Enhancing IoT Anomaly Detection Model for Serverless Cloud Environment

MSc Research Project
Cloud Computing

Danni Zhang
Student ID: x22174435

School of Computing
National College of Ireland

Supervisor: Aqeel Kazmi

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Danni Zhang
Student ID:	x22174435
Programme:	Cloud Computing
Year:	2023
Module:	MSc Research Project
Supervisor:	Aqeel Kazmi
Submission Due Date:	14/12/2023
Project Title:	Enhancing IoT Anomaly Detection Model for Serverless Cloud Environment
Word Count:	5787
Page Count:	20

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

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

Signature:	
Date:	28th January 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

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

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

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

Enhancing IoT Anomaly Detection Model for Serverless Cloud Environment

Danni Zhang
x22174435

Abstract

The rapid proliferation of Internet of Things (IoT) devices has underscored the critical need for effective anomaly detection mechanisms within the serverless cloud environment. This study delves into exploring and implementing diverse anomaly detection models, including Decision Tree Classifier, Logistic Regression, Random Forest Classifier, Convolutional Neural Network with Long Short-Term Memory (CNN with LSTM) and Convolutional Neural Network with Bidirectional Long Short-Term Memory (CNN with BiLSTM) to identify the most effective model for this purpose. Through comprehensive experimentation and evaluation, it was found that CNN with BiLSTM outperforms other models, demonstrating an impressive accuracy of approximately 83%. The bidirectional aspect of the BiLSTM layer permits to the model to capture both past and future context, enabling a better understanding of the sequential nature of IoT device behaviour. The implications of this research are substantial, underscoring the significance of leveraging advanced deep learning architectures, particularly CNN with BiLSTM, for anomaly detection in IoT applications. The superior performance of this model suggests its potential to significantly enhance IoT security, cloud computing and reliability. The findings of this study pave the way for future research and practical implementations, propelling the domain of IoT anomaly detection forward and fostering a safer and more resilient IoT ecosystem.

1 Introduction

The fast rise of the Internet of Things (IoT) has resulted in an exponential increase in the number of linked devices, helping to create a hyper-connected society. These Internet of Things devices, which range from smart household appliances to industrial sensors, create vast amounts of data that are important for a variety of applications and decision-making processes. The foundation of a cloud system and its array of services is a cloud platform created and managed by the respective vendor (e.g., Amazon for AWS, IBM for IBM Cloud, and Google for Google Cloud). This platform assumes responsibility for delivering and overseeing a range of cloud services, governing their deployment, managing identity and access, and overseeing essential business operations such as client billing. Each vendor tailors its platform to support specific services and functionalities, aligning with their unique cloud offerings. This central platform is critical in facilitating efficient and secure cloud service provisioning, ensuring seamless business processes, and enabling clients to access and utilize the cloud services provided by the vendor (Islam et al.; 2021).

Furthermore, AWS provides AWS Lambda, a cloud-computing service that allows customers to execute code and no need to deploy or manage servers. Users are invoiced for the precise amount of constitution which gives and computes time, assessed in 100-millisecond intervals, in this serverless paradigm, with no expenses incurred during inactive times. This method maximizes cost-effectiveness by matching expenditures with the precise usage of computational resources, hence increasing the attraction and acceptance of cloud computing, specifically within the AWS ecosystem (Shankar et al.; 2020).

1.1 Background

The context for this study is the fast-growing IoT (Internet of Things) landscape and its interaction with serverless cloud settings. The Internet of Things concept incorporates networked devices that create massive amounts of data, enabling a plethora of applications across several areas. Additionally, serverless cloud computing has gained popularity due to its scalability, low cost, and event-driven processing, making it a perfect platform for IoT deployments. The proliferation of IoT devices has transformed how data is received, processed, and used, altering a wide range of industries like transportation, healthcare, smart homes and agriculture. These networked devices communicate with one another and with central cloud platforms, producing data that is critical for monitoring, analysis, and decision-making. However, this exponential development has also raised serious security issues, since the sheer quantity of devices makes them an appealing target for nefarious activity (Yousuf and Kadri; 2023).

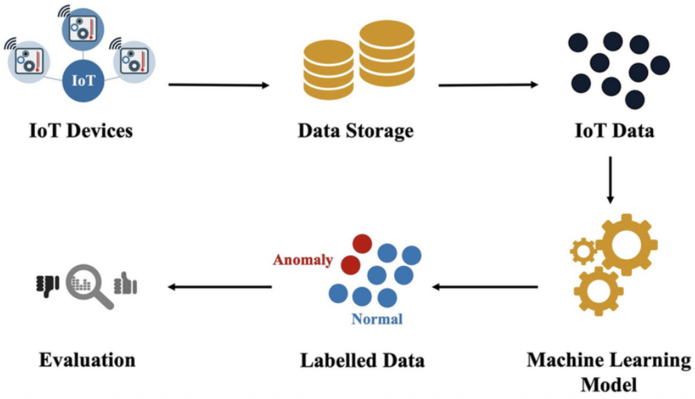


Figure 1: ML workflow for the detection of an anomaly in IoT (Alghanmi et al.; 2022)

