensor	Passive
23	officeCarp	Binary	Office room carpet sensor	Passive
24	officeDoor	Binary	Office door sensor	Active
25	officeDoorLock	Binary	Office door lock sensor	Active
26	officeLight	Binary	Office ceiling light	Active
27	oven	Binary	Kitchen oven sensor	Active
28	tv	Binary	Living room TV sensor	Active
29	wardrobe	Binary	Bedroom wardrobe sensor	Active

Table 5.1 OpenSHS Smart Home Sensors (Alshammari *et al.*, 2017)

The OpenSHS aggregation phase utilises an aggregation algorithm (embedded within the simulator) to merge the diverse scenarios produced by different participants into a single, unified dataset. This consolidated dataset is then commonly presented in a format compatible with Comma-Separated Values (CSV) files. An example of such a dataset is illustrated in Table *5.2*.

wardrobe	officeLight	kitchenDoorLock	fridge	bedroomDoor	bathroomCarp	...	Activity	timestamp
1	1	0	1	0	0	…	sleep	2021-03-01 07_55_16
1	0	1	1	0	0	…	leisure	2021-03-01 07_55_17
0	1	1	1	1	1	…	other	2021-03-01 07_55_18
1	1	0	0	0	0	…	personal	2021-03-01 07_55_19
0	0	1	0	0	1	…	anomaly	2021-03-01 07_55_20
1	1	0	1	0	0	…	eat	2021-03-01 07_55_21
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Table 5.2 A sample of the full generated dataset

The full generated dataset is published on GitHub repository (Lutta, 2023).

5.13 Conclusion

Setting up a fully-fledged real smart home for research purposes is costly as highlighted in the literature. The existence of smart home simulators like OpenSHS aid researchers in the field of IoT to easily come up with different scenarios that utilise these simulators to generate highly elaborate and representative datasets.

In conclusion, this chapter has provided an in-depth exploration of forensic analysis in smart homes, focusing on the integration of simulation techniques and the utilisation of the OpenSHS simulator. Through a comprehensive review of smart home simulation tools, the chapter highlighted the reasons for selecting OpenSHS as the preferred simulator for data generation and forensic analysis in smart home environments.

The methodology for data acquisition in simulated environments was meticulously outlined, emphasising the significance of generating realistic datasets for effective forensic investigations. The chapter delved into the features and capabilities of OpenSHS, showcasing its ability to simulate various smart home devices, sensor readings, and user activities. This simulator proved instrumental in creating accurate and reliable datasets that closely resemble real-world smart home scenarios.

The chapter further elucidated the process of creating hypothetical case scenarios within the OpenSHS simulator and generating corresponding datasets. These datasets were carefully crafted to encompass a wide range of activities, interactions, and events that investigators may encounter in real-world forensic analysis.

It is worth noting that the generated dataset was made publicly available on the GitHub repository, fostering collaboration and knowledge sharing within the forensic analysis community. This open approach promotes the advancement of smart home forensics and encourages researchers and practitioners to explore, evaluate, and refine forensic techniques.

In summary, this chapter has extensively examined the various aspects of forensic analysis in smart homes, with a particular emphasis on the integration of simulation techniques using the OpenSHS simulator. By thoroughly reviewing smart home simulation tools, detailing the methodology for data acquisition, highlighting anomaly activities, creating hypothetical case scenarios, and generating the datasets, this chapter has significantly contributed to the field of digital forensics. The insights provided through the simulations will help researchers in IoT forensics to tackle the complexities of smart home environments and enhance their ability to develop solutions that can investigate crimes, anomalies, and security breaches effectively.

CHAPTER 6. THE APPLICATION OF HI-SDR IN ANOMALY DETECTION

The previous chapter (CHAPTER 5) has enabled the generation of datasets through simulation. With the datasets, this chapter 6 explores the application of Hash Indexed Sparse Distributed Representation (HI-SDR) in anomaly detection within the context of IoT environments, particularly with the generated smart homes dataset. It delves into the concept of Sparse Distributed Representation (SDR) as a fundamental data encoding method in Hierarchical Temporal Memory (HTM) systems. The chapter introduces the HI-SDR encoder, its properties, capabilities, and the role it plays in representing multi-dimensional input data for anomaly detection. The chapter concludes by presenting the practical application of anomaly detection in IoT forensics, the legal implications, and the limitations that may exist for the forensic process.

6.1 Introduction

Sparse Distributed Representation (SDR) has been proposed by the Cortical Learning Algorithms (CLAs) to encode input data, which serves as the fundamental data structure in the Hierarchical Temporal Memory (HTM) theory (Ahmad and Hawkins, 2015; Alshammari, 2018b).

The encoder's task is to convert the input data, regardless of whether it is numerical, categorical, single-column, or multi-columnar, into a format that enables the HTM system to learn and recognise patterns. The quality of the encoders is vital for the overall performance of the entire system, as they function like our senses, translating visual, auditory, or tactile information into representations that our brains can process.

This research leverages the High Indexed SDR encoder developed by Alshammari *et al.* (2018a) to preprocess (transform) the simulated IoT forensic dataset generated from the OpenSHS simulator (see Section 5.12.1.1) before being analysed by the machine learning algorithms chosen in subsection 3.13 of this thesis.

6.2 Sparse Distributed Representation

An SDR serves as the fundamental information representation and a key component in every HTM system. This section provides a mathematical formalisation of SDRs and outlines the fundamental operations that can be applied to them.

The notations and definitions presented in this section are derived from the work of Ahmad and Hawkins (2015).

The following compilation comprises a collection of definitions and mathematical notations that will be utilised consistently throughout this research:

SDR: is a binary array with primarily zeros and a small proportion of ones (active bits). Typically, the active bits constitute around 2% of the array. The total number of bits in an SDR is denoted as *'n'*. An SDR *'x'* consists of an *n*-length array of binary components *'bi'*, where *'i'* represents the index of each component in the array. For instance, x = [b_0, b_1, ……, b_{n-1}]. The total number of components in *'x'* with a value of 1 is indicated by *'wx'*. The variable *'w'* is chosen as an abbreviation for '*width'*. The components with a value of 1 are referred to as active bits since they represent firing or active neurons. An example of an SDR *'x'* with *n = 10* and *w = 3* is shown below:

$$X = [0\ 0\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0]$$

It is not a must for the set of active bits *w* to be consecutive.

6.2.1 SDR Properties

The work of Alshammari (2018b) has elaborated on the properties of SDRs. There is an optimal value where the chance of having two identical SDRs is at its global minimum. The author shows that SDRs have good noise robustness which can be exploited to achieve several interesting capabilities. Another property of an SDR is that it can be recognised and identified correctly from a group of SDRs.

A collection of SDRs can be stored and retrieved with a high level of confidence. This further demonstrates the noise tolerance of SDRs when the parameters are appropriately configured within a reasonable range. Another crucial property of SDRs, known as the 'union property' in HTM literature, allows multiple SDRs to be combined into a single representation by performing a logical OR operation on all the SDRs. Due to the sparsity and inherent properties of SDRs, it is possible to store these SDRs without corruption as long as there are a sufficient number of bits available. (Alshammari, 2018a).

Operations on SDRs are exclusively performed on the active bits (*w*), resulting in a time complexity of *0*(*w*). The total size of the SDR (*n*) does not affect these computations. Due to the sparsity of SDRs, the number of active bits (w) is typically much smaller than the total number of bits (*n*).

6.2.2 SDR Encoders

According to Purdy (2016) a well-functioning HTM system must possess certain properties to produce good results. These properties are as follows:

- **Deterministic**: When given the same input, the resulting Sparse Distributed Representation (SDR) should always be the same upon repeated processing.
- **Fixed in dimensions**: The SDRs generated by the system should have a consistent number of total bits, maintaining a fixed dimensionality throughout.
- **Fixed in sparsity**: The SDRs should maintain a fixed number of active (1) bits, ensuring a consistent level of sparsity in the representations.
- **Capturing semantics**: Similar inputs should have an overlapping set of active bits in their respective SDRs, enabling the system to capture and represent the semantic similarities between inputs.

A deterministic encoder is crucial for an effective encoder within an HTM system. Without determinism, the system would struggle to recognise and learn the sequence of SDRs when the representations of original values change over time.

Preserving the dimensionality of the encoder's output is essential throughout the learning process. Many operations in the Spatial Pooler (SP) and Temporal Memory (TM) rely on bit-wise comparisons between successive SDRs. If the dimensionality of the SDRs changes over time, it can lead to incorrect calculations by the HTM system's components. Thus, maintaining a fixed number of bits is important.

Similarly, the sparsity of the SDRs, which refers to the ratio of active bits to the total number of bits, should also remain fixed. The choice of sparsity depends on the specific application but typically ranges from 1% to 35%, as suggested by Purdy (2016). This fixed sparsity ensures consistent calculations during the HTM system's operations.

Defining semantic similarity between inputs is a challenging task and highly dependent on the data type. For example, in the case of natural numbers, semantic similarity can be easily defined within a known value space. By specifying the minimum and maximum values allowed in the dataset, one can determine the semantic similarity between two natural numbers. For instance, if the value space ranges from 1 to 100, the numbers 1 and 2 would be considered semantically similar, while 1 and 100 would be the most dissimilar.

NuPIC offers a range of built-in encoders designed to handle different data types, which can be broadly categorised into numerical data (encodes numerical data types) and categorical data (encodes categorical data types).

6.3 The Use of SDR Encoders for IoT Forensics Datasets

To handle multiple dimensions or columns of input data in a Machine Learning model, the recommended approach is to encode each dimension separately using the previously mentioned encoding techniques. These individual encoded representations can then be concatenated into a single Sparse Distributed Representation (SDR) that is sequentially fed into the Machine Learning model.

However, working with a large number of dimensions can lead to a challenge known as "the curse of dimensionality," as described by Bellman and Dreyfus (2015). This issue arises when the learning model struggles to effectively learn from high-dimensional data. Increasing the number of training samples can be a potential solution, but it can be costly and impractical depending on the specific application.

A more practical approach is to reduce the number of dimensions by employing techniques such as Principal Component Analysis (PCA) or feature selection. PCA helps in identifying the most important components of the data, while feature selection techniques assign more weight to relevant features or eliminate irrelevant ones (columns) that have minimal impact on the Machine Learning model's accuracy.

Referring back to CHAPTER 5 in Table *5.1*, the dataset has twenty-nine binary sensors fed into the Machine Learning model, the question arises: how can we represent or encode this multi-dimensional input? In Machine Learning literature, the typical solution is to concatenate the outputs of several encoders into a single SDR. Scalar encoders can be utilised to represent the binary state of each sensor, along with categorical encoders if applicable. It is crucial to ensure that these encoders possess the properties mentioned earlier in Section 6.2.1 for optimal performance.

By using appropriate encoders and addressing the dimensionality challenge, it becomes possible to effectively encode and represent multi-dimensional input data within a Machine Learning model.

6.4 The chosen HI-SDR Encoder

This research adopted the Hash-Indexed Sparse Distributed Representation (HI-SDR) encoder developed by Alshammari (2018c). HI-SDR is designed to meet the necessary criteria for effective encoders, as outlined in Section 6.2.1. HI-SDR encoder exhibits determinism, meaning that if it receives two identical inputs, it will generate the same SDR for both inputs. Additionally, the sparsity level and the number of bits used by the encoder are fixed, satisfying the first three prerequisites for reliable encoders. However, the aspect that proves challenging for the encoders discussed in the prior section is the fourth property, which involves capturing semantics.

In order to fulfil the fourth property requirement, it was necessary to devise a method that ensured both unique encoding for each record and preserved the determinism and sparsity of the resulting SDR. The HI-SDR presents a solution by employing a hash function, which takes the input records and generates a digest (Alshammari, 2018c). The author utilised the generated digest to assign unique positions to *w*-bits within the resulting SDR. Whenever the same record is encountered again, the hash function will produce an identical digest, thus maintaining determinism and enabling the creation of distinct SDRs for every input configuration.

At a high level, the functioning of the HI-SDR encoder is illustrated in Figure *6.1*. The encoder operates by taking a complete dataset record, such as [0, 1, 0, 0, ..., 0], and passing it through a hashing function to generate a hash digest. This digest is then utilised to determine the placement of active bits within the SDR. The process involves extracting each digit from the digest and utilising it as the index for assigning the location of active bits, resulting in an outcome similar to what is depicted in the Figure *6.1*. The HI-SDR encoder incorporates several parameters, which are extensively elaborated upon in the next section.

The author (Alshammari, 2018b) conducted tests and experiments involving various hashing functions. Two key requirements considered were the speed of the function and the randomness of the resulting hash digest. To satisfy the speed requirement, cryptographic hashing functions were excluded, as they are intentionally designed to be slow for enhanced security. Instead, the author explored several non-cryptographic hashing functions, including the Python implementations of CRC32 and Adler-32. After thorough experimentation, the author ultimately selected the xxHash function by Collet (2015) for use in their study. This choice was driven by its superior speed and the desirable level of randomness exhibited by the generated digests.

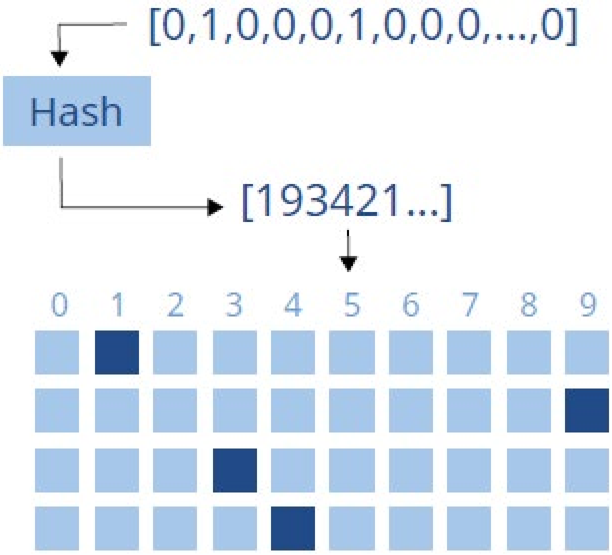


Figure 6.1 The HI-SDR encoder (Alshammari, 2018c)

The distinctive aspect of the Hi-SDR encoder is its ability to assign a unique representation to each activity (i.e., sleep, personal, eat, work). By employing varied placements of active bits throughout the input space, the encoder facilitates faster learning and enables clearer differentiation of inputs by the Machine Learning Model, surpassing the performance of standard encoders. This characteristic becomes evident when observing the encoder's operation (Alshammari, 2018b).

The HI-SDR encoder possesses the capability to provide a unique representation for any input, regardless of its density. Whether a record contains no active bits (e.g., [0, 0, 0, 0, ..., 0]) or is fully dense (e.g., [1, 1, 1, 1, ..., 1]), both will receive a distinct Sparse Distributed Representation (SDR) with consistent sparsity and a fixed number of active *w*-bits. This characteristic allows the encoder to meet the necessary properties of an effective encoder. It becomes particularly valuable when the encoder needs to operate in an unsupervised manner with an arbitrary number of sensors.

To assess the capabilities of the HI-SDR encoder, additional experiments were conducted by the author (Alshammari, 2018b), utilising the same set of their datasets.

The HI-SDR encoder achieved a score of 76.59%. The parameters for the HI-SDR encoder were set as follows: n = 600, w = 3, and p = 8. These experiments provided further insight into the potential effectiveness of the HI-SDR encoder. The author also noticed that the SDRs are more active because of the number of partitions (*p*). The details of the parameters of the HI-SDR and the algorithm are explained in Section 7.2 of this thesis.

6.5 The Proposed HI-SDR Approach

The innovative approach outlined in this context involves the utilisation of the HI-SDR as a transformative mechanism for datasets, enhancing their representations to make them more understood by machine learning models. By adopting the HI-SDR technique, the dataset undergoes a profound reconfiguration that enhances its suitability for comprehension and utilisation by various machine learning algorithms. This approach not only improves general data understanding but also plays a pivotal role in anomaly detection, a critical aspect of modern data analysis.

SDR, at the core of this methodology, holds remarkable advantages that ripple across various applications, as exemplified in Section 6.3, particularly within the domain of smart home datasets. SDR encoding offers a unique way of representing data, wherein information is conveyed through patterns of active and inactive elements. This approach stands in contrast to conventional methods that often involve dense representations where each element carries independent meaning.

One of the primary merits of SDR is its inherent robustness against noise and distortions. Due to its distributed nature, minor corruptions in the data have limited impact, making SDR particularly adept at handling real-world data that might suffer from imperfections during collection or transmission. This noise resilience is of paramount importance in the context of smart home datasets, where sensor readings or inputs can occasionally be erroneous due to environmental factors or technical glitches, which can be indicative of anomalies.

Furthermore, the HI-SDR technique augments the utility of SDR by introducing a structured yet efficient indexing mechanism. Hashing provides a means to map complex data patterns onto fixed-length representations, facilitating quicker retrieval and comparison of data instances. This is crucial in scenarios like smart homes, where rapid decision-making based on real-time data is pivotal. By incorporating hash indexing, the HI-SDR approach not only enhances the interpretability of transformed datasets but also provides computational advantages, enabling expedited querying and processing — essential for timely anomaly detection.

The significance of employing HI-SDR within smart home datasets stems from the multifaceted nature of such data. Smart homes generate a diverse array of data, spanning from occupancy patterns and energy consumption to device interactions and environmental variables. This inherent heterogeneity demands a representation that can encapsulate these nuances without succumbing to data overload. SDR, in its sparsity and resilience, aligns perfectly with this requirement, and the Hash Index extension further bolsters its applicability by introducing a structured framework for dealing with anomalies.

The proposed HI-SDR approach introduces an innovative solution for enhancing dataset representations, making them more compatible with machine learning models. The advantages of leveraging SDR are underscored by its noise resilience and capacity to capture intricate data patterns. These benefits are magnified in the context of smart home datasets, where the diverse and dynamic nature of data necessitates a flexible yet efficient representation scheme. Through the incorporation of hash indexing, the HI-SDR approach not only enhances data interpretability but also empowers streamlined data handling, making it a compelling avenue for advancing machine learning applications, especially in the sphere of anomaly detection within smart homes and beyond.

6.5.1 The Proposed Approach in Context

The proposed approach is depicted in Figure 6.2. Upon the preparation of the generated IoT forensic dataset (see Section 5.12.1.1), several critical steps were taken to ensure accurate results. This involved careful preprocessing and a comprehensive approach to both training and testing the machine learning models, with a primary focus on mitigating false alarms and enhancing the efficacy of anomaly detection. The dataset has a total of 524,287 records which were split into training and testing with a ratio of 70:30. The two state-of-the-art machine learning algorithms used for the experiments were Isolation Forest and OCSVM from the four reviewed algorithms in Section 3.13. These were unsupervised, meaning that they do not require any labelled data. The processed data is fed into these state-of-the-art machine learning algorithms to produce anomaly scores. The anomaly scores obtained from these two algorithms are recorded for performance evaluation in the experiments.

The experiment is then repeated, with the same training and testing data split. However, in these subsequent experiments, the dataset is then transformed by encoding using the HI-SDR encoder, the generated SDR data is fed to the state-of-the-art ML algorithms to produce the anomaly scores. To ascertain the effectiveness of the applied models, a robust evaluation methodology is adopted. This entails measuring performance across four well-established evaluation metrics: Accuracy, F1 Measure, Precision, and Recall, all elaborated upon in Section 7.3. These metrics collectively provide a comprehensive view of the model's performance – accuracy quantifies overall correctness, the F1 Measure offers a balanced view of precision and recall, while precision and recall individually shed light on false positives and false negatives.

The HI-SDR algorithm consists of two distinct components: the hashing part and the SDR construction part.

In the first part, the array containing sensor readings is passed through a hashing function to generate a hash digest. The authors (Alshammari, 2018b) tested multiple non-encryption hashing functions, and the results indicated that the xxHash function (Collet, 2015) yielded the most favourable outcomes when applied to the binary datasets.

The second part involves constructing the SDR based on the obtained hash digest. The algorithm below provides the pseudo-code for the implementation of the construction algorithm. This algorithm accepts four parameters, namely:

- "***hash***": This represents the hash digest generated in the hashing part.
- "***n***": This parameter denotes the total number of bits in the SDR.
- "***w***": This parameter represents the desired number of active bits in the SDR.
- "***p***": This parameter indicates the number of partitions to be used in the construction process.

By specifying these parameters, the algorithm can effectively construct the SDR representation.

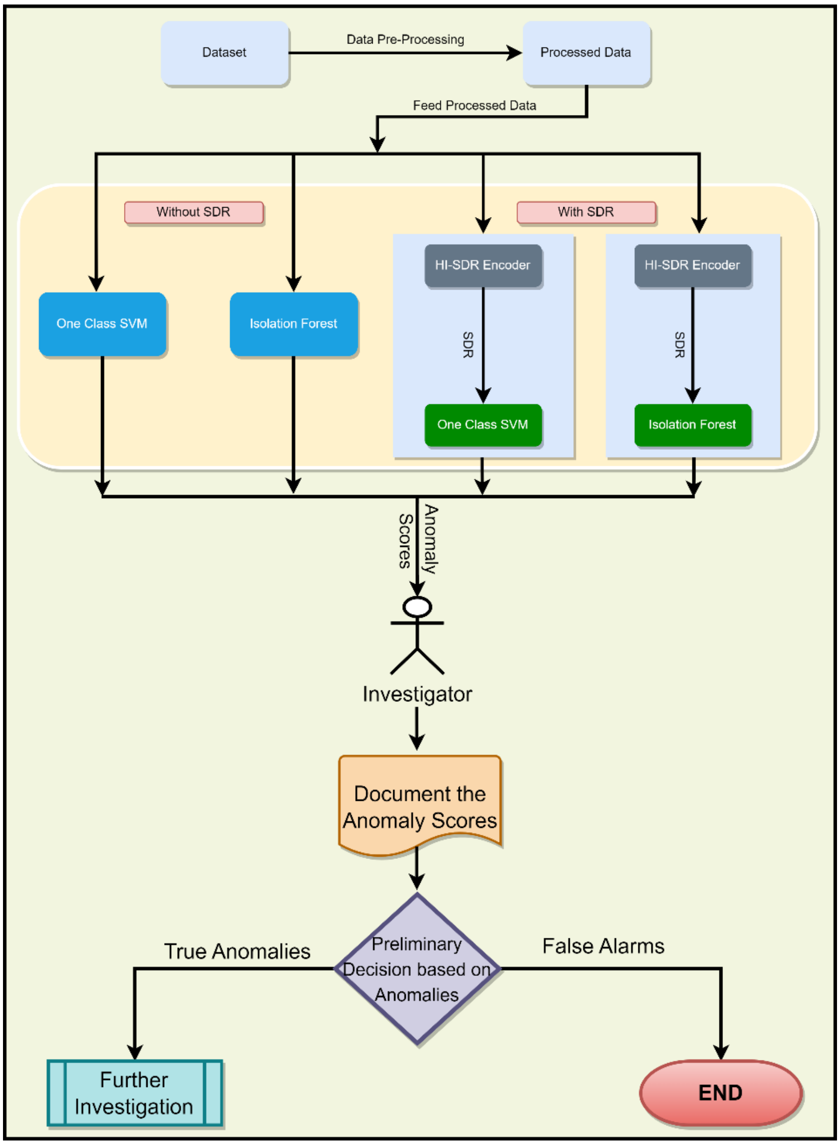


Figure 6.2 The Proposed Approach using SDR

The algorithm can plainly be explained as follows:

Algorithm: HI-SDR Encoder

Inputs:

- sensorReadings: Array of binary dataset

- **n**: Total number of bits in the SDR

- **w**: Number of active bits in the SDR

- **p**: Number of partitions

Output:

- SDR: Sparse Distributed Representation

Procedure:

1. Hashing Part:

1.1. Compute hashDigest using xxHash on the binary dataset.

2. SDR Construction Part:

2.1. Initialise SDR as an array of size ***n*** with all bits set to 0.

2.2. Divide the hashDigest into ***p*** partitions.

2.3. For each partition:

2.3.1. Convert the partition into a decimal value, partitionValue.

2.3.2. Calculate the position in the SDR as position = partitionValue % n.

2.3.3. Set the bit at position in the SDR as active (1).

3. Randomise *w* Active Bits:

3.1. Create a list, activeBits, of size n with all indices.

3.2. Shuffle the activeBits list randomly.

3.3. Select the first w elements from the shuffled activeBits list.

3.4. For each selected index, set the corresponding bit in the SDR as active (1).

4. Return the generated SDR.

The pseudocode for the SDR construction is in Algorithm 6.1:

Algorithm 6.1: SDR construction algorithm.

```
procedure CONSTRUCT-SDR ($hash, n, w, p$)
    $SDR \leftarrow [0] * n$                              ►SDR is n length zero array
    $skip \leftarrow$ INT ($n/partitions$)
    $hashDigits \leftarrow$ STR ($hash$)                  ►Converts the hash digits to string
    if $w > (skip/10)$ then                               ►Divide by 10 because there are 10 digits
        Raise ValueError
    end if
    for $d$ in $hashDigits$ do
        $i \leftarrow$ INDEX($d$)
        if $i == p$ then
            break
        end if
        $ri \leftarrow$ INT ($d$) + 1                     ►Calculating the relative index position
        $pct \leftarrow$ FLOAT ($ri$)/10
        $ri \leftarrow$ ROUND ($skip * pct$) $- 1$
        for $j$ in RANGE ($w$) do
            $diff \leftarrow (w - 1)$
            $SDR[ri + (i * skip) + j - diff] = 1$
        end for
    end for
    return $SDR$
end procedure
```

6.6 Practical Forensic Application of the Proposed Approach

The proposed approach leverages on anomaly detection derived from experimental analysis of the IoT forensic dataset generated. For this research and for forensic purposes, the generated IoT forensic dataset is treated as the collected evidence from the simulated smart home environment.

This research proposed to semi-automate the examination and analysis of the IoT investigation process as explained in Section 4.6. The logging of files and classification steps are equated to the data processing where the dataset is pre-processed (see the proposed approach in Figure 6.2) before being analysed (in this case, encoding then fed to the models). The anomaly scores are then leveraged to ascertain the degree of deviation from the normal behaviour.

The final part of the proposed approach (see Figure 6.2) is on forensic decision making by forensic investigation team. It is critical to understand how anomaly detection could help investigators in their investigation process. This application must be undertaken in strict adherence to the legal regulations established within the specific jurisdiction where it is being employed. Moreover, it's imperative to recognise that this proposed approach, involving the integration of anomaly detection for IoT forensics, should not be solely relied upon by forensic investigators. Moreover, it's imperative to recognise that this proposed approach, involving the integration of anomaly detection for IoT forensics, should not be solely relied upon by forensic investigators.

While anomaly detection is a powerful tool for flagging unusual patterns of behaviour, it should serve as an initial step rather than a conclusive determination. The anomalies detected should be considered as prompts, triggering further investigative efforts. The investigators must proceed to gather concrete evidence to support their observations. Relying solely on anomaly scores without substantial evidence could lead to erroneous conclusions. Additionally, the contextual and situational nuances surrounding the deviations must be taken into account. Anomaly detection might spotlight activities that, on the surface, appear suspicious, but upon closer examination, could have innocuous explanations. These subtleties underscore the importance of coupling technological insights with human judgment and traditional investigative methods. In essence, anomaly detection's role lies in augmenting investigative efficiency by highlighting potential leads and deviations from established norms. However, it's pivotal to recognise that it is merely a steppingstone to a comprehensive investigation. Adhering to legal protocols and exercising sound investigative practices ensures that anomaly detection contributes substantively to the pursuit of truth while preventing undue reliance on potentially misleading data.

6.6.1 Anomaly Detection for the Forensic Scenarios

Anomaly detection can indeed be a valuable tool for identifying suspicious or unusual activities within a smart home environment. Going through each proposed forensic case scenario in Section 5.11, a discussion can be made on how anomaly scores produced by ML models can assist forensic investigators.

Smart Homeowner Not Waking Up on a Workday (Possible Death):

In the normal routine, the smart homeowner wakes up around 07:00 on workdays. An ML model trained on this routine would assign a certain anomaly score to this activity if it doesn't occur as expected.

If the homeowner fails to wake up, this could be an indicator of an unusual event like a medical emergency or even death. High anomaly scores in this context would trigger an alert for further investigation.

Owner Goes to Bathroom but Doesn't Come Out (Possible Injury/Faint):

After waking up, the homeowner typically goes to the bathroom and then proceeds with other activities. If the bathroom door remains closed for an extended period without other activities occurring, an anomaly score would increase.

An elevated anomaly score here might suggest an accident, fall, or medical issue. It could help investigators identify a potential problem within the bathroom.

Owner Doesn't Return Home After Work:

The owner usually returns home around 17:30 after work. If this activity doesn't occur, the ML model would assign a high anomaly score.

A significant anomaly score here could indicate an unusual event like a car accident or any incident preventing the owner from returning. This could prompt investigators to check the owner's whereabouts.

Accusation of Valuables Theft:

The activities of the owner and the domestic worker in areas like the home office and bedroom are part of the normal routine. Deviations, such as unexpected access during certain times, might lead to higher anomaly scores.

If valuable items go missing, analysing anomaly scores for access to those areas during non-routine times could help identify potential suspects or establish patterns of unusual behaviour.

Domestic Worker Watching TV All Day:

If the domestic worker deviates from their usual activities (tidying up, cooking, etc.) to just watch TV, this behaviour would receive a high anomaly score.

An elevated anomaly score could suggest uncharacteristic behaviour, helping investigators focus on unusual patterns that might indicate neglect of duties or unauthorised activities.

Domestic Worker Misusing Privileges (Wasting Electricity):

Turning on all lights and not turning them off is a deviation from normal behaviour. Anomaly scores would reflect such changes in patterns.

High anomaly scores in this case could highlight power wastage or unauthorised behaviour. Investigating patterns of energy consumption could help confirm the claim.

Domestic Worker Leaving and Returning Hours Before Owner:

The domestic worker typically enters and leaves the house based on the owner's schedule. If they deviate from this pattern, the anomaly score would rise.

Anomaly scores indicating this behaviour could be a clue to investigate if the worker is engaging in unauthorised activities outside their work hours.

In all these cases, the anomaly scores provide a quantitative measure of how much an observed activity deviates from the established norm. High anomaly scores can serve as triggers for further investigation and prioritisation of cases. Investigators can use these scores to identify potentially suspicious activities, corroborate claims, and gather evidence to prove or disprove allegations.

Cautiously, the effectiveness of anomaly detection heavily depends on the quality of the training data and the chosen machine learning algorithm. It's therefore crucial to fine-tune the models to minimise false positives and negatives and to interpret the anomaly scores within the context of the specific smart home environment and its residents' behaviour patterns.

6.6.2 Enhancing Forensic Investigations through Anomaly Detection in IoT Environments

In the contemporary landscape of forensic investigations, the integration of anomaly detection within IoT environments has emerged as a game-changing approach. By harnessing the power of anomaly detection, investigators can uncover unusual behaviour patterns that warrant scrutiny, significantly enhancing their capacity to solve complex real-life cases. This multifaceted strategy capitalises on the potential of IoT technologies to provide insights and leads, while also requiring meticulous attention to legal considerations and acknowledging the inherent limitations of anomaly detection. This section examines the applications of anomaly detection across various scenarios, navigates the legal complexities associated with this approach, and examines the nuanced limitations that necessitate a balanced approach. Through this exploration, the profound implications of integrating anomaly detection into forensic investigations come to light.

Unusual Behaviour Detection in Smart Homes:

In scenarios involving smart homes, anomaly detection serves as a valuable tool for identifying unusual activities that might warrant forensic investigation. For instance, if a homeowner reports a theft, anomaly detection can be applied to analyse access patterns to specific areas, such as the bedroom, during the reported time of the incident. By assigning higher anomaly scores to activities that deviate from the established norms, investigators can focus their efforts on individuals with elevated scores, leading to targeted inquiries and evidence collection.

Additionally, anomaly detection can be utilised to address suspicions of misuse of privileges within the smart home environment. By monitoring energy consumption patterns and light usage, the system can identify deviations from the expected behaviour. This information helps investigators determine if unauthorised activities, such as excessive energy usage or lighting, have taken place and evaluate the impact of such behaviour.

Suspicious Activity in Cybercrime Investigations:

In the realm of cybercrime investigations, anomaly detection proves its worth by detecting unauthorised activities and potential security breaches. For instance, in a scenario where a hacker gains control over a smart home system, anomaly detection can track unusual access and control patterns. Activities such as changes in thermostat settings during the night can be flagged as suspicious, alerting investigators to a possible security breach. This approach allows for the early identification of ongoing cyberattacks and provides insights into the hacker's tactics and objectives.

Employee Misconduct Detection:

Anomaly detection can play a crucial role in addressing suspicions of employee misconduct within a workplace. For example, if an employee is suspected of using office resources for personal tasks, anomaly detection can be employed to monitor digital activities. Deviations from the expected behaviour, such as accessing personal websites during non-working hours, can result in elevated anomaly scores. These scores enable investigators to pinpoint employees engaging in unauthorised behaviour, potentially leading to policy adjustments or disciplinary actions.

Network Intrusion Detection:

In the context of network intrusion and insider threat detection, anomaly detection provides a means to identify potentially malicious activities. By observing data access patterns, the system can detect abrupt changes in behaviour related to sensitive data. When an employee suddenly accesses significant amounts of sensitive information outside their established pattern, the system assigns high anomaly scores. This aids in the detection of insider threats or potential data breaches by analysing deviations from the norm.

Detecting Insider Threats:

For organisations aiming to detect insider threats, anomaly detection proves invaluable. In scenarios involving data theft, anomaly detection can identify employees who are accessing large quantities of sensitive data outside their typical patterns. Such behaviours result in high anomaly scores, indicating potential data theft. This approach allows investigators to prevent insider threats by monitoring and addressing abnormal access behaviour.

6.6.3 Legal Considerations and Limitations:

Legal Considerations

When applying anomaly detection in forensic investigations and cybercrime cases, certain legal considerations must be considered. Privacy laws must be upheld, ensuring compliance with regulations when collecting and analysing data from smart devices or digital activities. Unauthorised access to personal information or network data can lead to legal repercussions. Furthermore, digital rights should be respected, and proper authorisation must be obtained before monitoring any activities involving individuals' devices or networks. In cases where data collection involves individuals' privacy, obtaining informed consent is a legal imperative. See Section 2.6 where this research discussed the legal issues in IoT. The proposed Theoretical IoT Forensic in Figure 4.1 clearly provides the steps that need to be followed in the IoT forensic investigation process which includes data collection.

Limitations

Despite its effectiveness, anomaly detection has certain limitations that investigators should be aware of. False positives and false negatives can occur due to factors such as insufficient historical data or evolving patterns. False positives might lead to unnecessary investigations, while false negatives could result in overlooking crucial evidence. Moreover, context plays a significant role. Anomaly detection might not account for the full context of an activity, potentially leading to misinterpretations. Privacy concerns should also be acknowledged, as monitoring activities without proper authorisation can infringe upon individuals' privacy rights, raising ethical and legal concerns. Lastly, the quality of training data is pivotal; inaccurate or incomplete data can lead to misleading results.

Incorporating anomaly detection into smart home environments and digital activity monitoring significantly enhances the investigative capabilities in real-life forensic cases and cybercrime investigations. By identifying deviations from established norms, investigators can uncover suspicious behaviour, pinpoint patterns, and prioritise inquiries (forensic triage). However, ensuring legal compliance with privacy laws and digital rights is paramount throughout the investigative process. Recognising the limitations and potential for false results, anomaly detection should be used as a complementary tool alongside traditional investigative methods, leveraging its strengths to enhance the effectiveness of forensic investigations.

6.7 Conclusion

In conclusion, Chapter 6 demonstrates the significance of applying HI-SDR in the realm of anomaly detection within IoT environments. It highlights the crucial role of SDRs in encoding data for HTM systems and outlines the properties that make SDRs suitable for anomaly detection. The chapter presents the HI-SDR encoder as a powerful tool that not only maintains determinism and fixed sparsity but also introduces a mechanism for capturing semantics through hash indexing.

Furthermore, the proposed approach of leveraging HI-SDR in enhancing dataset representations for machine learning models is discussed. The chapter emphasises the benefits of employing SDRs in noise-resilient and robust anomaly detection, particularly in the context of smart homes where real-world data can be imperfect due to various factors. The incorporation of hash indexing is highlighted as a means to structure and expedite data processing, enabling rapid decision-making and timely anomaly detection.

The chapter concludes by addressing the application of anomaly detection in various forensic scenarios, ranging from identifying suspicious activities in smart homes to addressing cybercrime and employee misconduct. Legal considerations surrounding privacy and data collection are emphasised, and the potential limitations and nuances of anomaly detection are discussed. The proposed approach is positioned as a complementary tool to traditional investigative methods, emphasising the importance of combining technological insights with human judgment for accurate and meaningful forensic investigations. Overall, this chapter provides a comprehensive understanding of the role and application of HI-SDR in anomaly detection within IoT environments, laying the foundation for the subsequent discussions in the thesis.

CHAPTER 7. TEST AND EVALUATION

In this chapter, the test and evaluation of the performance of the proposed model is carried out to gain deeper insights into its ability to detect anomalies in the smart home forensic dataset generated. Additionally, the chapter compares the performance of the state-of-the-art machine learning algorithms selected in Section 3.13 without SDR encoding and with SDR encoding to determine the best-performing approach for the predefined model scenario.

The chapter begins by providing a comprehensive explanation of how the experiments are designed, including discussions on contextual information, different experiment parameters, and evaluation metrics. This serves as the foundation for the subsequent phase, where extensive experiments are carried out through parameter optimisation to select the parameter settings that yield the best performance for the proposed model.

These optimised parameter settings were used to compare the performance of the proposed model with other anomaly detection algorithms and techniques. The evaluation is carried out using the state-of-the-art machine learning algorithms (OCSVM and Isolation Forest), where performance metrics which included accuracy, recall, precision, and F1-measure are employed for performance evaluation.

7.1 Experimental Setup

Experimental studies were conducted on the selected state of the art anomaly detection techniques and algorithms to test and compare their performance with and without SDR. The overall experimental design is depicted in Figure 7.1.

The experiments start by preparing the dataset which is then fed to the selected state-of-the art Machine Learning models. The models produce an anomaly score that is recorded. The prepared dataset is then encoded using the HI-SDR encoder and the encoded data (SDR data) is then fed to the models to produce an anomaly score.

Embedded within the testing methodology is an empirical experimentation aimed at refining (optimising) the parameter settings of the HI-SDR in the proposed approach. This entails carrying out numerous performance iterations to comprehensively assess the behaviour of the proposed approach using the resulting anomaly scores. Through this empirical experimentation, the parameter settings that yield the best anomaly scores are identified. These settings are subsequently employed to conduct an additional evaluation of the proposed approach, comparing it against the state-of-the-art models without SDR.

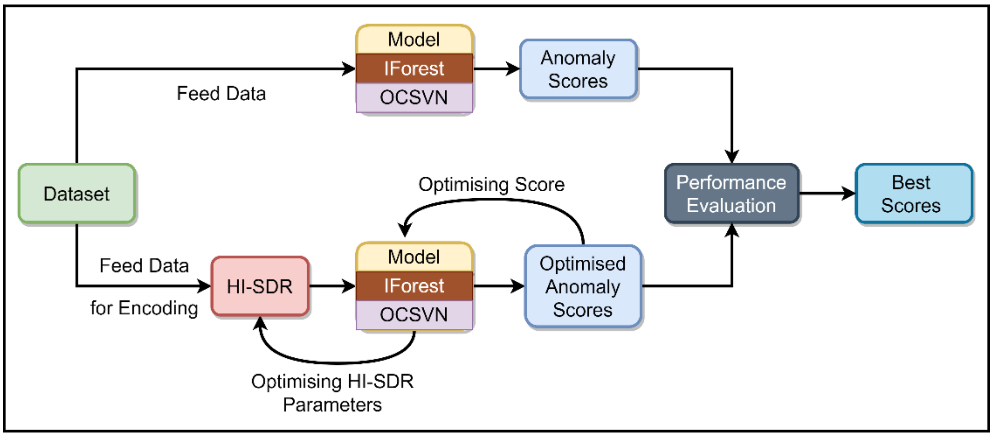


Figure 7.1 The Experimental Setup

It is important to note that, in each experiment, a consistent set of fixed parameters was applied to a given model for execution on the dataset. The model is initialised and reset as though encountering the currently employed dataset for the very first time.

Every examined model was supplied with the complete dataset generated (see Section 5.12.1) by all sensors, gathered at a consistent sampling rate (once every second). The decision to provide the model with the entirety of the data stemmed from the belief that the model should have the capacity to grasp the comprehensive pattern of the smart home resident's behaviours. These patterns exhibit a natural evolution, entailing an inherent time-related aspect to the activities. Opting for individual models for each sensor would risk obfuscating this temporal dependency and the sequential nature of these activities.

The range of anomaly scores generated by the machine learning models spans from 0.0 to 1.0, correlating to a percentage range of 0.0% to 100%. These scores signify the degree of anomaly exhibited by a specific record in relation to the model.

7.2 Experiment Parameters

Within this section, the parameters employed in the experiments defined in Section 6.5.1 are utilised to determine the optimal settings for anomaly detection. Conversely, the parameters used for the evaluation of performance are outlined in Section 7.3.

These parameters are listed in Table 7.1.

SDR Parameters	
hash	Hash digest generated
n	Total number of bits in the SDR
w	The desired number of active bits in the SDR
p	The number of partitions to be used in the construction process
Performance Metrics Parameters	
Accuracy	Overall correctness
F1 Measure	Balance of precision and recall
Precision	True positive proportion
Recall	True positive coverage.

Table 7.1 Experiment Parameters

The implementation codes for all the models used in the experiments are provided in the appendix as:

a) Appendix A: Isolation Forest Implementation Code
b) Appendix B: OCSVM Implementation Code
c) Appendix C: Isolation Forest with HI-SDR Implementation Code
d) Appendix D: OCSVM with HI-SDR Implementation Code

7.3 Performance Metrics

This thesis employs objective metrics to evaluate the performance of the proposed anomaly detection algorithm. The performance is carried out by comparing the anomaly score outputs of the algorithm under two conditions: one without the implementation of the HI-SDR encoder, and the other with the HI-SDR encoder incorporated. This comparative analysis enables a comprehensive evaluation of the algorithm's performance and sheds light on the added value brought by the HI-SDR encoder in enhancing anomaly detection accuracy. Consideration for errors is based on false alarms and failures to detect anomalies. A false alarm arises when an anomaly is erroneously detected in a location where none exists.

		True Class	
		Anomaly	Normal
Output Class	Anomaly	**TP**	**FP**
	Normal	**FN**	**TN**

Table 7.2 Contingency Table

Given the contingency table where True Positives (*TP)*, number of correctly detected anomaly activities, False Positives (*FP*), number of activities falsely detected as anomaly, and False Negatives (*FN*), number of activities falsely detected as normal, as shown in Table 7.2. The Precision and Recall may be estimated as in Equation 7.1 and Equation 7.2:

$$Recall = \frac{TP}{TP + FP} \qquad \text{Equation 7.1}$$

Recall serves as an indicator for evaluating an algorithm's inclination towards under estimation. It is the percentage of observations that are actually positive that were predicted to be positive. A greater recall value signifies a reduced likelihood of encountering under-detection issues.

$$Precision = \frac{TP}{TP + FN} \qquad \text{Equation 7.2}$$

Precision offers insights into an algorithm's predisposition towards over estimation. A higher precision value indicates a reduced probability of encountering over estimation. This is because higher precision signifies that the algorithm is being more cautious in labelling instances as positive, thereby minimising the risk of including false positives in the results. In essence, precision serves as a valuable metric to evaluate the balance between true positives and false positives, helping to ensure the algorithm's accuracy in avoiding over-segmentation scenarios.

The Harmonic Mean (HM), often referred to as the F-measure (F1 measure of F Score), serves as a singular metric that seamlessly integrates Precision and Recall, assigning them equal weights. It comes into play as a valuable tool, yielding a substantial value only when both Precision and Recall are concurrently elevated (high). This measure encapsulates the essence of striking a balance between accurate positive predictions and comprehensive capture of actual positives. In situations where an optimal equilibrium between Precision and Recall is sought after, the F-measure becomes a critical assessment criterion. The F1 Measure is estimated as in Equation 7.3:

$$F1\,Measure = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \qquad \text{Equation 7.3}$$

Within the realm of evaluating machine learning models, the accuracy metric emerges as a cornerstone for assessing a model's prowess in classification tasks. This metric serves as a comprehensive gauge of the model's ability to accurately predict outcomes across various classes. Its popularity is attributed to its straightforwardness and inherent comprehensibility, rendering it a ubiquitous yardstick for model assessment. However, beneath its apparent simplicity lies a necessity to grasp its intricacies and acknowledge its constraints (its reliability diminishes when classes are imbalanced). This ensures a more informed and balanced evaluation of a model's true performance. As with any metric, accuracy should be used in conjunction with other performance measures. Accuracy can be estimated as in Equation 7.4:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad \text{Equation 7.4}$$

To assess the effectiveness of a ML model, the anomaly scalar score, which falls within the range of 0.0 to 1.0 (0% to 100%) is generated by the specific ML algorithm under examination. This categorisation enables a direct comparison with the input value. As elaborated earlier in this section (Performance Metrics), these performance metrics play a crucial role in evaluating and examining the capabilities of the ML algorithms.

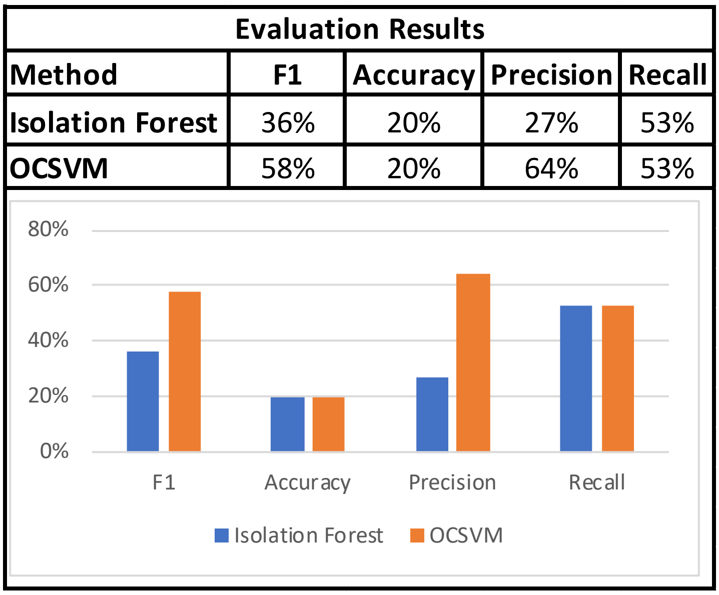
In accordance with the findings of Nehmer *et al.* (2006) and the work of (Haque, Rahman and Aziz, 2015), anomaly detection models employed within smart home systems ought to exhibit a high recall rate, ensuring the detection of nearly all anomalies, coupled with a high precision rate to effectively minimise false positives in anomaly detection.