1.2 Aim of the study

The goal of this research is to investigate and enhance anomaly detection approaches designed particularly for IoT devices in a serverless cloud environment, hence creating a more efficient and secure ecosystem for IoT applications. The report aims to examine innovative techniques and methodologies that improve anomaly detection consistency and accuracy while addressing the particular constraints given by virtualized cloud architectures and IoT device characteristics. The fundamental goal of this research is to develop and improve anomaly detection methods for IoT devices by utilizing advances in machine learning, artificial intelligence, and statistical analysis and want to increase our capacity to recognise anomalous patterns and behaviours that may indicate security risks, illegal access, or malfunctioning IoT devices by doing so. In the ambitious aim of the study, cloud computing emerges as a crucial enabler. The seamless integration of cloud computing amplifies the potential of our IoT-based anomaly detection systems, offering scalability and flexibility in handling vast data streams. By harnessing the power of the cloud, we aim to optimize resource allocation, reduce computational burden, and enhance the overall performance of our anomaly detection algorithms. This fusion aligns with the rapid advancements in cloud technologies, positioning our research at the forefront of innovation in anomaly detection within IoT ecosystems.

This report aims to create and implement anomaly detection algorithms capable of analyzing multiple characteristics such as device behaviour, network traffic, data transmission patterns, and resource utilization in a serverless cloud environment. Furthermore, the goal of this study is to contribute to current efforts to reduce security concerns associated with IoT installations in serverless cloud infrastructures. The goal is to offer effective and scalable anomaly detection approaches that allow for early identification of possible threats and prompt actions to improve the security posture of IoT systems. The study intends to decrease mistakes in anomaly detection by guaranteeing precise and fast anomaly warnings, hence optimizing the operational efficiency of IoT systems. Another goal is to assess how serverless computing affects anomaly detection in IoT devices.

1.3 Research Objectives

The study's research aims are to develop anomaly detection of IoT devices in a serverless cloud environment, extending the frontiers of efficacy, efficiency, and practicability. The methodology is comprehensive, concentrating on three main dimensions: the development of new procedures, the assessment of existing approaches, and the improvement of anomaly detection systems. To begin, the study stresses the development of novel approaches. This entails developing unique methodologies and algorithms that may detect abnormalities particular to IoT devices. These approaches should improve the accuracy and speed of anomaly detection, hence increasing the overall security of IoT implementations. Second, the research includes a critical assessment of existing methodologies. The study attempts to find gaps and possibilities for development by comprehensively examining current anomaly detection technologies, their strengths, shortcomings, and adaptation to serverless cloud systems. This evaluation will be used to refine and innovate on these strategies. The following are some research objectives given below:

1. To Develop Innovative Anomaly Detection Algorithms
2. To Enhance Accuracy and Timeliness of Anomaly Detection
3. To Optimize Resource Utilization and Scalability
4. To Evaluate Serverless Computing's Impact on Anomaly Detection
5. To Explore the Fusion of Multiple Data Sources

1.4 Research Questions

In this study, I provide a set of essential research issues in order to investigate novel ways and methodologies for advancing anomaly detection for IoT devices in a serverless cloud context. These questions cover a wide range of topics, including anomaly detection, serverless computing, and the specific issues given by IoT devices, with the goal of propelling the area ahead and introducing innovative insights and solutions.

1. How can anomaly detection techniques be tailored to suit the characteristics and constraints of IoT devices within a serverless cloud environment?
2. What novel machine learning and artificial intelligence models can be developed to enhance anomaly detection for IoT devices within a serverless cloud environment?
3. How does serverless computing affect the scalability and real-time processing capabilities necessary for anomaly detection in IoT devices?
4. What role does data fusion from diverse IoT sources play in enhancing anomaly detection accuracy and early threat identification?

2 Literature Review

This section will discuss a literature review on various algorithms of anomaly detection of Machine Learning (ML) and Deep Learning (DL) techniques for IoT.

2.1 Research Gaps

In identifying research gaps within the domain of anomaly detection for IoT devices in a serverless cloud environment, our aim is to shed light on areas that require further exploration and innovation. These gaps represent opportunities for developing novel methodologies, enhancing existing approaches, and fostering a deeper understanding of the unique challenges posed by this context. The following are some of the research gaps:

1. Integration of IoT-Specific Characteristics into Anomaly Detection
2. Optimization of Anomaly Detection for Scalability of Edge Computing
3. Real-time Anomaly Detection and Response
4. Incorporation of Edge Computing for Anomaly Detection
5. Privacy-Preserving Anomaly Detection in IoT

Identifying these research gaps provides a roadmap for future studies to focus on novel approaches, methodologies, and frameworks. Bridging these gaps will lead to the development of more efficient and effective anomaly detection systems tailored for IoT devices within serverless cloud environments, ultimately fostering a more secure and resilient IoT ecosystem.