7.4 Results

This section describes the experimental results obtained from the two state-of-the art ML algorithm followed by a comparison of their performance against the proposed HI-SDR solution after the parameters were optimised. The performance is tested using the objective metrics discussed in Section 7.3.

7.4.1 State-of-Art Algorithms Results

Here, the experiments were conducted to establish the performance of Isolation Forest and OCSVM algorithms without SDR. These results are depicted in Figure 7.2.



Evaluation Results				
Method	F1	Accuracy	Precision	Recall
Isolation Forest	36%	20%	27%	53%
OCSVM	58%	20%	64%	53%

Figure 7.2 Results of Isolation Forest and OCSVM without SDR

7.4.2 Results of the Proposed HI-SDR Solution

The HI-SDR parameters and algorithm are explained in Section 7.2. The first step in utilising the HI-SDR encoder was to optimise its parameters. This section, therefore, conducts experiments to determine the optimised parameters by combining the HI-SDR parameters for higher performance.

7.4.2.1 HI-SDR Parameter Optimisation

Given the wide array of applications in anomaly detection and the continuous advancement of algorithms designed for detecting anomalies, there is a growing need for an automated assessment approach to gauge the performance of various algorithms. This approach is commonly known as objective evaluation, and its purpose is to compare the outcomes produced by distinct algorithms. This comparison involves contrasting each algorithm's output with the established ground truth and quantifying the disparities using objective metrics. The main hurdle in this context is the existence of one or more decision process parameters (referred to as thresholds) that significantly impact the results yielded by the algorithms. Consequently, these evaluation methods should serve the dual purpose of aiding in the selection of optimal parameters for each algorithm while also ranking different algorithms based on the specific requirements of the given application.

The HI-SDR parameters are discussed in Section 7.2. The author Alshammari (2018a) established that the optimised parameters for the HI-SDR encoder as conducted in their experiments were ***n*** = 600; ***w*** = 3; and ***p*** = 8. This research adopted this approach and conducted experiments by using a mix (combination) of ***n***, ***w***, and ***p*** parameters to establish the best combination for better performance. The dataset being a binary dataset of 1s and 0s, the ***w*** parameter which denotes the active bits in an SDR array was used as the base of the other parameters. Therefore, the first combination done was to compare the performance of ***w***=1, with respective increments of both ***n*** and ***p***. The results are shown in Figure 7.3.

		No SDR	n=100	n= 200	n = 300	n = 300 p=2	n = 600
IForest	**F1**	36%	65%	65%	60%	46%	60%
	Accuracy	20%	37%	37%	37%	27%	37%
	Precision	27%	49%	49%	43%	33%	43%
	Recall	53%	99%	99%	99%	73%	99%

		No SDR	n=100	n= 200	n = 300	n = 300 p=2	n = 600
OCSVM	**F1**	58%	55%	0%	55%	0%	55%
	Accuracy	20%	37%	0%	37%	0%	37%
	Precision	64%	38%	0%	38%	0%	38%
	Recall	53%	100%	0%	100%	0%	100%

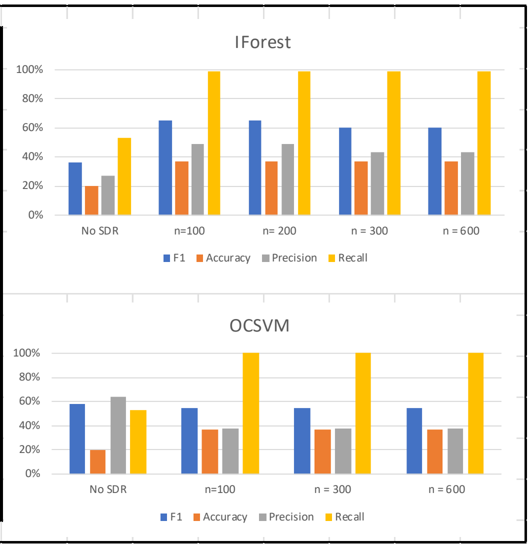


Figure 7.3 Results of w=1 with n and p Combinations

The results in Figure 7.3 show that there is no significant improvement of performance with these parameter combinations. For the IForest, an increase in the size of the SDR (***n***) decreases the performance for F1 Measure and Precision by 5% and 6% respectively. The performance is even much lower when ***p*** is increased. For OCSVM, the performance is low when ***p*** is increased as the results are null.

The best performance for ***w=1*** is seen when ***n=100*** and ***p=1***.

The next combination was to use ***w***=2. The results were:

		No SDR	n=150	n = 300	n=300 p=2	n = 600	n = 600 p=2
IForest	F1	36%	65%	60%	46%	21%	46%
	Accuracy	20%	37%	37%	27%	7%	27%
	Precision	27%	49%	43%	38%	23%	33%
	Recall	53%	99%	99%	73%	19%	73%
		No SDR	n=150	n = 300	n=300 p=2	n = 600	n = 600 p=2
OCSVM	F1	58%	0%	0%	0%	55%	0%
	Accuracy	20%	0%	0%	0%	37%	0%
	Precision	64%	0%	0%	0%	38%	0%
	Recall	53%	0%	0%	0%	100%	0%

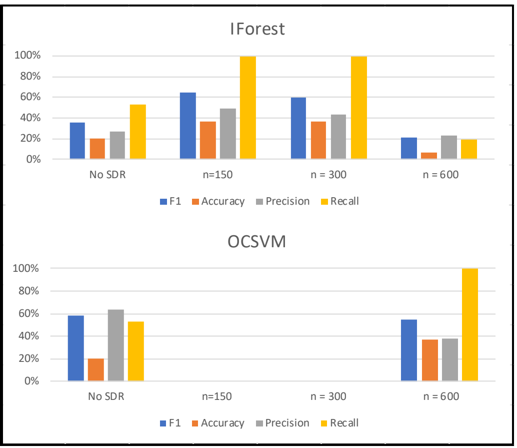


Figure 7.4 Results of w=2 with n and p Combinations

The results in Figure 7.4 again do not show any significant improvement of performance. The results also show that the parameter ***p*** when increased, reduces the performance with an increase for ***w***. When the length of the SDR ***n*** is increased, both the IForest and OCSVM performance is low. For example, when ***w=2***, ***n=300***, and ***p=1***, the F1 Measure and Precision reduce by 5% and 6% respectively for IForest. The OCSVM model does not return any results with these combinations. Although there are some good performances of OCSVM when ***n=600***, all the performance metrics for IForest are severely affected by the increase in the size of ***n***.

The combination of the parameter ***w=3*** had the results shown in the following Figure 7.5.

The results when ***w=3*** as seen from Figure 7.5. There is a significant improvement of performance results when the size of the SDR (***n***) is increased to 300. This improvement is exhibited in both the IForest and OCSVM. On the contrary, the increase of p does not yield any improved results and neither does the increase of the size of the SDR (n). For example, using the optimised parameters from the author of the HI-SDR - Alshammari (2018a), by an increase of ***n*** to 600 and ***p*** to 8, performs poorly compared to when ***n=300*** and ***p=1***. This is illustrated in Figure 7.6

		No SDR	n=100	n = 300	n = 300 p = 3	n=450	n=600	n = 600 p = 3	n = 600 p = 8
IForest	F1	36%	60%	65%	46%	60%	60%	46%	45%
	Accuracy	20%	37%	37%	27%	37%	37%	27%	27%
	Precision	27%	43%	49%	33%	43%	43%	33%	32%
	Recall	53%	99%	99%	73%	99%	99%	73%	73%
		No SDR	n=100	n = 300	n = 300 p = 3	n=450	n=600	n = 600 p = 3	n = 600 p = 8
OCSVM	F1	58%	18%	55%	0%	0%	0%	18%	18%
	Accuracy	20%	7%	37%	0%	0%	0%	7%	7%
	Precision	64%	17%	38%	0%	0%	0%	17%	16%
	Recall	53%	19%	100%	0%	0%	0%	19%	19%

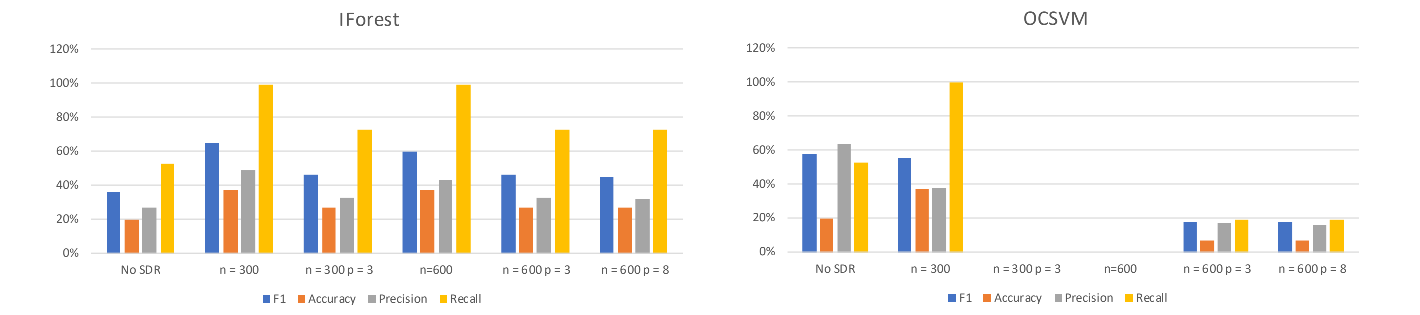


Figure 7.5 Results of w=3 with n and p Combinations

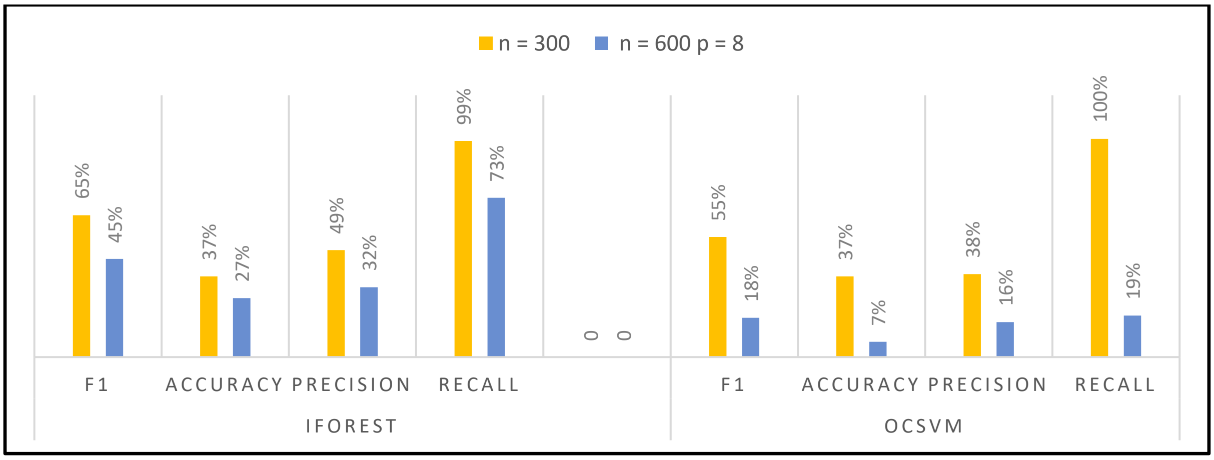


Figure 7.6 Comparative Results for Increase of n and p Parameters

These comparative results in Figure 7.6 confirm that the increase of ***n*** to more than 300 and ***p*** to 8, decrease the performance of both IForest and OCSVM in all the four performance evaluation metrics.

More experiments were conducted to evaluate the effectiveness of increasing the value of w and p as seen in Figure 7.7 and Figure 7.8. In both cases, the performance of OCSVM is not desirable because there are null results indicating that the combination does not achieve the expected results for good model performance.

		No SDR	n = 300	n=300 p=2	n = 600
IForest	F1	36%	65%	46%	65%
	Accuracy	20%	37%	27%	37%
	Precision	27%	49%	33%	49%
	Recall	53%	99%	73%	99%
OCSVM		No SDR	n = 300	n=300 p=2	n = 600
	F1	58%	0%	55%	0%
	Accuracy	20%	0%	27%	0%
	Precision	64%	0%	38%	0%
	Recall	53%	0%	73%	0%

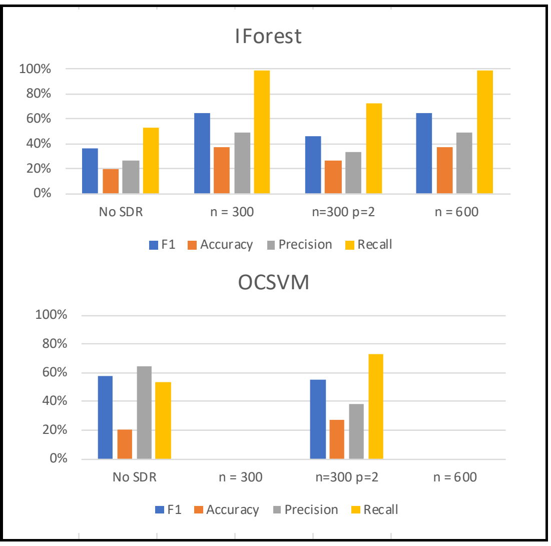


Figure 7.7 Results of w=4 with n and p Combinations

		No SDR	n=100 p=2	n = 300	n = 600
IForest	F1	36%	46%	65%	65%
	Accuracy	20%	27%	37%	37%
	Precision	27%	33%	49%	49%
	Recall	53%	73%	99%	99%
		No SDR	n=100 p=2	n = 300	n = 600
OCSVM	F1	58%	0%	0%	0%
	Accuracy	20%	0%	0%	0%
	Precision	64%	0%	0%	0%
	Recall	53%	0%	0%	0%

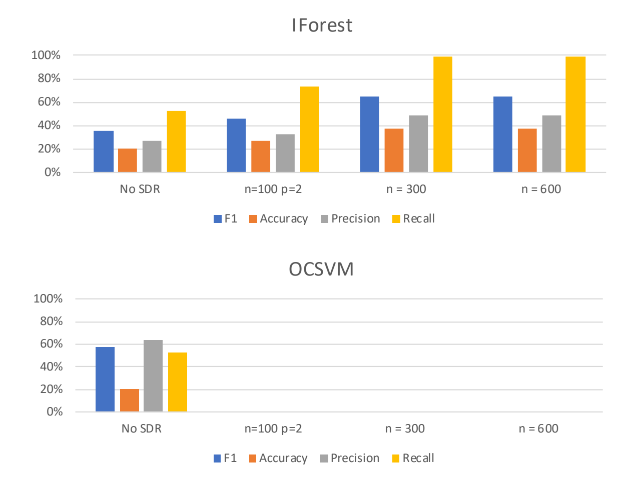


Figure 7.8 Results of w=5 with n and p Combinations

From these experiments, the optimised parameters are achieved when ***w=3***, ***n=300***, and ***p=1***. These optimised parameters produced the following results in Table 7.3.

	IForest with SDR	OCSVM with SDR
F1	65%	55%
Accuracy	37%	37%
Precision	49%	38%
Recall	99%	100%

Table 7.3 Optimised Results with SDR

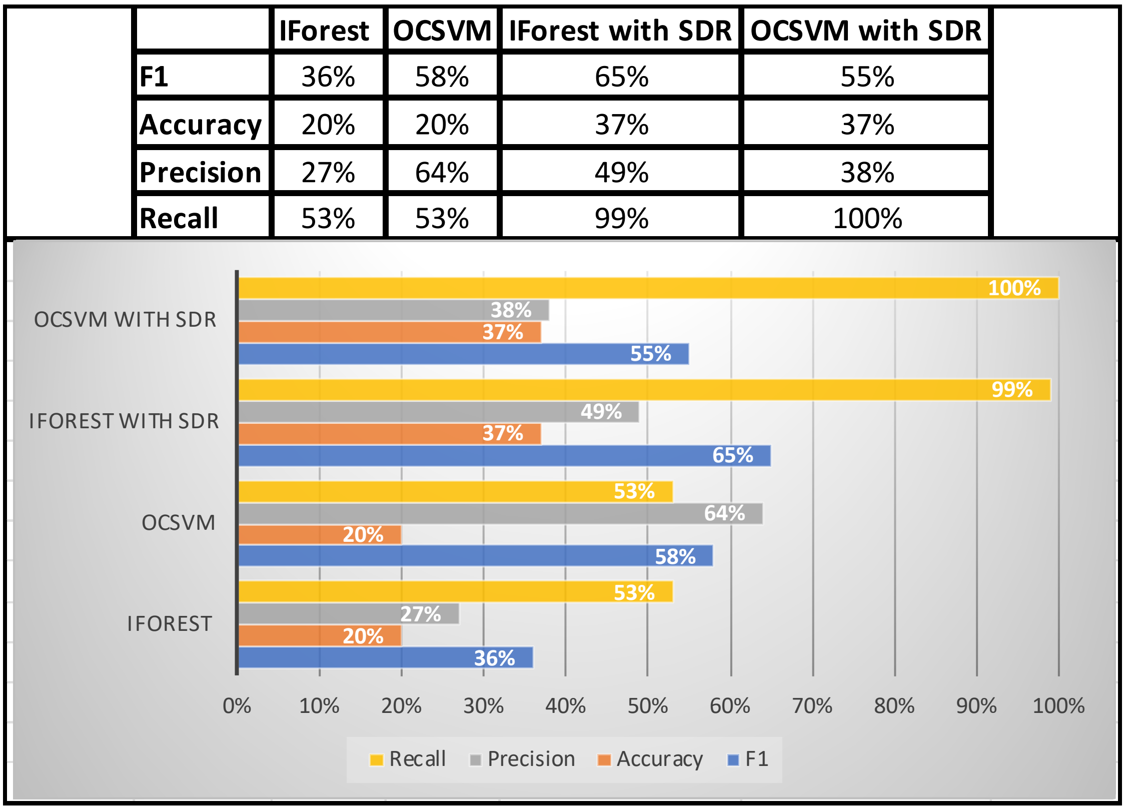
7.5 Results Analysis and Discussion

This section of the thesis discusses the results by analysing how the state-of-the-art algorithms performed without the SDR against with SDR for anomaly detection. The performance evaluation was carried out using the traditional learning method with the objective performance metrics F1 Measure, Accuracy, Precision, and Recall.

Following an evaluation of anomaly detection performance using the state-of-the art algorithms and techniques, this research has summarised the optimal scores achieved by each algorithm and technique through the experiment outlined in Section 7.1. These results are presented in Figure 7.9. The results garnered from this evaluation distinctly underscore the remarkable advantages of the proposed HI-SDR solution, showcasing its superior performance in comparison to the established state-of-the-art algorithms. This not only emphasises the substantial efficacy of HI-SDR but also elucidates the incremental advantage gained by introducing this innovative approach to anomaly detection.

The incorporation of an SDR encoder into state-of-the-art algorithms presents a compelling approach that leads to notable enhancements in algorithmic performance. Notably, when applying the SDR encoding to both IForest and OCSVM, there emerges a remarkable improvement of 17% in the Accuracy score and a significant leap of over 45% in the Recall score. These findings exemplify the robust impact of SDR on enhancing the algorithms' ability to accurately identify anomalies.

Furthermore, the utilisation of SDR encoding results in a substantial enhancement for IForest's Precision score. Notably, the Precision score, which starts at 27% without SDR encoding, experiences a remarkable boost to 49% when integrated with SDR.



	IForest	OCSVM	IForest with SDR	OCSVM with SDR
F1	36%	58%	65%	55%
Accuracy	20%	20%	37%	37%
Precision	27%	64%	49%	38%
Recall	53%	53%	99%	100%

Figure 7.9 Evaluation Results with and without SDR

This notable uplift in precision reinforces the value of SDR encoding in refining the algorithms' ability to precisely detect anomalies.

Delving deeper into the evaluation metrics, the F1 Measure score for IForest also demonstrates compelling growth with the integration of SDR encoding. Specifically, the F1 Measure score witnesses a remarkable 29% improvement, advancing from an initial score of 36% to an impressive 65%. This upswing in F1 Measure underscores the substantial benefit gained by leveraging SDR encoding to optimise the overall anomaly detection performance of the algorithm.

However, it's worth highlighting that an intriguing observation emerged: OCSVM demonstrated slightly better performance without the incorporation of SDR, particularly in relation to F1 Measure and Precision metrics. This nuanced finding adds depth to the understanding of the interplay between SDR and algorithmic behaviour, showcasing that while the general trend favours SDR integration, there are specific instances, such as with OCSVM, where its absence results in more favourable outcomes for certain evaluation criteria.

These results underscore the enhanced capabilities of the state-of-the-art algorithms when integrated with SDR encoding. Through this integration, the acquired patterns from both the original simulated data and the semantically enriched data representation are synergistically harnessed in innovative combinations. This strategy not only facilitates improved performance but also underscores the power of leveraging learned patterns across different dimensions of data processing. The integration of SDR encoding yields multifaceted improvements across various evaluation metrics, reaffirming its capacity to significantly elevate the proficiency of state-of-the-art algorithms in anomaly detection scenarios.

To leverage the use of Anomaly Detection for forensic investigations, it should be noted that the used algorithm should correctly identify true anomalies and avoid false positives.

Reducing False Alarm:

The integration of SDR encoding generally improves the precision and accuracy of anomaly detection algorithms, making them more discerning in labelling anomalies. This can help to reduce false alarms and minimises false alarms while still effectively detecting genuine anomalies.

Enhancing Anomaly Detection:

The integration of SDR encoding also leads to a significant improvement in the recall of anomaly detection algorithms. This means that the algorithms are less likely to fail to detect anomalies. The improved recall indicates that more true anomalies are successfully identified. However, it is important to balance recall with precision. While higher recall is desirable, it should not come at the expense of significantly increased false alarms.

7.6 Summary

In conclusion, this chapter focused on the testing and evaluation of anomaly detection techniques and algorithms, with a particular emphasis on the integration of SDR using the HI-SDR encoder. The experimental setup involved rigorous experimentation on the state-of-the-art machine learning algorithms, specifically Isolation Forest and OCSVM, both with and without the incorporation of SDR. The experimental design encompassed various parameters such as SDR size (***n***), number of active bits (***w***), and number of partitions (***p***). — to unveil the intricate interplay within the anomaly detection landscape.

The primary objective of the experiments was to assess the impact of SDR on anomaly detection performance, measured through objective metrics including Accuracy, F1 Measure, Precision, and Recall. The results showed that the integration of SDR encoding consistently improved the overall performance of the algorithms across multiple metrics. This enhancement was particularly pronounced in terms of Accuracy, Recall, Precision, and the F1 Measure. The experimental findings indicated that SDR encoding contributed to more accurate and comprehensive anomaly detection, highlighting the benefits of leveraging SDR in refining algorithmic performance.

However, it is noteworthy that while the overarching trend exhibited the positive impact of SDR, there were nuanced instances where specific algorithms, particularly OCSVM, showcased slightly better performance in the absence of SDR integration for certain evaluation criteria. Nevertheless, the overall trend emphasised the effectiveness of SDR encoding in enhancing anomaly detection accuracy, which is crucial for applications in various fields, including forensic investigations.

For instance, the integration of SDR encoding led to an impressive improvement of 17% in Accuracy and an astonishing leap of over 45% in Recall. Additionally, in the case of IForest, the Precision score witnessed a remarkable boost from 27% to 49%, an uplift of 22%. Moreover, the F1 Measure, a pivotal metric capturing the equilibrium between Precision and Recall, experienced a substantial 29% improvement, ascending from an initial score of 36% to an impressive 65%. These percentages underscore the palpable enhancements attributed to the strategic combination of SDR encoding with state-of-the-art algorithms.

In summary, the experimental results underscored the significant advantages of incorporating SDR encoding into state-of-the-art anomaly detection algorithms. This innovative approach not only demonstrated the potential for enhanced performance but also showcased the broader capability of leveraging learned patterns in multiple dimensions of data processing. The chapter's findings serve to support the feasibility and utility of utilising SDR-based techniques in enhancing the accuracy and effectiveness of anomaly detection algorithms, with potential implications for various real-world applications.

CHAPTER 8. SUMMARY, CONCLUSION, AND FUTURE WORK

This thesis has presented a novel IoT forensics framework that aids IoT forensic process. To evaluate the practicality of the framework, a smart home environment was simulated, dataset generated, and this research proposed an approach using HI-SDR which leveraged Machine Learning approaches to detect anomalies. Additionally, this thesis has meticulously analysed the status of IoT forensic domain by conducting a state-of-the-art systematic literature review to highlight future directions.

8.1 Summary

The rapid proliferation of IoT technologies has ushered in a transformative era, reshaping how we interact with our surroundings and how interconnected devices communicate. This remarkable integration of IoT into diverse aspects of contemporary life has not only brought about numerous advantages but has also presented a set of distinct challenges and opportunities within the realm of digital forensics. As IoT devices continue to grow exponentially in number and variety, a significant challenge arises in handling the sheer volume and diversity of data they generate. These devices produce copious amounts of data in varying formats and from disparate sources. This diversity can complicate the process of collecting, storing, and analysing relevant data during digital forensic investigations. Developing effective methodologies and frameworks for managing this influx of data becomes paramount.

One of the complexities of IoT is the diverse array of devices, operating systems, and communication protocols that it encompasses. This diversity creates a landscape where each device may possess unique hardware and software characteristics, necessitating tailor-made investigation techniques. Consequently, digital forensics professionals must adapt to this heterogeneity and devise strategies to extract pertinent information.

A pressing issue that has emerged alongside these challenges is the lack of standardised regulations within IoT forensics. The vast and diverse IoT landscape, coupled with the unique attributes of each device, makes it arduous to establish a one-size-fits-all forensic guideline. This absence of standardisation impedes the consistency and reliability of investigations. Without a unified framework, investigators often find themselves navigating a labyrinth of diverse protocols, leading to inefficiencies and potential oversights.

An intriguing aspect of IoT is its real-time functionality, with many devices operating within dynamic environments where data is generated and transmitted continuously. This real-time nature presents a departure from traditional forensic methodologies, requiring investigators to develop and adopt real-time forensic approaches that align with these evolving scenarios.

However, the ubiquity of IoT also raises concerns about privacy and data security. IoT devices frequently collect sensitive personal and environmental data, sparking a delicate balance between the imperatives of digital forensics and safeguarding user privacy. Adhering to privacy regulations while still extracting vital information adds another layer of complexity to IoT digital forensics. Moreover, the susceptibility of IoT devices to cybersecurity threats poses an additional challenge. Due to factors for example, limited processing power, inadequate security measures, and infrequent updates, these devices are often vulnerable. This vulnerability can significantly impact digital forensic investigations, as compromised devices might not preserve data as expected.

Despite these challenges, there are substantial opportunities within IoT digital forensics. The rich contextual data generated by IoT devices, such as timestamps and geolocation, can enhance the accuracy and depth of investigations, providing crucial insights into events. The interconnectivity of IoT devices creates intricate digital footprints that offer insights into user behaviours, actions, and interactions, enabling investigators to reconstruct events with greater clarity.

Collaboration is emerging as a pivotal aspect of IoT digital forensics. The intricate nature of IoT ecosystems necessitates cooperation between disciplines like cybersecurity, data science, and legal expertise. This interdisciplinary approach can yield more comprehensive and effective forensic investigations.

This thesis was motivated by the following research questions, with the intention of examining the hypotheses posited in the research.

1. ***Research Question*:** What is the current state of IoT digital forensics methodologies, models, and frameworks, and how can they be improved to address the legal and technical challenges in the field?

Hypothesis: Given the rapid evolution of IoT technologies and the increasing complexity of digital ecosystems, it is hypothesised that the current state of IoT digital forensics methodologies, models, and frameworks is characterised by a fragmented landscape with varied approaches, lacking standardised procedures and comprehensive frameworks. Additionally, it is expected that existing methodologies may struggle to effectively address the intersection of legal and technical challenges in IoT forensics, including issues such as data privacy, jurisdictional concerns, and the diverse range of IoT devices and protocols. However, it is anticipated that advancements in interdisciplinary collaboration, the development of specialised tools and techniques, and the establishment of clearer regulatory guidelines can lead to significant improvements in IoT digital forensics methodologies, enabling more robust investigations and better alignment with legal requirements.

2. ***Research Question:*** How can standardisation of rules be achieved to mitigate legal and technical challenges in IoT digital forensics, and what role can Machine Learning play in enhancing the investigation process?

   Hypothesis: It is hypothesised that standardisation of rules in IoT digital forensics can be achieved through collaborative efforts among stakeholders, including industry professionals, policymakers, and regulatory bodies. Such standardisation efforts are expected to address legal and technical challenges by establishing uniform guidelines, protocols, and best practices for collecting, analysing, and presenting digital evidence from IoT devices. Furthermore, it is anticipated that Machine Learning algorithms can significantly enhance the investigation process by automating certain aspects of forensic analysis, such as anomaly detection, pattern recognition, and predictive modelling. Machine Learning techniques have the potential to improve the efficiency and accuracy of digital forensic investigations in IoT environments, enabling investigators to sift through large volumes of data, identify relevant evidence, and uncover insights that may otherwise be overlooked. Through the integration of standardised rules and Machine Learning driven approaches, it is believed that IoT digital forensics can become more effective, reliable, and adaptable to the evolving landscape of connected devices and technologies.

3. ***Research Question:*** What is the effectiveness of the proposed IoT forensic framework and the proposed approach of integration of the Machine Learning technique in addressing legal and technical challenges, and how does it compare to existing methodologies?

   Hypothesis: It is hypothesised that the proposed IoT forensic framework, integrated with Machine Learning techniques, will demonstrate effectiveness in addressing both legal and technical challenges compared to existing methodologies. This hypothesis assumes that the integration of Machine Learning algorithms will enhance the capability to automate the analysis of complex IoT data, leading to improved accuracy, efficiency, and scalability in digital forensic investigations. Additionally, it is expected that the proposed framework will provide better support for addressing legal challenges by incorporating standardized rules and procedures, thereby ensuring the admissibility and reliability of digital evidence in legal proceedings.

In Chapter 2, a Literature Review is carried out to answer the research questions 1 and 2. This chapter provides a comprehensive overview of IoT forensics by presenting a state-of-the-art Systematic Literature Review. It defines IoT forensics as a branch of digital forensics encompassing device, network, and cloud-level investigations. The challenges of uncertainty, chain of custody, and cross-border jurisdiction are discussed, highlighting the difficulties in handling IoT data. The chapter emphasises the lack of standardised processes and the limitations of existing theoretical models. It points out the need for practical methodologies and tools that can be scientifically validated. The significance of smart analysis, privacy protection, and recommendations for legal solutions to combat the lack of standardisation in IoT forensic are outlined, along with the potential of digital warrants. The chapter concludes by underscoring the importance of addressing IoT forensics challenges due to the growing adoption of IoT devices and cloud-based technologies and proposes significant future research directions.

Chapter 3 answers research 3 and explores the integration of automated processes, AI, and machine learning in digital forensics. It highlights the advancements brought by these technologies, including faster results and improved efficiency in handling cybercrime cases. However, it also notes that automated tools still require human oversight to ensure accuracy. The chapter emphasises the potential for AI and automation in risk mitigation and effective digital evidence handling. It concludes by encouraging forensic investigators to embrace these technologies to adapt to the evolving landscape of cybercrimes.

Chapter 4 answers research questions 1 and 3 and proposes a novel IoT forensic framework that provides a step-by-step process to aid the forensic process. The framework outlines the various phases of the investigation process, from preparation and live investigation to offline investigation and presentation of findings. The chapter emphasises the need for security measures, documentation, and flexibility while adhering to the proposed framework. It recognises the significance of reconstructing events and structuring investigations for effective communication within the forensic community. The framework serves as a structured guide for investigators navigating complex IoT investigations and promoting transparency, traceability, and accountability.

Chapter 5 discusses the significance of using simulation techniques for generating realistic datasets in IoT forensic analysis. It highlights the OpenSHS simulator as a valuable tool for creating elaborate and representative datasets for smart home environments. The chapter explains the methodology of dataset acquisition in simulated environments and provides unique hypothetical forensic case scenarios for simulations. This chapter generates IoT forensic datasets depicting real life scenarios and publishes this dataset to be accessed publicly by the research community. It stresses the importance of these datasets in advancing the field of smart home forensics and promoting collaboration among researchers. The chapter positions simulation techniques as a means of tackling the complexities of IoT forensic analysis effectively due to inadequate real smart homes for research.

Chapter 6 introduces the idea of Sparse Distributed Representation and explores the application of HI-SDR in anomaly detection within IoT environments. It discusses the properties of SDRs that make them suitable for anomaly detection and introduces the HI-SDR encoder. The chapter emphasises the benefits of using SDRs in noise-resilient and robust anomaly detection. It highlights the role of HI-SDR in enhancing dataset representations and proposes an approach to which the HI-SDR can be incorporated with ML models for better performance for anomaly detection. The chapter illuminates how anomaly scores can be attributed to the forensic scenarios to help investigators eliminate false alarms and lay emphasis on true anomalies for further investigations.

Chapter 7 illustrates and carries out the experiments to test the hypothesis of this research on whether the incorporation of HI-SDR for better representation can improve the anomaly scores compared to state-of-the-art ML models.

Using anomaly detection for forensics purposes requires that the algorithm is accurate enough to provide true anomalies while reducing the false alarms. Therefore, the performance of the proposed approach of using SDR was compared with and without SDR on the state-of-the-art models using the traditional ML performance metrics, Accuracy, F1 Measure, Precision, and Recall.

8.2 Conclusion

While the evolution of IoT forensics has led researchers to start thinking of ways to develop specialised tools and software tailored to address the unique challenges posed by IoT ecosystems, the available frameworks often lean more towards the theoretical than the practical. Theoretical frameworks provide a conceptual understanding of how IoT forensic investigations should be conducted, but they often lack the detailed guidance needed for real-world implementation. This divide between theory and practice further accentuates the need for standardised regulations and methodologies that can bridge this gap.

Another significant concern is the lack of real-life scenarios to test and refine IoT forensic research and development. Without access to diverse and realistic case studies, it becomes challenging to validate and improve existing forensic techniques, tools, and frameworks. Real-life scenarios are essential for assessing the effectiveness of digital forensic approaches in different IoT contexts and uncovering potential limitations and areas for improvement.

Furthermore, the available IoT simulated datasets often fall short in representing the complexity and diversity of real-world IoT scenarios. These datasets might not accurately capture the intricacies of IoT device interactions, data generation, and transmission within dynamic environments. Consequently, relying solely on such simulated datasets can hinder the development of robust forensic methodologies that can effectively address the challenges of real-world IoT investigations.

As the significance of IoT digital forensics gains recognition, the establishment of standardised security practices and best-in-class methodologies becomes crucial. The IoT industry is gradually acknowledging the necessity of uniformity in security and forensic procedures, paving the way for greater consistency and reliability in investigations. To address the lack of standardised regulations, the theoretical nature of existing frameworks, the dearth of real-life scenarios, and the unsuitability of available datasets, there is a pressing need to develop new datasets that accurately reflect the complexities of IoT environments. These datasets should encompass a variety of devices, communication protocols, and dynamic scenarios to enable researchers and practitioners to create, test, and refine forensic techniques that are truly applicable to the challenges of IoT investigations. Only through such advancements can IoT digital forensics effectively keep pace with the rapid evolution of IoT technologies and the diverse array of challenges they bring.

The statistical analysis of the integration of SDR encoding resulted in a noteworthy advancement: Accuracy increased by a significant 17%, and Recall demonstrated a remarkable surge of more than 45%. Furthermore, when examining the IForest algorithm, Precision scores exhibited an impressive rise from 27% to 49%, indicating a notable uplift of 22%. Notably, the F1 Measure, a critical metric that balances Precision and Recall, experienced a substantial improvement of 29%, progressing from an initial score of 36% to an impressive 65%. These percentage increments highlight the tangible improvements achieved through the strategic fusion of SDR encoding with state-of-the-art algorithms.

The behaviour patterns learned from the data stream at different hierarchical levels of the proposed approach, along with the enhanced semantic representation of the data using HI-SDR encoding, are effectively fused through innovative combinations with state-of-the-art model to produce better performance. As a result, these findings provide substantial support for the proposed hypothesis.

8.3 Contributions to Knowledge

The contributions to knowledge offered by this thesis are divided into primary and secondary contributions and are summarised as follows:

1. The primary contributions are:
    (a) A novel IoT forensic framework has been proposed, addressing legal and technical challenges in IoT forensic processes. This framework offers a structured approach for investigators to navigate the complexities of IoT environments, facilitating efficient and effective investigations.
    (b) Validation of the proposed framework by selecting acceptable Machine Learning technique for analysing IoT forensic data. The introduction of a new approach of applying HI-SDR as an input to state-of-the-art anomaly detectors represents a significant contribution. This technique enhances the accuracy of anomaly detection algorithms, contributing to improved forensic investigations in IoT environments. Performance testing of the proposed approach is comprehensively provided to show how it has been fused with the state-of-the-art models and how it improves the performance of these models.
2. The secondary contributions are:
    (a) A comprehensive SLR of IoT forensics and a review of the current legal and technical challenges of IoT forensics. The SLR identified gaps in existing IoT forensic frameworks, methodologies, and models and highlighted the need for practical and validated approaches. The review sets the stage for further research by identifying future directions for the research community.
    (b) Generation of an IoT Forensic dataset of a smart home capturing forensically simulated scenarios with annotated anomalies. A review of the literature on smart homes highlights the absence of a standardised dataset specifically designed for IoT forensics within smart environments. Utilising OpenSHS, this research emulated forensic scenarios in the daily routines of a smart home resident and provided annotations to the dataset. This dataset is now publicly accessible, enabling the research community to test and evaluate their machine learning algorithms and develop intelligent applications to aid the Digital Forensics domain.

8.4 Future Work

The culmination of this research project has revealed significant insights into the realm of IoT forensics, encompassing challenges, opportunities, and novel methodologies. Yet, as the IoT and digital forensics domains continue to evolve, numerous areas for future research and development come into focus. This section presents potential directions for further exploration and refinement within the realm of this research.

An important avenue for future work involves enhancing standardisation and guidelines in IoT digital forensics. Although this thesis has reviewed some of the legal aspects surrounding IoT forensics and proposed some guidelines, there remains much to be done. The absence of universally accepted regulations and protocols remains a challenge. Collaborative efforts between legal experts, cybersecurity professionals, digital forensics practitioners, and IoT industry stakeholders are essential to establish comprehensive standards that address the unique features of IoT devices and environments.

The gap between theoretical frameworks and practical implementation is another area warranting attention. While this thesis has introduced a novel IoT forensic framework, bridging this gap is crucial. Future research should be geared towards providing detailed implementation guidelines, methodologies, and toolkits aligned with the proposed frameworks. Practical resources would empower investigators to apply frameworks effectively in real-world situations, ensuring consistency and reliability in IoT forensic investigations.

To overcome the limitations of simulated datasets and better represent real-world IoT environments, generating authentic case studies and datasets is essential. These case studies should encompass diverse IoT devices, communication protocols, and dynamic scenarios. Researchers and practitioners can then use these real-life scenarios to validate and refine forensic methodologies, tools, and frameworks.

One of the key challenges in IoT forensics research is the absence of standardised and representative datasets for anomaly detection. Reliable datasets are essential to validate any anomaly detection methods for forensics. Yet, the very nature of anomalies makes detection a complex endeavour. In smart home contexts, anomalies are rare and inherently subjective. Each resident has distinct habits and routines, making it challenging to impartially determine if an event is anomalous. Recognising this subjectivity, this research allowed the participants to classify events as anomalies based on their personal routines. The dataset then recorded these participant-defined anomalies. For future studies, it would be beneficial to develop datasets employing the same approach but delving into intricate scenarios and representing multiple residents.

Exploring advanced machine learning approaches within IoT forensics holds potential for further advancement. Integration of deep learning techniques, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), could enhance anomaly detection accuracy. Techniques like transfer learning and ensemble methods could also lead to more robust solutions.

In light of data privacy concerns, research should focus on privacy-preserving forensic techniques. Methods for extracting relevant forensic evidence while upholding privacy and adhering to regulations deserve exploration. Techniques like differential privacy and homomorphic encryption could be integrated into IoT forensic methodologies to ensure responsible data handling.

Interdisciplinary collaboration emerges as a crucial aspect. Engaging legal experts, cybersecurity specialists, data scientists, and digital forensics practitioners in collaborative efforts can lead to integrated solutions. This approach could yield unified tools, methodologies, and frameworks addressing the multifaceted challenges of IoT investigations.

In conclusion, the potential for innovative research and practical solutions in IoT digital forensics is substantial. As IoT technologies continue reshaping our world, the digital forensics field must evolve concurrently to address challenges and opportunities. The suggestions provided offer a roadmap for researchers, practitioners, and industry stakeholders to collectively contribute to IoT forensics, upholding the security and integrity of our interconnected world. This thesis has explored the complex landscape of IoT forensics, laying the groundwork for practical solutions that can enhance digital investigations in the era of IoT. The proposed IoT forensic framework, the application of HI-SDR encoding to improve the performance for anomaly detection, and the insights gained from this research contribute to the advancement of knowledge in the field, enabling more effective and efficient approaches to address the challenges posed by IoT environments.

REFERENCES

Abdel-Fattah, F., Fayyad, S., Heyari, A. M. and Al-Zoubi, H. (2023) 'A Survey of Internet of Things (IoT) Forensics Frameworks and Challenges', in *2023 International Conference on Information Technology: Cybersecurity Challenges for Sustainable Cities, ICIT 2023 - Proceeding*. Institute of Electrical and Electronics Engineers Inc., pp. 373–377. doi: 10.1109/ICIT58056.2023.10226103.

Abraham, B. and Chuang, A. (1989) 'Outlier Detection and Time Series Modeling', *Technometrics*. JSTOR, 31(2), p. 241. doi: 10.2307/1268821.

Adam, I. Y. and Varol, C. (2020) 'Intelligence in Digital Forensics Process', *8th International Symposium on Digital Forensics and Security, ISDFS 2020*. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ISDFS49300.2020.9116442.

Adjei, O., Babu C, N. and Yakubu, O. (2018) 'A REVIEW OF DIGITAL FORENSIC CHALLENGES IN THE INTERNET OF THINGS (IOT)', *International Journal of Mechanical Engineering and Technology (IJMET)*, 9(1), pp. 915–923. Available at: http://www.iaeme.com/IJMET/index.asp915http://www.iaeme.com/IJMET/issues.asp?JType=IJMET&VType=9&IType=1http://www.iaeme.com/IJMET/issues.asp?JType=IJMET&VType=9&IType=1http://www.iaeme.com/IJMET/index.asp916 (Accessed: 16 January 2019).

Aggarwal, C. C. (2015) 'Outlier Analysis', in *Data Mining*. Springer International Publishing, pp. 237–263. doi: 10.1007/978-3-319-14142-8_8.

AGGARWAL, S. (2023) 'Research on Anomaly Detection in Time Series: Exploring United States Exports and Imports Using Long Short-Term Memory', *Journal of Research, Innovation and Technologies (JoRIT)*. RITHA Publishing, 2(16), p. 199. doi: 10.57017/jorit.v2.2(4).06.

Ahmad, S. and Hawkins, J. (2015) 'Properties of Sparse Distributed Representations and their Application to Hierarchical Temporal Memory', *arXiv*, p. arXiv:1503.07469. doi: 10.48550/ARXIV.1503.07469.

Ahmed, H., Yousef, S. and Mohammad, A. (2021) 'An Internet of Things (IoT) forensics model using third-party logs-vault', in *ACM International Conference Proceeding Series*. Association for Computing Machinery, pp. 143–146. doi: 10.1145/3460620.3460746.

Ahmed, S. F., Alam, M. S. Bin, Afrin, S., Rafa, S. J., Taher, S. B., Kabir, M., Muyeen, S. M. and Gandomi, A. H. (2024) 'Toward a Secure 5G-Enabled Internet of Things: A Survey on Requirements, Privacy, Security, Challenges, and Opportunities', *IEEE Access*. Institute of Electrical and Electronics Engineers Inc., 12, pp. 13125–13145. doi: 10.1109/ACCESS.2024.3352508.

Al-Hussaeni, K., Brits, J., Praveen, M., Yaqoob, A. and Karamitsos, I. (2023) 'A Review of Internet of Things (IoT) Forensics Frameworks and Models', in *Lecture Notes in Business Information Processing*. Springer Science and Business Media Deutschland GmbH, pp. 515–533. doi: 10.1007/978-3-031-30694-5_37.

Al-Masri, E., Bai, Y. and Li, J. (2018) 'A Fog-Based Digital Forensics Investigation Framework for IoT Systems', *2018 IEEE International Conference on Smart Cloud (SmartCloud)*. IEEE, pp. 196–201. doi: 10.1109/SmartCloud.2018.00040.

Al-Sadi, M. B., Chen, L. and Haddad, R. J. (2018) 'Internet of Things Digital Forensic Investigation Using Open Source Gears', in *Conference Proceedings - IEEE SOUTHEASTCON*. IEEE, pp. 1–5. doi: 10.1109/SECON.2018.8479042.

Alabdulsalam, S., Schaefer, K., Kechadi, T. and Le-Khac, N. A. (2018) 'Internet of things forensics – Challenges and a case study', in *IFIP Advances in Information and Communication Technology*, pp. 35–48. doi: 10.1007/978-3-319-99277-8_3.

Alemdar, H., Ertan, H., Incel, O. D. and Ersoy, C. (2013) 'ARAS human activity datasets in multiple homes with multiple residents', in *Proceedings of the 2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops, PervasiveHealth 2013*. IEEE, pp. 232–235. doi: 10.4108/icst.pervasivehealth.2013.252120.

Aleskerov, E., Freisleben, B. and Rao, B. (1997) 'CARDWATCH: A neural network based database mining system for credit card fraud detection', in *IEEE/IAFE*

Conference on Computational Intelligence for Financial Engineering, Proceedings (CIFEr). IEEE, pp. 220–226. doi: 10.1109/cifer.1997.618940.

AlFahdi, M., Clarke, N. L. and Furnell, S. M. (2014) ‘Towards an automated forensic examiner (AFE) based upon criminal profiling & artificial intelligence’, in *Proceedings of the 11th Australian Digital Forensics Conference, ADF 2013*. SRI Security Research Institute, Edith Cowan University, Perth, Western Australia, pp. 1–9. doi: 10.4225/75/57b3be61fb866.

Alfaiz, N. S. and Fati, S. M. (2022) ‘Enhanced Credit Card Fraud Detection Model Using Machine Learning’, *Electronics (Switzerland)*. Multidisciplinary Digital Publishing Institute, 11(4), p. 662. doi: 10.3390/electronics11040662.

Alshammari, N., Alshammari, T., Sedky, M., Champion, J. and Bauer, C. (2017) ‘OpenSHS: Open smart home simulator’, *Sensors (Switzerland)*. MDPI AG, 17(5). doi: 10.3390/s17051003.

Alshammari, N. O. (2018a) ‘Anomaly Detection Using Hierarchical Temporal Memory in Smart Homes’, *PQDT - UK & Ireland*, (March). Available at: https://search.proquest.com/docview/2083754133?accountid=49007%0Ahttp://www.yidu.edu.cn/educhina/educhina.do?artifact=&svalue=Anomaly+Detection+Using+Hierarchical+Temporal+Memory+in+Smart+Homes&stype=2&s=on%0Ahttp://sfx.cceu.org.cn:3410/bisu?url_ver=Z39.8 (Accessed: 27 June 2021).

Alshammari, N. O. (2018b) ‘Anomaly Detection Using Hierarchical Temporal Memory in Smart Homes Exploring the Adoption of Physical Security Controls in Smart-phones’, *PQDT - UK & Ireland*, (March). Available at: https://search.proquest.com/docview/2083754133?accountid=49007%0Ahttp://www.yidu.edu.cn/educhina/educhina.do?artifact=&svalue=Anomaly+Detection+Using+Hierarchical+Temporal+Memory+in+Smart+Homes&stype=2&s=on%0Ahttp://sfx.cceu.org.cn:3410/bisu?url_ver=Z39.8 (Accessed: 21 September 2020).

Alshammari, N. O. (2018c) ‘Anomaly Detection Using Hierarchical Temporal Memory in Smart Homes Nasser Owaid Alshammari List of Publications’, (March).

Alshammari, T., Alshammari, N., Sedky, M. and Howard, C. (2018a) ‘Evaluating Machine Learning Techniques for Activity Classification in Smart Home Environments’, *International Journal of Information and Communication Engineering*, 12(February), pp. 72–78. doi: 10.1999/1307-6892/10008539.

Alshammari, T., Alshammari, N., Sedky, M. and Howard, C. (2018b) ‘SIMADL: Simulated activities of daily living dataset’, *Data*. MDPI AG, 3(2), p. 11. doi: 10.3390/data3020011.

Amaratunga, D., Baldry, D., Sarshar, M. and Newton, R. (2002) ‘Quantitative and qualitative research in the built environment: application of “mixed” research approach’, *Work Study*. MCB UP Ltd, 51(1), pp. 17–31. doi: 10.1108/00438020210415488.

Amer, M. and Goldstein, M. (2012) ‘Nearest-Neighbor and Clustering based Anomaly Detection Algorithms for RapidMiner’, *Proceedings of the 3rd RapidMiner Community Meeting and Conferernce (RCOMM 2012)*, pp. 1–12. Available at: http://www.dfki.de (Accessed: 8 July 2021).

Amer, M., Goldstein, M. and Abdennadher, S. (2013) ‘Enhancing one-class support vector machines for unsupervised anomaly detection’, in *Proceedings of the ACM SIGKDD Workshop on Outlier Detection and Description - ODD ’13*. New York, New York, USA: ACM Press, pp. 8–15. doi: 10.1145/2500853.2500857.

Anda, F., Lillis, D., Le-Khac, N. A. and Scanlon, M. (2018) ‘Evaluating automated facial age estimation techniques for digital forensics’, *Proceedings - 2018 IEEE Symposium on Security and Privacy Workshops, SPW 2018*. Institute of Electrical and Electronics Engineers Inc., pp. 129–139. doi: 10.1109/SPW.2018.00028.

Angiulli, F. and Pizzuti, C. (2002) ‘Fast outlier detection in high dimensional spaces’, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer Verlag, pp. 15–27. doi: 10.1007/3-540-45681-3_2.

Ariani, A., Redmond, S. J., Chang, D. and Lovell, N. H. (2013) ‘Simulation of a smart home environment', in *Proc. of 2013 3rd Int. Conf. on Instrumentation, Communications, Information Technol., and Biomedical Engineering: Science and Technol. for Improvement of Health, Safety, and Environ., ICICI-BME 2013*. IEEE Computer Society, pp. 27–32. doi: 10.1109/ICICI-BME.2013.6698459.

Armac, I. and Retkowitz, D. (2007) 'Simulation of smart environments', in *2007 IEEE International Conference on Pervasive Services, ICPS*, pp. 257–266. doi: 10.1109/PERSER.2007.4283924.

Aronszajn, N. (1950) 'Theory of reproducing kernels', *Transactions of the American Mathematical Society*. American Mathematical Society (AMS), 68(3), pp. 337–404. doi: 10.1090/s0002-9947-1950-0051437-7.

Babun, L., Sikder, A. K., Acar, A. and Uluagac, A. S. (2018) *IoTDots: A Digital Forensics Framework for Smart Environments*. Available at: http://arxiv.org/abs/1809.00745 (Accessed: 5 December 2018).

Baig, Z. A. *et al.* (2017) 'Future challenges for smart cities: Cyber-security and digital forensics', *Digital Investigation*, pp. 3–13. doi: 10.1016/j.diin.2017.06.015.

Bajramovic, E., Waedt, K., Ciriello, A. and Gupta, D. (2016) 'Forensic readiness of smart buildings: Preconditions for subsequent cybersecurity tests', in *IEEE 2nd International Smart Cities Conference: Improving the Citizens Quality of Life, ISC2 2016 - Proceedings*. IEEE, pp. 1–6. doi: 10.1109/ISC2.2016.07580754.

Baker, M. J. (2004) 'Selecting a Research Methodology', *The Marketing Review*. Westburn Publishers Ltd, 1(3), pp. 373–397. doi: 10.1362/1469347002530736.

Balram, N., Hsieh, G. and McFall, C. (2019) 'Static malware analysis using machine learning algorithms on apt1 dataset with string and PE header features', *Proceedings - 6th Annual Conference on Computational Science and Computational Intelligence, CSCI 2019*. Institute of Electrical and Electronics Engineers Inc., pp. 90–95. doi: 10.1109/CSCI49370.2019.00022.

Banday, M. (2018) 'Enhancing the security of IOT in forensics', in *2017 International Conference on Computing and Communication Technologies for Smart Nation,* *IC3TSN 2017*. IEEE, pp. 193–198. doi: 10.1109/IC3TSN.2017.8284475.

Barbará, D., Wu, N. and Jajodia, S. (2001) ‘Detecting Novel Network Intrusions Using Bayes Estimators’, in, pp. 1–17. doi: 10.1137/1.9781611972719.28.

Bell, E. and Bryman, A. (2007) ‘The ethics of management research: An exploratory content analysis’, *British Journal of Management*. John Wiley & Sons, Ltd, 18(1), pp. 63–77. doi: 10.1111/j.1467-8551.2006.00487.x.

Bellman, R. E. and Dreyfus, S. E. (2015) *Applied dynamic programming*, *Applied Dynamic Programming*. Princeton University Press. doi: 10.2307/3149350.

Bentotahewa, V., Yousif, M., Hewage, C., Nawaf, L. and Williams, J. (2022) ‘Correction to: Privacy and security challenges and opportunities for IoT technologies during and beyond COVID-19’, in *Privacy, Security And Forensics in The Internet of Things (IoT)*. Springer International Publishing, pp. C1–C1. doi: 10.1007/978-3-030-91218-5_11.

Bijalwan, A. (2020) ‘Botnet Forensic Analysis Using Machine Learning’, *Security and Communication Networks*. Hindawi Limited, 2020. doi: 10.1155/2020/9302318.

Birk, D. and Wegener, C. (2011) ‘Technical issues of forensic investigations in cloud computing environments’, in *2011 6th IEEE International Workshop on Systematic Approaches to Digital Forensic Engineering, SADFE 2011*. IEEE, pp. 1–10. doi: 10.1109/SADFE.2011.17.

Bolton, R. J., Hand, D. J. and H, D. J. (2001) ‘Unsupervised Profiling Methods for Fraud Detection’, *Proc. Credit Scoring and Credit Control VII*, pp. 5–7. Available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.24.5743 (Accessed: 19 July 2021).

Boser, B. E., Guyon, I. M. and Vapnik, V. N. (1992) ‘Training algorithm for optimal margin classifiers’, in *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*. Publ by ACM, pp. 144–152. doi: 10.1145/130385.130401.

Bouchard, K., Ajroud, A., Bouchard, B. and Bouzouane, A. (2010) ‘SIMACT: A 3D open source smart home simulator for activity recognition', in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, Berlin, Heidelberg, pp. 524–533. doi: 10.1007/978-3-642-13577-4_47.

Bouchaud, F., Grimaud, G. and Vantroys, T. (2018) 'IoT Forensic', in *Proceedings of the 13th International Conference on Availability, Reliability and Security - ARES 2018*. New York, New York, USA: ACM Press, pp. 1–9. doi: 10.1145/3230833.3233257.

Branch, J. W., Giannella, C., Szymanski, B., Wolff, R. and Kargupta, H. (2013) 'In-network outlier detection in wireless sensor networks', *Knowledge and Information Systems*. Springer, 34(1), pp. 23–54. doi: 10.1007/s10115-011-0474-5.

Brause, R., Langsdorf, T. and Hepp, M. (1999) 'Neural data mining for credit card fraud detection', in *Proceedings of the International Conference on Tools with Artificial Intelligence*. IEEE, pp. 103–106. doi: 10.1109/tai.1999.809773.

Breuniq, M. M., Kriegel, H. P., Ng, R. T. and Sander, J. (2000) 'LOF: Identifying density-based local outliers', *SIGMOD Record (ACM Special Interest Group on Management of Data)*. New York, New York, USA: ACM Press, 29(2), pp. 93–104. doi: 10.1145/335191.335388.

Brown, A., Hutchinson, B., Tuor, A. and Nichols, N. (2018) 'Recurrent neural network attention mechanisms for interpretable system log anomaly detection', in *Proceedings of the 1st Workshop on Machine Learning for Computing Systems, MLCS 2018 - In conjunction with HPDC*. Association for Computing Machinery, Inc. doi: 10.1145/3217871.3217872.

Bruneau, J., Jouve, W. and Consel, C. (2012) 'DiaSim: A parameterized simulator for pervasive computing applications', in. Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (ICST). doi: 10.4108/icst.mobiquitous2009.6851.

Butterfield, E., Dixon, M., Miller, S. and Schreuders, Z. (2018) 'Automated Digital Forensics'.

Cai, Y., Ma, Y., Yang, H. and Hang, H. (2023) ‘Bagged Regularized k-Distances for Anomaly Detection’, pp. 1–50. Available at: http://arxiv.org/abs/2312.01046.

Cartier, A. D., Lee, D. H., Kantarci, B. and Foschini, L. (2018) ‘IoT-big data software ecosystems for smart cities sensing: challenges, open issues, and emerging solutions’, in *Communications in Computer and Information Science*. Springer, Cham, pp. 5–18. doi: 10.1007/978-3-319-72125-5_1.

Chandola, V., Banerjee, A. and Kumar, V. (2009) ‘Anomaly detection: A survey’, *ACM Reference Format*, 41(15). doi: 10.1145/1541880.1541882.

Chang, C. C. and Lin, C. J. (2011) ‘LIBSVM: A Library for support vector machines’, *ACM Transactions on Intelligent Systems and Technology*. ACM PUB27 New York, NY, USA, 2(3), p. 27. doi: 10.1145/1961189.1961199.

Chen, C., Shi, Y. Q. and Su, W. (2008) ‘A machine learning based scheme for double JPEG compression detection’, in *2008 19th International Conference on Pattern Recognition*. IEEE, pp. 1–4. doi: 10.1109/ICPR.2008.4761645.

Chen, D., Shao, X., Hu, B. and Su, Q. (2005) ‘Simultaneous wavelength selection and outlier detection in multivariate regression of near-infrared spectra’, *Analytical Sciences*. Anal Sci, 21(2), pp. 161–166. doi: 10.2116/analsci.21.161.

Chen, L. (2019) ‘Intelligent Malware Detection Using File-to-file Relations and Enhancing its Security against Adversarial Attacks’. Available at: https://researchrepository.wvu.edu/etd (Accessed: 17 August 2022).

Chen, T., Liu, X., Xia, B., Wang, W. and Lai, Y. (2020) ‘Unsupervised anomaly detection of industrial robots using sliding-window convolutional variational autoencoder’, *IEEE Access*. Institute of Electrical and Electronics Engineers Inc., 8, pp. 47072–47081. doi: 10.1109/ACCESS.2020.2977892.

Cheng, D., Wang, X., Zhang, Y. and Zhang, L. (2022) ‘Graph Neural Network for Fraud Detection via Spatial-Temporal Attention’, *IEEE Transactions on Knowledge and Data Engineering*. IEEE Computer Society, 34(8), pp. 3800–3813. doi: 10.1109/TKDE.2020.3025588.

Chernyshev, M., Zeadally, S., Baig, Z. and Woodward, A. (2018) 'Internet of things forensics: The need, process models, and open issues', *IT Professional*, 20(3), pp. 40–49. doi: 10.1109/MITP.2018.032501747.

Chhabra, G. S., Singh, V. P. and Singh, M. (2018) 'Cyber forensics framework for big data analytics in IoT environment using machine learning', *Multimedia Tools and Applications*, 13 July, pp. 1–20. doi: 10.1007/s11042-018-6338-1.

Chi, H., Aderibigbe, T. and Granville, B. C. (2019) 'A Framework for IoT Data Acquisition and Forensics Analysis', in *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*. IEEE, pp. 5142–5146. doi: 10.1109/BigData.2018.8622019.

Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin (2008) 'A Practical Guide to Support Vector Classification', *BJU international*, 101(1), pp. 1396–1400. Available at: http://www.csie.ntu.edu.tw/%7B~%7Dcjlin/papers/guide/guide.pdf.

Chike, C. P. (2018) 'The Legal Challenges of Internet of Things Mass Communications View project Cybersecurity Law View project'. doi: 10.13140/RG.2.2.31475.84004.

Cisco (2016) *At-a-Glance Connected Means Informed*. Available at: www.cisco.com/go/iot. (Accessed: 28 November 2018).

Cohen, S., Glaser, B. G. and Strauss, A. L. (2017) 'The Discovery of Grounded Theory: Strategies for Qualitative Research', *The British Journal of Sociology*, 20(2), p. 227. doi: 10.2307/588533.

Collins, A., Fleisher, A. J., Freeman, R. and Maughan, A. (2014) *SCL: The Internet of Things: The Old Problem Squared*. Available at: https://www.scl.org/articles/3055-the-internet-of-things-the-old-problem-squared (Accessed: 24 October 2019).

Conti, M., Dehghantanha, A., Franke, K. and Watson, S. (2018) 'Internet of Things security and forensics: Challenges and opportunities', *Future Generation Computer Systems*, January, pp. 544–546. doi: 10.1016/j.future.2017.07.060.

Cook, D. J., Youngblood, M., Heierman, E. O., Gopalratnam, K., Rao, S., Litvin, A. and Khawaja, F. (2003) 'MavHome: An agent-based smart home', in *Proceedings of* *the 1st IEEE International Conference on Pervasive Computing and Communications, PerCom 2003*, pp. 521–524. doi: 10.1109/percom.2003.1192783.

Copeland, J. (2000) *AlanTuring.net What is AI?* Available at: http://www.alanturing.net/turing_archive/pages/reference articles/what is ai.html (Accessed: 24 October 2019).

Ćosić, J., Ćosić, Z. and Bača, M. (2012) 'Knowledge Sharing and Reuse in Digital Forensic Domain a Review', *ITIS 2012 Novo Mesto-Slovenija*, (January 2016). doi: 10.13140/RG.2.1.3852.0405.

Costantini, S., De Gasperis, G. and Olivieri, R. (2019) 'Digital forensics and investigations meet artificial intelligence', *Annals of Mathematics and Artificial Intelligence*. Springer, 86(1–3), pp. 193–229. doi: 10.1007/s10472-019-09632-y.

D'Orazio, C. J., Choo, K. K. R. and Yang, L. T. (2017) 'Data Exfiltration from Internet of Things Devices: IOS Devices as Case Studies', *IEEE Internet of Things Journal*, 4(2), pp. 524–535. doi: 10.1109/JIOT.2016.2569094.

Dehghantanha, A. and Franke, K. (2014) 'Privacy-respecting digital investigation', in *2014 12th Annual Conference on Privacy, Security and Trust, PST 2014*. IEEE, pp. 129–138. doi: 10.1109/PST.2014.6890932.

Dilek, S., Cakır, H. and Aydın, M. (2015) 'Applications of Artificial Intelligence Techniques to Combating Cyber Crimes: A Review', *International Journal of Artificial Intelligence & Applications*. Academy and Industry Research Collaboration Center (AIRCC), 6(1), pp. 21–39. doi: 10.5121/ijaia.2015.6102.

Diro, A., Chilamkurti, N., Nguyen, V. D. and Heyne, W. (2021) 'A comprehensive study of anomaly detection schemes in iot networks using machine learning algorithms', *Sensors*. Multidisciplinary Digital Publishing Institute, p. 8320. doi: 10.3390/s21248320.

Dorai, G., Houshmand, S. and Baggili, I. (2018) 'I Know What You Did Last Summer', in *Proceedings of the 13th International Conference on Availability, Reliability and Security - ARES 2018*, pp. 1–10. doi: 10.1145/3230833.3232814.

Dunsin, D., Ghanem, M. C., Ouazzane, K. and Vassilev, V. (2023) ‘A Comprehensive Analysis of the Role of Artificial Intelligence and Machine Learning in Modern Digital Forensics and Incident Response Article info’, *Forensic Science International: Digital Investigation*. Elsevier, 48, p. 301675. doi: 10.1016/J.FSIDI.2023.301675.

Easterby-Smith, M., Thorpe, R. and Jackson, P. (Paul R. . (2002) *Management research.* 2nd edn. London: SAGE PublicationsSage UK: London, England.

Eden, P., Blyth, A., Jones, K., Soulsby, H., Burnap, P., Cherdantseva, Y. and Stoddart, K. (2017) ‘SCADA System Forensic Analysis Within IIoT’, in. Springer, Cham, pp. 73–101. doi: 10.1007/978-3-319-50660-9_4.

Edgar, T. W. and Manz, D. O. (2017) *Research Methods for Cyber Security*, *Research Methods for Cyber Security*. Elsevier Inc. doi: 10.1016/s1353-4858(18)30053-9.

Ertöz, L., Steinbach, M. and Kumar, V. (2004) ‘Finding Topics in Collections of Documents: A Shared Nearest Neighbor Approach’, in. Springer, Boston, MA, pp. 83–103. doi: 10.1007/978-1-4613-0227-8_3.

Esmaeili, M., Toosi, A., Roshanpoor, A., Changizi, V., Ghazisaeedi, M., Rahmim, A. and Sabokrou, M. (2023) ‘Generative Adversarial Networks for Anomaly Detection in Biomedical Imaging: A Study on Seven Medical Image Datasets’, *IEEE Access*. Institute of Electrical and Electronics Engineers Inc., 11, pp. 17906–17921. doi: 10.1109/ACCESS.2023.3244741.

Ester, M., Kriegel, H.-P., Sander, J. and Xu, X. (1996) ‘A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise’, in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, pp. 226–231. Available at: www.aaai.org (Accessed: 29 June 2021).

European Union (2016) ‘Regulation, G. D. P. (2016). Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing D’, *Official Journal of the European Union*, 59(294), pp. 1–88.

Fagbola, F. I. and Venter, H. (2022) ‘Smart Digital Forensic Readiness Model for Shadow IoT Devices’, *Applied Sciences (Switzerland)*. Multidisciplinary Digital Publishing Institute, 12(2), p. 730. doi: 10.3390/app12020730.

Al Fahdi, M., Clarke, N. L., Li, F. and Furnell, S. M. (2016) ‘A suspect-oriented intelligent and automated computer forensic analysis’, *Digital Investigation*. Elsevier, 18, pp. 65–76. doi: 10.1016/j.diin.2016.08.001.

Falcão, F., Santos, A., Zoppi, T., Fonseca, B., Bondavalli, A., Silva, C. B. V. and Ceccarelli, A. (2019) ‘Quantitative comparison of unsupervised anomaly detection algorithms for intrusion detection’, in *Proceedings of the ACM Symposium on Applied Computing*. Association for Computing Machinery, pp. 318–327. doi: 10.1145/3297280.3297314.

Feng, X. and Zhao, Y. (2018) ‘Digital forensics challenges to big data in the cloud’, in *Proceedings - 2017 IEEE International Conference on Internet of Things, IEEE Green Computing and Communications, IEEE Cyber, Physical and Social Computing, IEEE Smart Data, iThings-GreenCom-CPSCom-SmartData 2017*. IEEE, pp. 858–862. doi: 10.1109/iThings-GreenCom-CPSCom-SmartData.2017.132.

Franke, K. and Srihari, S. N. (2008) ‘Computational forensics: An overview’, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Berlin, Heidelberg: Springer, Berlin, Heidelberg, 5158 LNCS, pp. 1–10. doi: 10.1007/978-3-540-85303-9_1/COVER.

Fu, K., Cheng, D., Tu, Y. and Zhang, L. (2016) ‘Credit card fraud detection using convolutional neural networks’, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, Cham, pp. 483–490. doi: 10.1007/978-3-319-46675-0_53.

Fu, Q., Li, P., Chen, C., Qi, L., Lu, Y. and Yu, C. (2011) ‘A configurable context-aware simulator for smart home systems’, in *Proceedings - 2011 6th International Conference on Pervasive Computing and Applications, ICPCA 2011*, pp. 39–44. doi: 10.1109/ICPCA.2011.6106476.

Ganesh, N. S. G., Venkatesh, N. G. M. and Prasad, D. V. V. (2022) ‘A Systematic Literature Review on Forensics in Cloud, IoT, AI & Blockchain’, *Lecture Notes on Data Engineering and Communications Technologies*. Springer Science and Business Media Deutschland GmbH, 109, pp. 197–229. doi: 10.1007/978-3-030-93453-8_9/FIGURES/3.

Garfinkel, S. L. (2010) ‘Digital forensics research: The next 10 years’, *Digital Investigation*, 7(SUPPL.). doi: 10.1016/j.diin.2010.05.009.

Godfrey, B. (2000) ‘Electronic work monitoring: An ethical model’, *Selected papers from the second Australian Institute of Computer Ethics Conference*, 1(figure 1), pp. 18–21. Available at: http://crpit.scem.westernsydney.edu.au/confpapers/CRPITV1Godfrey.pdf (Accessed: 6 September 2018).

Goldstein, M. and Dengel, A. (2012) ‘Histogram-based outlier score (hbos): A fast unsupervised anomaly detection algorithm’, *KI-2012: Poster and Demo Track*, (1), pp. 59–63. Available at: http://madm.dfki.de/rapidminer/anomalydetection. (Accessed: 11 July 2021).

Goldstein, M. and Uchida, S. (2016) ‘A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data’, *PLoS ONE*. Public Library of Science, 11(4), p. e0152173. doi: 10.1371/journal.pone.0152173.

Goodfellow, I. J., Shlens, J. and Szegedy, C. (2015) ‘Explaining and harnessing adversarial examples’, in *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR. doi: 10.48550/arxiv.1412.6572.

Goudbeek, A., Choo, K. K. R. and Le-Khac, N. A. (2018) ‘A Forensic Investigation Framework for Smart Home Environment’, in *Proceedings - 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications and 12th IEEE International Conference on Big Data Science and Engineering, Trustcom/BigDataSE 2018*. IEEE, pp. 1446–1451. doi: 10.1109/TrustCom/BigDataSE.2018.00201.

Gu, X., Akoglu, L. and Rinaldo, A. (2019) ‘Statistical analysis of nearest neighbor methods for anomaly detection’, in *Advances in Neural Information Processing Systems*. Available at: https://github.com/xgu1/DTM (Accessed: 29 February 2024).

Guha, S., Rastogi, R. and Shim, K. (2000) ‘Rock: a robust clustering algorithm for categorical attributes’, *Information Systems*. Elsevier Science Ltd, 25(5), pp. 345–366. doi: 10.1016/S0306-4379(00)00022-3.

Guo, C., Pleiss, G., Sun, Y. and Weinberger, K. Q. (2017) ‘On calibration of modern neural networks’, in *34th International Conference on Machine Learning, ICML 2017*. PMLR, pp. 2130–2143. Available at: https://proceedings.mlr.press/v70/guo17a.html (Accessed: 13 October 2022).

Gupta, C., Johri, I., Srinivasan, K., Hu, Y. C., Qaisar, S. M. and Huang, K. Y. (2022) ‘A Systematic Review on Machine Learning and Deep Learning Models for Electronic Information Security in Mobile Networks’, *Sensors*. Multidisciplinary Digital Publishing Institute, p. 2017. doi: 10.3390/s22052017.

HaddadPajouh, H., Dehghantanha, A., Khayami, R. and Choo, K. K. R. (2018) ‘A deep Recurrent Neural Network based approach for Internet of Things malware threat hunting’, *Future Generation Computer Systems*. Elsevier B.V., 85, pp. 88–96. doi: 10.1016/j.future.2018.03.007.

Haque, S. A., Rahman, M. and Aziz, S. M. (2015) ‘Sensor anomaly detection in wireless sensor networks for healthcare’, *Sensors (Switzerland)*. Multidisciplinary Digital Publishing Institute, 15(4), pp. 8764–8786. doi: 10.3390/s150408764.

Harbawi, M. and Varol, A. (2017) ‘An improved digital evidence acquisition model for the Internet of Things forensic I: A theoretical framework’, in *2017 5th International Symposium on Digital Forensic and Security, ISDFS 2017*. IEEE, pp. 1–6. doi: 10.1109/ISDFS.2017.7916508.

Harichandran, V. S., Breitinger, F., Baggili, I. and Marrington, A. (2016) ‘A cyber forensics needs analysis survey: Revisiting the domain’s needs a decade later’, *Computers and Security*, 57, pp. 1–13. doi: 10.1016/j.cose.2015.10.007.

Hassan, M. A., Samara, G. and Fadda, M. A. (2022) 'IoT Forensic Frameworks (DFIF, IoTDOTS, FSAIoT): A Comprehensive Study', *International Journal of Advances in Soft Computing and its Applications*, 14(1), pp. 72–86. doi: 10.15849/IJASCA.220328.06.

Hauser, C. (2017) *In Connecticut Murder Case, a Fitbit Is a Silent Witness*, *New York Times*. Available at: https://www.nytimes.com/2017/04/27/nyregion/in-connecticut-murder-case-a-fitbit-is-a-silent-witness.html (Accessed: 14 July 2020).

Hawkins, D. M. (1980) *Identification of Outliers*, *Identification of Outliers*. Springer Netherlands. doi: 10.1007/978-94-015-3994-4.

He, Z., Xu, X. and Deng, S. (2003) 'Discovering cluster-based local outliers', *Pattern Recognition Letters*. Elsevier, 24(9–10), pp. 1641–1650. doi: 10.1016/S0167-8655(03)00003-5.

Hegarty, R. C., Lamb, D. J. and Attwood, A. (2014) 'Digital evidence challenges in the internet of things', *10th International Network Conference, INC 2014*, pp. 163–172. Available at: https://pdfs.semanticscholar.org/b789/f84ef58d5963996e134aa51fd9d01613b922.pdf (Accessed: 12 September 2018).

Helal, S., Lee, J. W., Hossain, S., Kim, E., Hagras, H. and Cook, D. (2011) 'Persim - Simulator for human activities in pervasive spaces', in *Proceedings - 2011 7th International Conference on Intelligent Environments, IE 2011*, pp. 192–199. doi: 10.1109/IE.2011.34.

Hempstalk, K. and Frank, E. (2008) 'Discriminating aainst new classes: One-class versus multi-class classification', in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, Berlin, Heidelberg, pp. 325–336. doi: 10.1007/978-3-540-89378-3_32.

Hikmatyar, M., Prayudi, Y. and Riadi, I. (2017) 'Network Forensics Framework Development using Interactive Planning Approach', *International Journal of Computer Applications*. Foundation of Computer Science (FCS), NY, USA, 161(10), pp. 41–48.

doi: 10.5120/ijca2017913352.

Hildebrandt, M. (2008) ‘Defining profiling: A new type of knowledge?’, in *Profiling the European Citizen: Cross-Disciplinary Perspectives*. Springer Netherlands, pp. 17–45. doi: 10.1007/978-1-4020-6914-7_2.

Hinton, G., Vinyals, O. and Dean, J. (2015) ‘Distilling the Knowledge in a Neural Network’, pp. 1–9. Available at: http://arxiv.org/abs/1503.02531.

Hogan, K. and Maglienti, M. (2001) ‘Comparing the epistemological underpinnings of students’ and scientists’ reasoning about conclusions’, *Journal of Research in Science Teaching*. John Wiley & Sons, Ltd, 38(6), pp. 663–687. doi: 10.1002/tea.1025.

Hon, W. K., Millard, C. and Singh, J. (2016) *Twenty Legal Considerations for Clouds of Things*, *SSRN*. doi: 10.2139/ssrn.2716966.

Hossain, M., Hasan, R. and Zawoad, S. (2017) ‘Trust-IoV: A trustworthy forensic investigation framework for the internet of vehicles (IoV)’, in *Proceedings - 2017 IEEE 2nd International Congress on Internet of Things, ICIOT 2017*. IEEE, pp. 25–32. doi: 10.1109/IEEE.ICIOT.2017.13.

Hossain, M., Hasan, R. and Zawoad, S. (2018) ‘Probe-IoT: A public digital ledger based forensic investigation framework for IoT’, in *INFOCOM 2018 - IEEE Conference on Computer Communications Workshops*. IEEE, pp. 1–2. doi: 10.1109/INFCOMW.2018.8406875.

Hossain, M., Karim, Y. and Hasan, R. (2018) ‘FIF-IoT: A forensic investigation framework for IoT using a public digital ledger’, in *Proceedings - 2018 IEEE International Congress on Internet of Things, ICIOT 2018 - Part of the 2018 IEEE World Congress on Services*. IEEE, pp. 33–40. doi: 10.1109/ICIOT.2018.00012.

Huang, C., Lu, R. and Choo, K. K. R. (2017) ‘Vehicular Fog Computing: Architecture, Use Case, and Security and Forensic Challenges’, *IEEE Communications Magazine*, 55(11), pp. 105–111. doi: 10.1109/MCOM.2017.1700322.

Idé, T., Papadimitriou, S. and Vlachos, M. (2007) ‘Computing correlation anomaly scores using stochastic nearest neighbors’, in *Proceedings - IEEE International*

Conference on Data Mining, ICDM, pp. 523–528. doi: 10.1109/ICDM.2007.12.

Induruwa, A. (2011) ‘Hidden in the clouds: The impact on data security and forensic investigation’, in. Institute of Electrical and Electronics Engineers (IEEE), pp. 77–77. doi: 10.1109/icter.2011.6075014.

Iqbal, S., Abed Alharbi, S., Alharbi, S. A., Iqbal, S., Alharbi, S. A. and Homem, I. (2018) *Advancing Automation in Digital Forensic Investigations Using Machine Learning Forensics*, *Digital Forensic Science*. IntechOpen. doi: 10.5772/intechopen.90233.

Irons, A. and Lallie, H. (2014) ‘Digital Forensics to Intelligent Forensics’, *Future Internet*. Multidisciplinary Digital Publishing Institute, 6(3), pp. 584–596. doi: 10.3390/fi6030584.

Islam, M. J., Mahin, M., Khatun, A., Debnath, B. C. and Kabir, S. (2019) ‘Digital Forensic Investigation Framework for Internet of Things (IoT): A Comprehensive Approach’, in *1st International Conference on Advances in Science, Engineering and Robotics Technology 2019, ICASERT 2019*. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ICASERT.2019.8934707.

Jahankhani, H. and Hosseinian-Far, A. (2017) ‘Challenges of cloud forensics’, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, Cham, pp. 1–18. doi: 10.1007/978-3-319-54380-2_1.

James, J. I. and Gladyshev, P. (2013) ‘Challenges with Automation in Digital Forensic Investigations’. doi: 10.48550/arxiv.1303.4498.

Janakiram, D., Reddy V, A. M. and Kumar, A. V. U. P. (2006) ‘Outlier detection in wireless sensor networks using bayesian belief networks’, in *First International Conference on Communication System Software and Middleware, Comsware 2006*. doi: 10.1109/comswa.2006.1665221.

Jarrett, A. and Choo, K.-K. R. K. R. (2021) ‘The impact of automation and artificial intelligence on digital forensics’, *WIREs Forensic Science*. Wiley, 3(6), p. e1418. doi: 10.1002/wfs2.1418.

Jiang, M. F., Tseng, S. S. and Su, C. M. (2001) ‘Two-phasee clustering process for outliers detection’, *Pattern Recognition Letters*. North-Holland, 22(6–7), pp. 691–700. doi: 10.1016/S0167-8655(00)00131-8.

Jin, W., Tung, A. K. H., Han, J. and Wang, W. (2006) ‘Ranking outliers using symmetric neighborhood relationship’, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, Berlin, Heidelberg, pp. 577–593. doi: 10.1007/11731139_68.

Kamarinou, D., Millard, C. and Singh, J. (2017) ‘Machine Learning with Personal Data’. Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2865811 (Accessed: 10 September 2018).

Kanimozhi, V. and Jacob, T. (2019) ‘Artificial intelligence based network intrusion detection with hyper-parameter optimization tuning on the realistic cyber dataset CSE-CIC-IDS2018 using cloud computing’, in *Proceedings of the 2019 IEEE International Conference on Communication and Signal Processing, ICCSP 2019*. Institute of Electrical and Electronics Engineers Inc., pp. 33–36. doi: 10.1109/ICCSP.2019.8698029.

Karie, N. M., Kebande, V. R. and Venter, H. S. (2019) ‘Diverging deep learning cognitive computing techniques into cyber forensics’, *Forensic Science International: Synergy*. Elsevier, 1, pp. 61–67. doi: 10.1016/j.fsisyn.2019.03.006.

Kebande, V. R. *et al.* (2018) ‘Towards an integrated digital forensic investigation framework for an IoT-based ecosystem’, in *Proceedings - 2018 IEEE International Conference on Smart Internet of Things, SmartIoT 2018*. IEEE, pp. 93–98. doi: 10.1109/SmartIoT.2018.00-19.

Kebande, V. R., Karie, N. M. and Venter, H. S. (2017) ‘Cloud-Centric Framework for isolating Big data as forensic evidence from IoT infrastructures’, in *2017 1st International Conference on Next Generation Computing Applications, NextComp 2017*. IEEE, pp. 54–60. doi: 10.1109/NEXTCOMP.2017.8016176.

Kebande, V. R., Karie, N. M. and Venter, H. S. (2018) ‘Adding Digital Forensic

Readiness as a Security Component to the IoT Domain', *International Journal on Advanced Science, Engineering and Information Technology*, 8(1), p. 1. doi: 10.18517/ijaseit.8.1.2115.

Kebande, V. R. and Ray, I. (2016) 'A generic digital forensic investigation framework for Internet of Things (IoT)', in *Proceedings - 2016 IEEE 4th International Conference on Future Internet of Things and Cloud, FiCloud 2016*. IEEE, pp. 356–362. doi: 10.1109/FiCloud.2016.57.

Kim, I. S., Park, H. M., Lee, Y. L., Lee, S. Y., Lee, H. H. and Noh, B. N. (2006) 'Design and implementation of context-awareness simulation toolkit for context learning', in *Proceedings - IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, pp. 96–103. doi: 10.1109/SUTC.2006.51.

Kim, J., Park, J. and Lee, S. (2023) 'An improved IoT forensic model to identify interconnectivity between things', *Forensic Science International: Digital Investigation*. Elsevier, 44, p. 301499. doi: 10.1016/j.fsidi.2022.301499.

Kim, S., Park, M., Lee, S. and Kim, J. (2020) 'Smart home forensics—data analysis of iot devices', *Electronics (Switzerland)*. MDPI AG, 9(8), pp. 1–13. doi: 10.3390/electronics9081215.

Kim, T. hoon, Ramos, C. and Mohammed, S. (2017) 'Smart City and IoT', *Future Generation Computer Systems*, 1 November, pp. 159–162. doi: 10.1016/j.future.2017.03.034.

Kitchenham, B. (2007) 'Guidelines for performing Systematic Literature Reviews in Software Engineering', *Software Engineering Group School of Computer Science and Mathematics*, p. 65. doi: 10.1145/1134285.1134500.

Kittidachanan, K., Minsan, W., Pornnopparath, D. and Taninpong, P. (2020) 'Anomaly detection based on GS-OCSVM classification', *KST 2020 - 2020 12th International Conference on Knowledge and Smart Technology*. Institute of Electrical and Electronics Engineers Inc., pp. 64–69. doi: 10.1109/KST48564.2020.9059326.

Kohonen, T. (1995) *Self-Organizing Maps*. Berlin, Heidelberg: Springer Berlin

Heidelberg (Springer Series in Information Sciences). doi: 10.1007/978-3-642-97610-0.

Kokott, J. and Sobotta, C. (2013) ‘The distinction between privacy and data protection in the jurisprudence of the CJEU and the ECtHR’, *International Data Privacy Law*. Oxford University Press (OUP), 3(4), pp. 222–228. doi: 10.1093/idpl/ipt017.

Koroniotis, N., Moustafa, N. and Sitnikova, E. (2020) ‘A new network forensic framework based on deep learning for Internet of Things networks: A particle deep framework’, *Future Generation Computer Systems*. North-Holland, 110, pp. 91–106. doi: 10.1016/j.future.2020.03.042.

Koroniotis, N., Moustafa, N., Sitnikova, E. and Slay, J. (2018) ‘towards developing network forensic mechanism for botnet activities in the IoT based on machine learning techniques’, in *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*. Springer, Cham, pp. 30–44. doi: 10.1007/978-3-319-90775-8_3.

Kriegel, H. P., Kröger, P., Schubert, E. and Zimek, A. (2009) ‘LoOP: Local outlier probabilities’, in *International Conference on Information and Knowledge Management, Proceedings*. New York, New York, USA: ACM Press, pp. 1649–1652. doi: 10.1145/1645953.1646195.

Küffner, R., Petri, T., Windhager, L. and Zimmer, R. (2010) ‘Petri nets with fuzzy logic (PNFL): Reverse engineering and parametrization’, *PLoS ONE*, 5(9), pp. 1–10. doi: 10.1371/journal.pone.0012807.

Kuhl, M. E., Steiger, N. M., Lada, E. K., Wagner, M. A. and Wilson, J. R. (2006) ‘Introduction to modeling and generating probabilistic input processes for simulation’, in *Proceedings - Winter Simulation Conference*, pp. 19–35. doi: 10.1109/WSC.2006.323035.

Kulatunga, K., Amaratunga, D. and Haigh, R. (2007) ‘Researching construction client and innovation: methodological perspective’, *7th International Postgraduate Research Conference in the Built and Human Environment, 27th - 28th March*, pp. 479–488.

Kumar Eugene Spafford, S. H., deep Kumar, S. and Spafford, E. H. (1994) *An Application of Pattern Matching in Intrusion Detection*. Available at: https://docs.lib.purdue.edu/cstech/1116 (Accessed: 14 October 2020).

Kwitt, R. and Hofmann, U. (2007) ‘Unsupervised Anomaly Detection in Network Traffic by Means of Robust PCA’, in. Institute of Electrical and Electronics Engineers (IEEE), pp. 37–37. doi: 10.1109/iccgi.2007.62.

Lazarevic, A., Ertoz, L., Kumar, V., Ozgur, A. and Srivastava, J. (2003) ‘A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection’, in, pp. 25–36. doi: 10.1137/1.9781611972733.3.

Le, D. P., Meng, H., Su, L., Yeo, S. L. and Thing, V. (2019) ‘BIFF: A Blockchain-based IoT Forensics Framework with Identity Privacy’, in *IEEE Region 10 Annual International Conference, Proceedings/TENCON*. IEEE, pp. 2372–2377. doi: 10.1109/TENCON.2018.8650434.

Lee, J. W., Cho, S., Liu, S., Cho, K. and Helal, S. (2015) ‘Persim 3D: Context-Driven Simulation and Modeling of Human Activities in Smart Spaces’, *IEEE Transactions on Automation Science and Engineering*. Institute of Electrical and Electronics Engineers Inc., 12(4), pp. 1243–1256. doi: 10.1109/TASE.2015.2467353.

Lei, Z., Yue, S., Yu, C. and Shi, Y. (2010) ‘SHSim: An OSGI-based smart home simulator’, in *2010 3rd IEEE International Conference on Ubi-Media Computing, U-Media 2010*, pp. 87–90. doi: 10.1109/UMEDIA.2010.5543920.

Lertlakkhanakul, J., Choi, J. W. and Kim, M. Y. (2008) ‘Building data model and simulation platform for spatial interaction management in smart home’, *Automation in Construction*. Elsevier, 17(8), pp. 948–957. doi: 10.1016/j.autcon.2008.03.004.

Li, H., Jiang, S., Zhang, L., Du, S., Ye, G. and Chai, H. (2023) ‘Fraud Detection with Binding Global and Local Relational Interaction’, *Proceedings of (Under Review)*, 1. Available at: http://arxiv.org/abs/2402.17472 (Accessed: 7 April 2024).

Liang, S., Li, Y. and Srikant, R. (2018) ‘Enhancing the reliability of out-of-distribution image detection in neural networks’, in *6th International Conference on Learning*

Representations, ICLR 2018 - Conference Track Proceedings. International Conference on Learning Representations, ICLR. doi: 10.48550/arxiv.1706.02690.

Lin, H. and Lin, C. (2003) ‘A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods’, *Neural Computation*, (2), pp. 1–32. Available at: http://home.caltech.edu/~htlin/publication/doc/tanh.pdf.

Lippmann, R., Haines, J. W., Fried, D. J., Korba, J. and Das, K. (2000) ‘1999 DARPA off-line intrusion detection evaluation’, *Computer Networks*. Elsevier, 34(4), pp. 579–595. doi: 10.1016/S1389-1286(00)00139-0.

Liu, F. T., Ting, K. M. and Zhou, Z.-H. (2008) ‘Isolation Forest’, in *2008 Eighth IEEE International Conference on Data Mining*. IEEE, pp. 413–422. doi: 10.1109/ICDM.2008.17.

Liu, Z. and Xu, H. (2014) ‘Kernel parameter selection for support vector machine classification’, *Journal of Algorithms and Computational Technology*. SAGE PublicationsSage UK: London, England, 8(2), pp. 163–177. doi: 10.1260/1748-3018.8.2.163.

Liu, Z., Zeng, Y., Yan, Y., Zhang, P. and Wang, Y. (2017) ‘Machine Learning for Analyzing Malware’, *Journal of Cyber Security and Mobility*. River Publishers, pp. 227–244–227–244. doi: 10.13052/2245-1439.631.

Lorig, F. and Timm, I. J. (2020) ‘Simulation-Based Data Acquisition’, in. Springer, Cham, pp. 1–15. doi: 10.1007/978-3-030-43981-1_1.

Losavio, M. M., Chow, K. P., Koltay, A. and James, J. (2018) ‘The Internet of Things and the Smart City: Legal challenges with digital forensics, privacy, and security’, *Security and Privacy*. John Wiley & Sons, Ltd, 1(3), p. e23. doi: 10.1002/spy2.23.

Lutta, P. (2023) *Smart Home Forensic Datatset*, *GitHub*. Available at: https://github.com/plolutta/dataset.git.

Lutta, P., Sedky, M., Hassan, M., Jayawickrama, U. and Bakhtiari Bastaki, B. (2021) ‘The complexity of internet of things forensics: A state-of-the-art review’, *Forensic Science International: Digital Investigation*. Elsevier Ltd, 38, p. 301210. doi:

10.1016/j.fsidi.2021.301210.

Macdermott, Á., Baker, T., Shi, Q., MacDermott, A., Baker, T. and Shi, Q. (2018) ‘Iot Forensics: Challenges for the Ioa Era’, in *2018 9th IFIP International Conference on New Technologies, Mobility and Security, NTMS 2018 - Proceedings*. IEEE, pp. 1–5. doi: 10.1109/NTMS.2018.8328748.

Magnet Forensics (2020) *Magnet AUTOMATE*. Available at: https://www.magnetforensics.com/products/magnet-automate/ (Accessed: 26 August 2022).

Makadiya, Y., Virparia, R. and Shah, K. (2023) ‘IoT Forensics System based on Blockchain’, *Proceedings of the 17th INDIACom; 2023 10th International Conference on Computing for Sustainable Global Development, INDIACom 2023*. Bharati Vidyapeeth, New Delhi, pp. 490–495.

Marturana, F., Me, G., Bertè, R. and Tacconi, S. (2011) ‘A quantitative approach to triaging in mobile forensics’, in *Proc. 10th IEEE Int. Conf. on Trust, Security and Privacy in Computing and Communications, TrustCom 2011, 8th IEEE Int. Conf. on Embedded Software and Systems, ICESS 2011, 6th Int. Conf. on FCST 2011*, pp. 582–588. doi: 10.1109/TrustCom.2011.75.

Mazhar, M. S., Saleem, Y., Almogren, A., Arshad, J., Jaffery, M. H., Rehman, A. U., Shafiq, M. and Hamam, H. (2022) ‘Forensic Analysis on Internet of Things (IoT) Device Using Machine-to-Machine (M2M) Framework’, *Electronics (Switzerland)*. Multidisciplinary Digital Publishing Institute, 11(7), p. 1126. doi: 10.3390/electronics11071126.

Meffert, C., Clark, D., Baggili, I. and Breitinger, F. (2017) ‘Forensic State Acquisition from Internet of Things (FSAIoT)’, in *Proceedings of the 12th International Conference on Availability, Reliability and Security - ARES ’17*. New York, New York, USA: ACM Press, pp. 1–11. doi: 10.1145/3098954.3104053.

Megas, K., Piccarreta, B., O’Rourke, D. G., Gabel, D. and ’rourke, O. (2017) *Internet of Things (IoT) Cybersecurity Colloquium A NIST Workshop Proceedings*.

Gaithersburg, MD. doi: 10.6028/NIST.IR.8201.

Mendez-Vazquez, A., Helal, A. and Cook, D. (2009) ‘Simulating events to generate synthetic data for pervasive spaces’, *Proceedings of the Workshop on Developing Shared Home Behaviour Datasets to Advance HCI and Ubiquitous Computing Research in Conjunction with CHI 2009*, pp. 4–9. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.212.2548&rep=rep1&type =pdf (Accessed: 6 July 2020).

Miller, Z., Dickinson, B., Deitrick, W., Hu, W. and Wang, A. H. (2014) ‘Twitter spammer detection using data stream clustering’, *Information Sciences*. Elsevier, 260, pp. 64–73. doi: 10.1016/j.ins.2013.11.016.

Milne, R. (2012) *Forensic Intelligence*, *Forensic Intelligence*. Taylor and Francis. doi: 10.1201/b10137.

Mitchell, F. (2010) ‘The Use of Artificial Intelligence in Digital Forensics: An Introduction’, *Digital Evidence and Electronic Signature Law Review*, 7. Available at: https://heinonline.org/HOL/Page?handle=hein.journals/digiteeslr7&id=35&div=&collection= (Accessed: 11 August 2022).

Mitchell, T. M. (1997) ‘Machine Learning’, *Computer*, 2005(April), p. 414. Available at: https://books.google.ca/books?id=EoYBngEACAAJ&dq=mitchell+machine+learning+1997&hl=en&sa=X&ved=0ahUKEwiomdqfj8TkAhWGslkKHRCbAtoQ6AEIKjAA (Accessed: 24 October 2019).

Mohamed, H., Koroniotis, N. and Moustafa, N. (2023) ‘Digital Forensics based on Federated Learning in IoT Environment’, in *ACM International Conference Proceeding Series*. Association for Computing Machinery, pp. 92–101. doi: 10.1145/3579375.3579387.

Mohammad, R., Saeed, F., Almazroi, Abdulwahab Ali, Alsubaei, F. S. and Almazroi, Abdulaleem Ali (2024) ‘Enhancing Intrusion Detection Systems Using a Deep Learning and Data Augmentation Approach’, *Systems*. Multidisciplinary Digital Publishing Institute, 12(3), p. 79. doi: 10.3390/systems12030079.

Mukundan, R., Madria, S. and Linderman, M. (2014) ‘Efficient integrity verification of replicated data in cloud using homomorphic encryption’, *Distributed and Parallel Databases*. Springer US, 32(4), pp. 507–534. doi: 10.1007/s10619-014-7151-0.

Nehmer, J., Becker, M., Karshmer, A. and Lamm, R. (2006) ‘Living assistance systems’, in. Association for Computing Machinery (ACM), pp. 43–50. doi: 10.1145/1134285.1134293.

Nieto, A., Rios, R. and Lopez, J. (2017) ‘A Methodology for Privacy-Aware IoT-Forensics’, in *2017 IEEE Trustcom/BigDataSE/ICESS*, pp. 626–633. doi: 10.1109/Trustcom/BigDataSE/ICESS.2017.293.

Nik Zulkipli, N. H., Alenezi, A. and B. Wills, G. (2017) ‘IoT Forensic: Bridging the Challenges in Digital Forensic and the Internet of Things’, in *Proceedings of the 2nd International Conference on Internet of Things, Big Data and Security*, pp. 315–324. doi: 10.5220/0006308703150324.

Nishikawa, H., Yamamoto, S., Tamai, M., Nishigaki, K., Kitani, T., Shibata, N., Yasumoto, K. and Ito, M. (2006) ‘UbiREAL: Realistic smartspace simulator for systematic testing’, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer Verlag, pp. 459–476. doi: 10.1007/11853565_27.

Noumir, Z., Honeine, P. and Richard, C. (2012) ‘On simple one-class classification methods’, in *IEEE International Symposium on Information Theory - Proceedings*, pp. 2022–2026. doi: 10.1109/ISIT.2012.6283685.

Nowroozi, E., Dehghantanha, A., Parizi, R. M. and Choo, K. K. R. (2021) ‘A survey of machine learning techniques in adversarial image forensics’, *Computers and Security*. Elsevier Advanced Technology, p. 102092. doi: 10.1016/j.cose.2020.102092.

Oriwoh, E. (2015) ‘A smart home anomaly detection framework’.

Oriwoh, E., Jazani, D., Epiphaniou, G. and Sant, P. (2013) ‘Internet of Things Forensics: Challenges and Approaches’, in *Proceedings of the 9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing*.

ICST, pp. 608–615. doi: 10.4108/icst.collaboratecom.2013.254159.

Oriwoh, E. and Sant, P. (2013) ‘The forensics edge management system: A concept and design’, in *Proceedings - IEEE 10th International Conference on Ubiquitous Intelligence and Computing, UIC 2013 and IEEE 10th International Conference on Autonomic and Trusted Computing, ATC 2013*. IEEE, pp. 544–550. doi: 10.1109/UIC-ATC.2013.71.

Papadimitriou, S., Kitagawa, H., Gibbons, P. B. and Faloutsos, C. (2003) ‘LOCI: Fast outlier detection using the local correlation integral’, in *Proceedings - International Conference on Data Engineering*, pp. 315–326. doi: 10.1109/ICDE.2003.1260802.

Park, J. S., Moon, M., Hwang, S. and Yeom, K. (2007) ‘CASS: A context-aware simulation system for smart home’, in *Proceedings - SERA 2007: Fifth ACIS International Conference on Software Engineering Research, Management, and Applications*, pp. 461–467. doi: 10.1109/SERA.2007.60.

Pavuluri, G. and Annem, G. (2023) ‘A Deep Learning Approach to Video Anomaly Detection using Convolutional Autoencoders’. Available at: http://arxiv.org/abs/2311.04351.

Perumal, S., Md Norwawi, N. and Raman, V. (2015) ‘Internet of Things(IoT) digital forensic investigation model: Top-down forensic approach methodology’, in *2015 5th International Conference on Digital Information Processing and Communications, ICDIPC 2015*. IEEE, pp. 19–23. doi: 10.1109/ICDIPC.2015.7323000.

Phua, C., Alahakoon, D. and Lee, V. (2004) ‘Minority report in fraud detection’, *ACM SIGKDD Explorations Newsletter*. ACM PUB27 New York, NY, USA, 6(1), pp. 50–59. doi: 10.1145/1007730.1007738.

Van Phuong, T., Hung, L. X., Cho, S. J., Lee, Y. K. and Lee, S. (2006) ‘An anomaly detection algorithm for detecting attacks in wireless sensor networks’, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, Berlin, Heidelberg, pp. 735–736. doi: 10.1007/11760146_111.

Pires, A. and Santos-Pereira, C. (2005) ‘Using clustering and robust estimators to detect outliers in multivariate data’, *… of the International Conference on Robust …*, (step 1), pp. 2–3. Available at: http://www.ics.uci.edu/ (Accessed: 29 June 2021).

Platzer, C., Stuetz, M. and Lindorfer, M. (2014) ‘Skin sheriff: A machine learning solution for detecting explicit images’, in *SFCS 2014 - Proceedings of the 2nd International Workshop on Security and Forensics in Communication Systems*. New York, New York, USA: ACM Press, pp. 45–55. doi: 10.1145/2598918.2598920.

Pokrajac, D., Lazarevic, A. and Latecki, L. J. (2007) ‘Incremental local outlier detection for data streams’, in *Proceedings of the 2007 IEEE Symposium on Computational Intelligence and Data Mining, CIDM 2007*, pp. 504–515. doi: 10.1109/CIDM.2007.368917.

Powers, D. M. W. (2011) ‘EVALUATION: FROM PRECISION, RECALL AND F-MEASURE TO ROC, INFORMEDNESS, MARKEDNESS & CORRELATION’.

Prasad, N. R., Almanza-Garcia, S. and Lu, T. T. (2009) ‘Anomaly detection’, *Computers, Materials and Continua*. ACM PUB27 New York, NY, USA, 14(1), pp. 1–22. doi: 10.1145/1541880.1541882.

Purdy, S. (2016) ‘Encoding Data for HTM Systems’. Available at: https://arxiv.org/abs/1602.05925v1 (Accessed: 4 July 2023).

Qadir, A. M. and Varol, A. (2020) ‘The Role of Machine Learning in Digital Forensics’, *8th International Symposium on Digital Forensics and Security, ISDFS 2020*. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ISDFS49300.2020.9116298.

Qatawneh, M., Almobaideen, W., Khanafseh, M., Qatawneh, I. Al and Al Ain, P. (2019) ‘DFIM: A new digital forensics investigation model for internet of things’, *Journal of Theoretical and Applied Information Technology*, 97(24), p. 24. Available at: www.jatit.org (Accessed: 8 January 2020).

Quick, D. and Choo, K. K. R. (2018a) ‘Digital forensic intelligence: Data subsets and Open Source Intelligence (DFINT+OSINT): A timely and cohesive mix’, *Future Generation Computer Systems*, 78, pp. 558–567. doi: 10.1016/j.future.2016.12.032.

Quick, D. and Choo, K. K. R. (2018b) 'IoT Device Forensics and Data Reduction', *IEEE Access*, 6, pp. 47566–47574. doi: 10.1109/ACCESS.2018.2867466.

Raghavan, P. and Gayar, N. El (2019) 'Fraud Detection using Machine Learning and Deep Learning', in *Proceedings of 2019 International Conference on Computational Intelligence and Knowledge Economy, ICCIKE 2019*. Institute of Electrical and Electronics Engineers Inc., pp. 334–339. doi: 10.1109/ICCIKE47802.2019.9004231.

Rahman, K. M. S., Bishop, M. and Holt, A. (2016) 'Internet of Things Mobility Forensics', in *Proceedings of the 2016 Information Security Research and Education (INSuRE) Conference (INSuRECon-16)*, pp. 1–7. Available at: https://developer.amazon.com/alexa-skills-kit (Accessed: 28 January 2019).

Rajeev, H. and Devi, U. (2022) 'Detection of Credit Card Fraud Using Isolation Forest Algorithm', in *Lecture Notes in Networks and Systems*. Springer, Singapore, pp. 23–34. doi: 10.1007/978-981-16-5640-8_3.

Ramaswamy, S., Rastogi, R. and Shim, K. (2000) 'Efficient algorithms for mining outliers from large data sets', in. Association for Computing Machinery (ACM), pp. 427–438. doi: 10.1145/342009.335437.

Renoux, J. and Klügl, F. (2019) 'Simulating daily activities in a smart home for data generation', in *Proceedings - Winter Simulation Conference*. Institute of Electrical and Electronics Engineers Inc., pp. 798–809. doi: 10.1109/WSC.2018.8632226.

Rondeau, C. M., Temple, M. A. and Lopez, J. (2018) 'Industrial IoT cross-layer forensic investigation', *Wiley Interdisciplinary Reviews: Forensic Science*. John Wiley & Sons, Ltd, 1(1), p. e1322. doi: 10.1002/wfs2.1322.

Rousseeuw, P. J. and Leroy, A. M. (2005) *Robust Regression and Outlier Detection*. John Wdey & Sons.

Rughani, P. H. (2017) 'IoT Evidence Acquisition – Issues and Challenges', *Research India Publications*, 10(5), pp. 1285–1293. Available at: http://www.ripublication.com (Accessed: 15 January 2019).

Ryu, J. H., Moon, S. Y. and Park, J. H. (2018) 'The study on data of smart home system as digital evidence', in *Lecture Notes in Electrical Engineering*. Springer, Singapore, pp. 967–972. doi: 10.1007/978-981-10-7605-3_154.

Ryu, J. H., Sharma, P. K., Jo, J. H. and Park, J. H. (2019) 'A blockchain-based decentralized efficient investigation framework for IoT digital forensics', *Journal of Supercomputing*. Springer New York LLC, 75(8), pp. 4372–4387. doi: 10.1007/s11227-019-02779-9.

Sadineni, L., Pilli, E. and Battula, R. B. (2019) 'A holistic forensic model for the internet of things', in *IFIP Advances in Information and Communication Technology*. Springer New York LLC, pp. 3–18. doi: 10.1007/978-3-030-28752-8_1.

Saikia, S., Fidalgo, E., Alegre, E. and Fernández-Robles, L. (2017) 'Object Detection for Crime Scene Evidence Analysis Using Deep Learning', in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer Verlag, pp. 14–24. doi: 10.1007/978-3-319-68548-9_2.

Saleh, M. A., Hajar Othman, S., Al-Dhaqm, A. and Al-Khasawneh, M. A. (2021) 'Common investigation process model for internet of things forensics', in *2021 2nd International Conference on Smart Computing and Electronic Enterprise: Ubiquitous, Adaptive, and Sustainable Computing Solutions for New Normal, ICSCEE 2021*. Institute of Electrical and Electronics Engineers Inc., pp. 84–89. doi: 10.1109/ICSCEE50312.2021.9498045.

Salem, Y., Owda, M. and Owda, A. Y. (2023) 'A Comprehensive Review of Digital Forensics Frameworks for Internet of Things (IoT) Devices', in *2023 International Conference on Information Technology: Cybersecurity Challenges for Sustainable Cities, ICIT 2023 - Proceeding*. Institute of Electrical and Electronics Engineers Inc., pp. 89–96. doi: 10.1109/ICIT58056.2023.10226145.

Sandoval Orozco, A. L., Quinto Huamán, C., Povedano Álvarez, D. and García Villalba, L. J. (2020) 'A machine learning forensics technique to detect post-processing in digital videos', *Future Generation Computer Systems*. North-Holland, 111, pp. 199–212. doi: 10.1016/j.future.2020.04.041.

Saqib, M., Khan, S. D., Sharma, N. and Blumenstein, M. (2019) ‘Crowd Counting in Low-Resolution Crowded Scenes Using Region-Based Deep Convolutional Neural Networks’, *IEEE Access*. Institute of Electrical and Electronics Engineers Inc., 7, pp. 35317–35329. doi: 10.1109/ACCESS.2019.2904712.

Sarker, I. H. (2021) ‘Machine Learning: Algorithms, Real-World Applications and Research Directions’, *SN Computer Science*. Springer, pp. 1–21. doi: 10.1007/s42979-021-00592-x.

Saunders, M. N. ., Thornhill, A. and Lewis, P. (2009) ‘Research Methods for Business Students (5th Edition) Mark N. K. Saunders’, *Pearson Education Limited.* 5th edn. Engaknd: Pearson Education Limited. Available at: https://www.academia.edu/34673883/Research_Methods_for_Business_Students_5th_Edition_Mark_N._K._Saunders (Accessed: 3 June 2020).

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J. and Williamson, R. C. (2001) ‘Estimating the support of a high-dimensional distribution’, *Neural Computation*. MIT Press PUB1010 Cambridge, MA, USA, 13(7), pp. 1443–1471. doi: 10.1162/089976601750264965.

Scitovski, R. and Sabo, K. (2020) ‘DBSCAN-like clustering method for various data densities’, *Pattern Analysis and Applications*. Springer, 23(2), pp. 541–554. doi: 10.1007/S10044-019-00809-Z/METRICS.

Sebyala, A. A., Olukemi, T. and Sacks, L. (2002) ‘Active platform security through intrusion detection using naive bayesian network for anomaly detection’, *London Communications Symposium*, pp. 1–4. Available at: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.19.9291&rep=rep1&type=pdf (Accessed: 27 June 2021).

Shalaginov, A., Banin, S., Dehghantanha, A. and Franke, K. (2018) ‘Machine learning aided static malware analysis: A survey and tutorial’, in *Advances in Information Security*. Springer New York LLC, pp. 7–45. doi: 10.1007/978-3-319-73951-9_2.

Shanableh, T. (2013) ‘Detection of frame deletion for digital video forensics’, *Digital*

Investigation. Elsevier, 10(4), pp. 350–360. doi: 10.1016/j.diin.2013.10.004.

Sharma, S., Rama Krishna, C. and Sahay, S. K. (2019) ‘Detection of advanced malware by machine learning techniques’, in *Advances in Intelligent Systems and Computing*. Springer Verlag, pp. 333–342. doi: 10.1007/978-981-13-0589-4_31.

Shin, C., Chandok, P., Liu, R., Nielson, S. J. and Leschke, T. R. (2018) ‘Potential forensic analysis of IoT data: An overview of the state-of-the-art and future possibilities’, in *Proceedings - 2017 IEEE International Conference on Internet of Things, IEEE Green Computing and Communications, IEEE Cyber, Physical and Social Computing, IEEE Smart Data, iThings-GreenCom-CPSCom-SmartData 2017*. IEEE, pp. 705–710. doi: 10.1109/iThings-GreenCom-CPSCom-SmartData.2017.182.

Silva, S. N., Reed, C. and Kennedy, E. (2016) ‘Responsibility , Autonomy and Accountability : legal liability for machine learning’, (243), pp. 1–31. Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2853462 (Accessed: 12 September 2018).

Singh, J., Millard, C., Reed, C., Cobbe, J. and Crowcroft, J. (2018) ‘Accountability in the IoT: Systems, Law, and Ways Forward’, *Computer*, 51(7), pp. 54–65. doi: 10.1109/MC.2018.3011052.

Singh, S. and Markou, M. (2004) ‘An Approach to Novelty Detection Applied to the Classification of Image Regions’, *IEEE Transactions on Knowledge and Data Engineering*, 16(4), pp. 396–407. doi: 10.1109/TKDE.2004.1269665.

Singhai, R. and Sushil, R. (2023) ‘An investigation of various security and privacy issues in Internet of Things’, *Materials Today: Proceedings*. Elsevier, 80, pp. 3393–3397. doi: 10.1016/j.matpr.2021.07.259.

Smith, R., Bivens, A., Embrechts, M., Palagiri, C. and Szymanski, B. (2002) ‘Clustering approaches for anomaly based intrusion detection’, in *Proceedings of intelligent engineering systems through artificial neural networks*. ASME, pp. 579–584.

Solberg, H. E. and Lahti, A. (2005) ‘Detection of outliers in reference distributions: Performance of horn’s algorithm’, *Clinical Chemistry*, 51(12), pp. 2326–2332. doi:

10.1373/clinchem.2005.058339.

Stage, F. K. and Manning, K. (2003) *Research in the college context: Approaches and methods*, *Research in the College Context: Approaches and Methods*. Routledge. doi: 10.4324/9780203952740.

De Stefano, C., Sansone, C. and Vento, M. (2000) 'To reject or not to reject: that is the question - an answer in case of neural classifiers', *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 30(1), pp. 84–94. doi: 10.1109/5326.827457.

Sun, P. and Chawla, S. (2004) 'On local spatial outliers', in *Proceedings - Fourth IEEE International Conference on Data Mining, ICDM 2004*, pp. 209–216. doi: 10.1109/icdm.2004.10097.

Suzuki, E., Watanabe, T., Yokoi, H. and Takabayashi, K. (2003) 'Detecting interesting exceptions from medical test data with visual summarization', in *Proceedings - IEEE International Conference on Data Mining, ICDM*, pp. 315–322. doi: 10.1109/icdm.2003.1250935.

Synnott, J., Chen, L., Nugent, C. D. and Moore, G. (2014) 'The creation of simulated activity datasets using a graphical intelligent environment simulation tool', in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC 2014*. Institute of Electrical and Electronics Engineers Inc., pp. 4143–4146. doi: 10.1109/EMBC.2014.6944536.

Synnott, J., Nugent, C. and Jeffers, P. (2015) 'Simulation of smart home activity datasets', *Sensors (Switzerland)*. MDPI AG, 15(6), pp. 14162–14179. doi: 10.3390/s150614162.

Szegedy, C., Toshev, A. and Erhan, D. (2013) 'Deep Neural Networks for object detection', in *Advances in Neural Information Processing Systems*.

Tan, P.-N. and Steinbach, M. (2006) 'Introduction to Data Mining Instructor ' s Solution Manual', *Names*, 28(1), pp. 9–35, v. Available at: http://pencilji.com/download/37729661.pdf (Accessed: 27 June 2021).

Tang, J., Chen, Z., Fu, A. W. C. and Cheung, D. W. (2002) ‘Enhancing effectiveness of Outlier detections for low Density Patterns’, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, Berlin, Heidelberg, pp. 535–548. doi: 10.1007/3-540-47887-6_53.

Tanner, A. and Dampier, D. (2009) ‘Concept mapping for digital forensic investigations’, in *IFIP Advances in Information and Communication Technology*. Springer Science and Business Media, LLC, pp. 291–300. doi: 10.1007/978-3-642-04155-6_22.

Tapia, E. M., Intille, S. S. and Larson, K. (2004) ‘Activity recognition in the home using simple and ubiquitous sensors’, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer, Berlin, Heidelberg, 3001, pp. 158–175. doi: 10.1007/978-3-540-24646-6_10.

Toppireddy, H. K. R., Saini, B. and Mahajan, G. (2018) ‘Crime Prediction & Monitoring Framework Based on Spatial Analysis’, in *Procedia Computer Science*. Elsevier, pp. 696–705. doi: 10.1016/j.procs.2018.05.075.

Underwood, M. (2016) ‘Big Data Complex Event Processing for Internet of Things Provenance: Benefits for Audit, Forensics, and Safety’, in *Cyber Assurance for the Internet of Things*. Hoboken, NJ, USA: John Wiley & Sons, Inc., pp. 209–223. doi: 10.1002/9781119193784.ch8.

Usman, N., Usman, S., Khan, F., Jan, M. A., Sajid, A., Alazab, M. and Watters, P. (2021) ‘Intelligent Dynamic Malware Detection using Machine Learning in IP Reputation for Forensics Data Analytics’, *Future Generation Computer Systems*. North-Holland, 118, pp. 124–141. doi: 10.1016/J.FUTURE.2021.01.004.

Valdes, A. and Skinner, K. (2000) ‘Adaptive, model-based monitoring for cyber attack detection’, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Springer Verlag, pp. 80–93. doi: 10.1007/3-540-39945-3_6.

Vanin, P., Newe, T., Dhirani, L. L., O’Connell, E., O’Shea, D., Lee, B. and Rao, M. (2022) ‘A Study of Network Intrusion Detection Systems Using Artificial Intelligence/Machine Learning’, *Applied Sciences (Switzerland)*. Multidisciplinary Digital Publishing Institute, p. 11752. doi: 10.3390/app122211752.

Vapnik, V. N. (2000) *The Nature of Statistical Learning Theory*, *The Nature of Statistical Learning Theory*. Springer New York. doi: 10.1007/978-1-4757-3264-1.

Varadharajan, V. and Bansal, S. (2016) ‘Data Security and Privacy in the Internet of Things (IoT) Environment’, in. Springer, Cham, pp. 261–281. doi: 10.1007/978-3-319-33124-9_11.

Venčkauskas, A., Damaševičius, R., Jusas, V., Toldinas, J., Rudzika, D. and Drėgvaitė, G. (2015) ‘A REVIEW OF CYBER-CRIME IN INTERNET OF THINGS: TECHNOLOGIES, INVESTIGATION METHODS AND DIGITAL FORENSICS’, *INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY*, 4(10), pp. 460–477. Available at: https://www.researchgate.net/publication/299105363_A_REVIEW_OF_CYBER-CRIME_IN_INTERNET_OF_THINGS_TECHNOLOGIES_INVESTIGATION_METHODS_AND_DIGITAL_FORENSICS (Accessed: 14 January 2019).

Wachter, S. (2018) ‘Normative challenges of identification in the Internet of Things: Privacy, profiling, discrimination, and the GDPR’, *Computer Law and Security Review*, 34(3), pp. 436–449. doi: 10.1016/j.clsr.2018.02.002.

Walden, I. (2013) *Accessing Data in the Cloud: The Long Arm of the Law Enforcement Agent*, *SSRN*. Springer, London. doi: 10.2139/ssrn.1781067.

Wan, X., Wang, W., Liu, J. and Tong, T. (2014) ‘Estimating the sample mean and standard deviation from the sample size, median, range and/or interquartile range’, *BMC Medical Research Methodology*. BioMed Central, 14(1), pp. 1–13. doi: 10.1186/1471-2288-14-135.

Wilkes, M. V. (1974) ‘The Art of Computer Programming, Volume 3, Sorting and Searching’, *The Computer Journal*. Oxford Academic, 17(4), pp. 324–324. doi:

10.1093/comjnl/17.4.324.

Williams, C. (2007) ‘Research Methods’, *Journal of Business & Economics Research (JBER)*, 5(3), p. 65. doi: 10.19030/JBER.V5I3.2532.

Wisdom, J. and Creswell, J. W. (2013) *Mixed Methods: Integrating Quantitative and Qualitative Data Collection and Analysis While Studying Patient-Centered Medical Home Models PCMH Research Methods Series*, *PCMH Research Methods Series*. Available at: https://pcmh.ahrq.gov/page/mixed-methods-integrating-quantitative-and-qualitative-data-collection-and-analysis-while (Accessed: 3 June 2020).

Wong, W. K., Moore, A., Cooper, G. and Wagner, M. (2003) ‘Bayesian Network Anomaly Pattern Detection for Disease Outbreaks’, in *Proceedings, Twentieth International Conference on Machine Learning*, pp. 808–815.

Xiao, J., Li, S. and Xu, Q. (2019) ‘Video-Based Evidence Analysis and Extraction in Digital Forensic Investigation’, *IEEE Access*. Institute of Electrical and Electronics Engineers Inc., 7, pp. 55432–55442. doi: 10.1109/ACCESS.2019.2913648.

Xu, X. and Lin, J. C. W. (2023) ‘Abnormal nodes sensing model in regional wireless networks based on convolutional neural network’, *Wireless Networks*. Springer, 29(7), pp. 2981–2992. doi: 10.1007/s11276-023-03255-2.

xxHash - Extremely fast hash algorithm (2015) *Github*. Available at: https://xxhash.com/ (Accessed: 9 July 2023).

Yaqoob, I., Hashem, I. A. T., Ahmed, A., Kazmi, S. M. A. and Hong, C. S. (2019) ‘Internet of things forensics: Recent advances, taxonomy, requirements, and open challenges’, *Future Generation Computer Systems*, 92, pp. 265–275. doi: S0167739X18315644.

Yue, Y., Li, S., Legg, P. and Li, F. (2021) ‘Deep Learning-Based Security Behaviour Analysis in IoT Environments: A Survey’, *Security and Communication Networks*. Hindawi Limited, 2021. doi: 10.1155/2021/8873195.

Zawoad, S. and Hasan, R. (2015) ‘FAIoT: Towards Building a Forensics Aware Eco System for the Internet of Things’, in *Proceedings - 2015 IEEE International*

Conference on Services Computing, SCC 2015, pp. 279–284. doi: 10.1109/SCC.2015.46.

Zhang, Y., Kundu, S. J., Goldgof, D. B., Sarkar, S. and Tsap, L. V. (2004) ‘Elastic face, an anatomy-based biometrics beyond visible cue’, in *Proceedings - International Conference on Pattern Recognition*, pp. 19–22. doi: 10.1109/ICPR.2004.1333969.

Zia, T., Liu, P. and Han, W. (2017) ‘Application-Specific Digital Forensics Investigative Model in Internet of Things (IoT)’, in *Proceedings of the 12th International Conference on Availability, Reliability and Security - ARES ’17*. New York, New York, USA: ACM Press, pp. 1–7. doi: 10.1145/3098954.3104052.

Ziegeldorf, J. H., Morchon, O. G. and Wehrle, K. (2014) ‘Privacy in the internet of things: Threats and challenges’, *Security and Communication Networks*. Wiley-Blackwell, 7(12), pp. 2728–2742. doi: 10.1002/sec.795.

APPENDICES

Appendix A: Isolation Forest Implementation Code

```
import pandas as pd
import numpy as np
from itertools import cycle
from sklearn.ensemble import IsolationForest
from sklearn import metrics

bankdata = pd.read_csv('D:/dataset.csv')
testing = pd.read_csv('D:/dataset.csv')

testing_data = testing.drop('timestamp', axis=1)
print ("Shape of testing_data_full>>>>>>>>>>",
testing_data.shape)
processed_data = bankdata.drop('Activity', axis=1)
processed_data = processed_data.drop('timestamp', axis=1)

num_of_samples= 524287
num_of_training_samples= 170000

processed_data = processed_data[0:num_of_samples]
print ("shape of processed data>>>>>>>>>>",
processed_data.shape)

training_data = processed_data[0:num_of_training_samples]
print ("shape of training data>>>>>>>>>>",
training_data.shape)

testing_data =
testing_data[num_of_training_samples+1:num_of_samples]
print ("shape of testing data>>>>>>>>>>", testing_data.shape)

# ## Convert pd dataframe to numpy arry - and loop round array
to select data row by row. Next step we feed each row to
encoder to compue SDR for each row

a = [] * 1
a = np.array(a, dtype= 'i4')

a = np.append(a, processed_data)
print ("shape of a data>>>>>>>>>>", a.shape)

###############DEFINE and Fit the model#########
## 'some of the parameters in IsolationForest are important
for model optimisation .. important one is contamination'
```

```
model = IsolationForest(n_estimators=100, max_samples='auto',
contamination='auto', max_features=1.0)

input = (a.reshape(-1,29))
print ("shape of input >>>>>>>>>>", input.shape)
training = input[0:num_of_training_samples]

testing = input[num_of_training_samples+1:num_of_samples]

model.fit(training)


print("=================testing for
IForest==================================")

testing_data['anomaly'] = model.predict(testing)

print ('----- model prediction --> number of 1 and -1')
print(testing_data['anomaly'].value_counts())

print ('f1 =')
print (metrics.f1_score(testing_data['Activity'],
testing_data['anomaly'], average='weighted',
labels=np.unique(testing_data['anomaly'])))
print ('Accuracy')
print (metrics.accuracy_score(testing_data['Activity'],
testing_data['anomaly'],normalize=True))
print ('Precision')
print (metrics.precision_score(testing_data['Activity'],
testing_data['anomaly'], average='weighted',
labels=np.unique(testing_data['anomaly'])))
print ('Recall')
print (metrics.recall_score(testing_data['Activity'],
testing_data['anomaly'], average='weighted',
labels=np.unique(testing_data['anomaly'])))

print("=================End of testing for
IForest==================================")
```

Appendix B: OCSVM Implementation Code

```
import pandas as pd
import numpy as np
from sklearn.svm import OneClassSVM as svm
from sklearn import metrics
```

```
bankdata = pd.read_csv('D:/dataset.csv')
testing = pd.read_csv('D:/dataset.csv')

testing_data = testing.drop('timestamp', axis=1)
print ("Shape of testing_data_full>>>>>>>>>>",
testing_data.shape)
processed_data = bankdata.drop('Activity', axis=1)
processed_data = processed_data.drop('timestamp', axis=1)

num_of_samples= 524287
num_of_training_samples= 170000

processed_data = processed_data[0:num_of_samples]
print ("shape of processed data>>>>>>>>>>",
processed_data.shape)

training_data = processed_data[0:num_of_training_samples]
print ("shape of training data>>>>>>>>>>",
training_data.shape)

testing_data =
testing_data[num_of_training_samples+1:num_of_samples]
print ("shape of testing data>>>>>>>>>>", testing_data.shape)

# ## convert pd dataframe to numpy arry - and loop round array
to select data row by row. Next step we feed each row to
encoder to compue SDR for each row

a = [] * 1
a = np.array(a, dtype= 'i4')

a = np.append(a, processed_data)
print ("shape of a data>>>>>>>>>>", a.shape)

################DEFINE and Fit the model########
## 'some of the parameters in IsolationForest are important
for model optimisation .. important one is contamination'

################divide data into training and test
sets##########

clf = svm(gamma='scale').fit(training)

testing_data['CLF_anomaly'] = clf.predict(testing)

##########Evaluation
print ("Testing for OCSVM
=========================================================")
print ('CLF_f1 =')
print (metrics.f1_score(testing_data['Activity'],
testing_data['CLF_anomaly'], average='weighted',
labels=np.unique(testing_data['CLF_anomaly'])))
print ('CLF_Accuracy')
print (metrics.accuracy_score(testing_data['Activity'],
testing_data['CLF_anomaly'],normalize=True))
print ('CLF_Precision')
print (metrics.precision_score(testing_data['Activity'],
testing_data['CLF_anomaly'], average='weighted',
labels=np.unique(testing_data['CLF_anomaly'])))
print ('CLF_Recall')
print (metrics.recall_score(testing_data['Activity'],
testing_data['CLF_anomaly'], average='weighted',
labels=np.unique(testing_data['CLF_anomaly'])))
print ("End of Testing for SVM
=========================================================")
```

Appendix C: Isolation Forest with HI-SDR Implementation Code

```
import self
import xxhash
import pandas as pd
import numpy as np
import pickle
from itertools import cycle
from sklearn.ensemble import IsolationForest
from sklearn import metrics


bankdata = pd.read_csv('D:/dataset.csv')
testing = pd.read_csv('D:/dataset.csv')

testing_data = testing.drop('timestamp', axis=1)
print ("Shape of testing_data_full>>>>>>>>>>",
testing_data.shape)
processed_data = bankdata.drop('Activity', axis=1)
processed_data = processed_data.drop('timestamp', axis=1)

num_of_samples= 524287

num_of_training_samples= 170000

processed_data = processed_data[0:num_of_samples]
print ("shape of processed data>>>>>>>>>>",
processed_data.shape)

training_data = processed_data[0:num_of_training_samples]
print ("shape of training data>>>>>>>>>>",
training_data.shape)

testing_data =
testing_data[num_of_training_samples+1:num_of_samples]
print ("shape of testing data>>>>>>>>>>", testing_data.shape)

##Build SDR##

def build_sdr(hash, n, w, partitions):
    sdr = [0] * n
    skip = int(n / partitions) # 500/5 = 100
    hash_digits = str(hash)

    if w > (skip / 10): #100/10 = 10
        # We divide by 10 because we have 10 possible values
per hash digit.

        raise ValueError('Not enough space. Please change the
parameters values.')

    for i, d in enumerate(cycle(hash_digits), start=0):
        if i == partitions:
            break

        # Calculating the relative index position (0:far left,
9:far right) in a partition
        ri = int(d) + 1

        pct = float(ri) / 10
        ri = int(round(skip * pct)) - 1

        for j in range(w):
            diff = (w - 1)
            sdr[ri + (i * skip) + j - diff] = 1

    return sdr
def xxhash32_encoder(processed_data_row):
    row = ''.join(map(str, processed_data_row))
    row = str.encode(row)
    x = xxhash.xxh32()
    x.update(row)
    hash = x.intdigest()
    return abs(hash)

#############this is weher we start to use 2 functions
created above: 1. xxhash32_encoder and 2. build_sdr

## convert pd dataframe to numpy arry - and loop round array
to select data row by row. Next step we feed each row to
```

```
encoder to compue SDR for each row

processed_data = np.array(processed_data)

a = [] * 1
a = np.array(a, dtype= 'i4')
n = 300
w = 3
p = 1
for row in processed_data:
   hash = xxhash32_encoder(row)

   sdr_data = np.array(build_sdr(hash, n=n, w=w,
partitions=p))
   #print ("SDR", sdr_data)
   a = np.append(a, sdr_data)
print ("shape of a data>>>>>>>>>>", a.shape)

################DEFINE and Fit the model#####
####### 'some of parameters in IsolationForest are important
for model optimisation .. important one is contamination'

model = IsolationForest(n_estimators=100, max_samples='auto',
contamination='auto', max_features=1.0)
sdrinput = (a.reshape(-1,n))
##############################################################
##############################################################
####################
# Convert the numpy array to a pandas DataFrame
sdr_df = pd.DataFrame(sdrinput)

# Save the DataFrame to a CSV file
sdr_df.to_csv('sdrinput.csv', index=False)
##############################################################
##############################################################
####################

print ("shape of SDR Input>>>>>>>>>>", sdrinput.shape)
sdrtraining = sdrinput[0:num_of_training_samples]
print ("Shape of sdrtraining>>>>>>>>>>", sdrtraining.shape)

model.fit(sdrtraining)

print("=================testing for SDR  +
IForest====================================")
print ("Shape of sdrinput>>>>>>>>>>", sdrinput.shape)

sdrtesting=sdrinput[num_of_training_samples +
1:num_of_samples]
```

```
print ("Shape of testing_data>>>>>>>>>>", testing_data.shape)
print ("Shape of sdrtesting>>>>>>>>>>", sdrtesting.shape)

testing_data['anomaly'] = model.predict(sdrtesting)

##############################################################
# sdrtesting is a numpy array
np.savetxt('sdrtesting.csv', sdrtesting, delimiter=',')
##############################################################
##############################################################
####################

print ('----- model prediction --> number of 1 and -1')
print(testing_data['anomaly'].value_counts())

print ('f1 =')
print (metrics.f1_score(testing_data['Activity'],
testing_data['anomaly'], average='weighted',
labels=np.unique(testing_data['anomaly'])))
print ('Accuracy')
print (metrics.accuracy_score(testing_data['Activity'],
testing_data['anomaly'],normalize=True))
print ('Precision')
print (metrics.precision_score(testing_data['Activity'],
testing_data['anomaly'], average='weighted',
labels=np.unique(testing_data['anomaly'])))
print ('Recall')
print (metrics.recall_score(testing_data['Activity'],
testing_data['anomaly'], average='weighted',
labels=np.unique(testing_data['anomaly'])))

print("=================End of testing for SDR  +
IForest==================================")
```

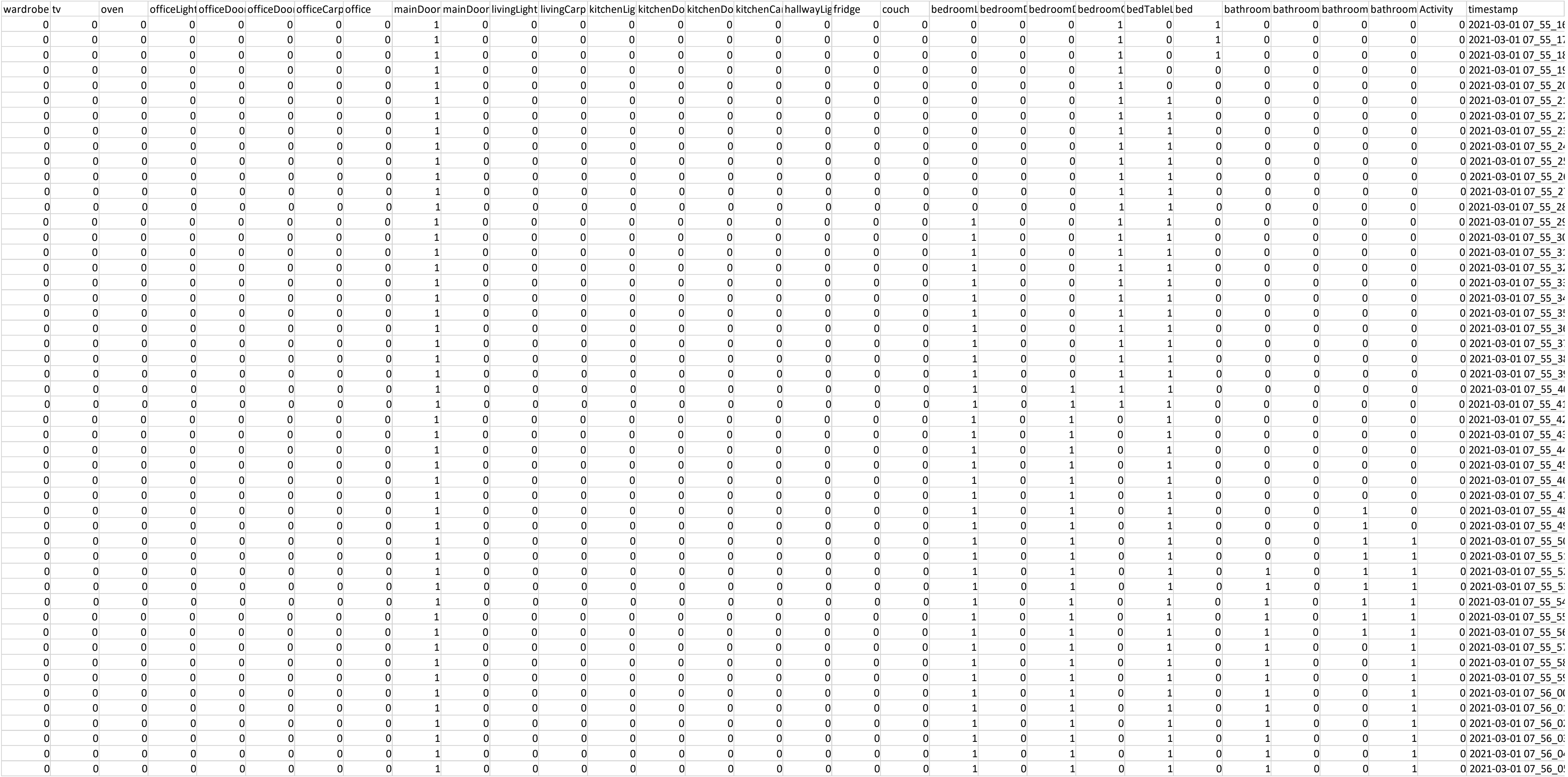
Appendix D: OCSVM with HI-SDR Implementation Code

```
import xxhash
import pandas as pd
import numpy as np
from sklearn.svm import OneClassSVM as svm
from itertools import cycle
from sklearn.ensemble import IsolationForest
from sklearn import metrics

bankdata = pd.read_csv('D:/dataset.csv')
testing = pd.read_csv('D:/dataset.csv')
```

```
testing_data = testing.drop('timestamp', axis=1)
print ("Shape of testing_data_full>>>>>>>>>>",
testing_data.shape)
processed_data = bankdata.drop('Activity', axis=1)
processed_data = processed_data.drop('timestamp', axis=1)

num_of_samples= 524287

num_of_training_samples= 170000

processed_data = processed_data[0:num_of_samples]
print ("shape of processed data>>>>>>>>>>",
processed_data.shape)

training_data = processed_data[0:num_of_training_samples]
print ("shape of training data>>>>>>>>>>",
training_data.shape)

testing_data =
testing_data[num_of_training_samples+1:num_of_samples]
print ("shape of testing data>>>>>>>>>>", testing_data.shape)

##Build SDR##

def build_sdr(hash, n, w, partitions):
    sdr = [0] * n
    skip = int(n / partitions) # 500/5 = 100
    hash_digits = str(hash)

    if w > (skip / 10): #100/10 = 10
        # We divide by 10 because we have 10 possible values
per hash digit.

        raise ValueError('Not enough space. Please change the
parameters values.')

    for i, d in enumerate(cycle(hash_digits), start=0):
        if i == partitions:
            break

        # Calculating the relative index position (0:far left,
9:far right) in a partition
        ri = int(d) + 1

        pct = float(ri) / 10
        ri = int(round(skip * pct)) - 1

        for j in range(w):
            diff = (w - 1)
            sdr[ri + (i * skip) + j - diff] = 1

    return sdr
def xxhash32_encoder(processed_data_row):
    row = ''.join(map(str, processed_data_row))
    row = str.encode(row)
    x = xxhash.xxh32()
    x.update(row)
    hash = x.intdigest()
    return abs(hash)

##############this is where we start to use 2 functions
created above: 1. xxhash32_encoder and 2. build_sdr

## convert pd dataframe to numpy arry - and loop round array
to select data row by row. Next step we feed each row to
encoder to compue SDR for each row

processed_data = np.array(processed_data)

a = [] * 1
a = np.array(a, dtype= 'i4')
n = 300
w = 3
p = 1
for row in processed_data:
   hash = xxhash32_encoder(row)

   sdr_data = np.array(build_sdr(hash, n=n, w=w,
partitions=p))
   #print ("SDR", sdr_data)
   a = np.append(a, sdr_data)
print ("shape of a data>>>>>>>>>>", a.shape)
sdrinput = (a.reshape(-1,n))

##############################################################
# Convert the numpy array to a pandas DataFrame
sdr_df = pd.DataFrame(sdrinput)

# Save the DataFrame to a CSV file
sdr_df.to_csv('sdrinput.csv', index=False)
##############################################################

print ("shape of SDR Input>>>>>>>>>>", sdrinput.shape)
sdrtraining = sdrinput[0:num_of_training_samples]
print ("Shape of sdrtraining>>>>>>>>>>", sdrtraining.shape)

sdrtesting=sdrinput[num_of_training_samples +
1:num_of_samples]
```

```
############Training the Algorithm >>> svm

clf = svm(gamma='scale').fit(sdrtraining)
##clf.predict(X)
testing_data['CLF_anomaly'] = clf.predict(sdrtesting)

##########Evaluation
print ("Testing for SDR + SVM
=========================================================")
print ('CLF_f1 =')
print (metrics.f1_score(testing_data['Activity'],
testing_data['CLF_anomaly'], average='weighted',
labels=np.unique(testing_data['CLF_anomaly'])))
print ('CLF_Accuracy')
print (metrics.accuracy_score(testing_data['Activity'],
testing_data['CLF_anomaly'],normalize=True))
print ('CLF_Precision')
print (metrics.precision_score(testing_data['Activity'],
testing_data['CLF_anomaly'], average='weighted',
labels=np.unique(testing_data['CLF_anomaly'])))
print ('CLF_Recall')
print (metrics.recall_score(testing_data['Activity'],
testing_data['CLF_anomaly'], average='weighted',
labels=np.unique(testing_data['CLF_anomaly'])))
print ("End of Testing for SDR + SVM
=========================================================")
```

Appendix E: Sample Datasets



wardrobe	tv	oven	officeLight	officeDoo	officeDoo	officeCarp	office	mainDoor	mainDoor	livingLight	livingCarp	kitchenLig	kitchenDo	kitchenDo	kitchenCa	hallwayLig	fridge	couch	bedroomL	bedroomD	bedroomD	bedroomC	bedTableL	bed	bathroom	bathroom	bathroom	bathroom	Activity	timestamp
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	2021-03-01 07_55_16
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	2021-03-01 07_55_17
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	2021-03-01 07_55_18
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	2021-03-01 07_55_19
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	2021-03-01 07_55_20
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_21
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_22
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_23
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_24
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_25
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_26
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_27
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_28
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_29
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_30
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_31
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_32
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_33
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_34
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_35
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_36
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_37
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_38
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	2021-03-01 07_55_39
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0	2021-03-01 07_55_40
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1	0	0	0	0	0	0	2021-03-01 07_55_41
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	2021-03-01 07_55_42
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	2021-03-01 07_55_43
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	2021-03-01 07_55_44
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	2021-03-01 07_55_45
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	2021-03-01 07_55_46
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	2021-03-01 07_55_47
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	0	0	2021-03-01 07_55_48
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	0	0	2021-03-01 07_55_49
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	1	0	2021-03-01 07_55_50
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	1	0	2021-03-01 07_55_51
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	1	0	2021-03-01 07_55_52
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	1	0	2021-03-01 07_55_53
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	1	0	2021-03-01 07_55_54
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	1	0	2021-03-01 07_55_55
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	1	1	0	2021-03-01 07_55_56
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	2021-03-01 07_55_57
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	2021-03-01 07_55_58
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	2021-03-01 07_55_59
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	2021-03-01 07_56_00
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	2021-03-01 07_56_01
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	2021-03-01 07_56_02
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	2021-03-01 07_56_03
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	2021-03-01 07_56_04
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0	0	1	0	2021-03-01 07_56_05