2.2 Anomaly Detection ML Techniques for IoT

In recent years, academics have been on the edge of addressing significant global health hazards, focusing on fall detection, which is a critical concern, especially for wheelchair users. Yousuf and Kadri (2023) introduced a ground-breaking architecture that employs a novel way to monitor wheelchair falls through the use of sensor data. Their architecture, a hybrid of Isolation Forest (IF) and a threshold-based approach (TBM), greatly improved fall anomaly detection accuracy. The MPU-6050 tri-axial orthogonal accelerometer and gyroscope sensor were used to acquire the necessary data. The hybrid solution utilizing IF and threshold achieved an astounding g-mean score accuracy of roughly 97.1 percent, demonstrating the possibility of combining machine learning techniques to improve anomaly detection and decrease major health hazards connected with falls. Innovative solutions to common problems have been sprouting in the quickly evolving landscape of technologies such as System on Chip (SoC), Internet of Things (IoT), cloud computing, and artificial intelligence.

Li and Zou (2021) investigated automated anomaly detection in IoT sensor data, an important step toward improving smart home security systems. Their research smoothly linked Raspberry Pis, IoT sensors and Amazon Web Services (AWS), applying a variety of machine-learning approaches to detect anomalous instances. Despite obstacles in properly integrating numerous technologies and providing real-time anomaly detection, their study indicated significant improvement. It demonstrated the power of an automated data engineering pipeline by making major advances in anomaly detection for IoT sensor data in smart home security systems. Poojara et al. (2022) made a similar contribution to the field of IoT applications with their Puhatu Monitoring (PM) system, which was designed to monitor water level fluctuations in the Puhatu Nature Protection Area (NPA) in North West Estonia. The use of IoT devices enabled accurate environmental monitoring. Outliers in the acquired data caused challenges that were resolved using unsupervised machine learning methods, assuring data accuracy. They built separate data facilities as virtual functions using the Serverless (FaaS) concept, enabling event-driven computation on data streams. The evaluation of performance showed useful insights, notably in streamlining data processing and outlier identification, and provided a template for properly monitoring environmental factors.

Luckow et al. (2021) have shed light on the growing requirement for effective data processing throughout the edge-to-cloud spectrum in various IoT applications. Recognizing the necessity for holistic factors such as productivity, reliability, cost, and security, they suggested "Pilot-Edge" as a key metaphor for managing resources uniformly throughout this continuum. This abstraction enabled applications to enclose basic functions into high-level activities that could be effortlessly configured and deployed across the edge-to-cloud continuum. They demonstrated how Pilot-Edge expertly controlled dispersed resources, allowing applications to analyze job allocation based on a variety of parameters, setting the groundwork for enhancing IoT application performance while solving multifarious difficulties. Whereas Ramesh et al. (2022) stressed the vital importance of distribution systems in power networks and future smart grids in a similar arena. Due to a variety of circumstances, transformer failures represent a substantial danger to customers with rising dynamic service demands. The authors suggested an Internet of Things system for legitimate distribution transformer monitoring and anomaly identification, with the goal of improving reliability and lifetime. Their system demonstrated the ability to drastically cut maintenance costs and simplify predictive fault diagnosis by efficiently utilizing IoT, cloud computing, and anomaly detection methods such as Isolation Forest. This complete IoT-based solution addressed significant difficulties in transformer monitoring, offering a unique and practical way to improve distribution transformer reliability and lifetime in changing power distribution systems and smart grids.

Recent work goes extensively into solving this need by concentrating on the identification of anomalous incidences within the cloud Limprasert et al. (2022). The study investigates several approaches such as API scanning, internal system error messages, and timeouts, all of which may indicate a possible threat such as a Slowloris attack. They use machine learning-based anomaly detection methods such as Local Outlier Factor (LOF), Isolation Forest, and Elliptic Envelope. These methods are used to discover the most successful ways for real-time event identification, which make use of stream processing technologies such as Kafka and message ingression. It also emphasizes Isolation Forest's supremacy in circumstances when log messages contain previously unknown terms that require preparation via hashing whereas Belhadi et al. (2021) introduce MASAD, a revolutionary framework that builds on this achievement in anomaly detection (Multi-Agent System for Anomaly Detection).

2.3 Anomaly Detection DL Techniques for IoT

IoT is a cornerstone of innovation in the ever-changing environment of technology, linking gadgets and enabling a smooth flow of information. This connectivity, however, poses distinct difficulties that necessitate creative solutions to assure efficiency, security, and maximum performance. Singh et al. (2022) acknowledge the need to optimize computing strategies for compute-intensive and hardware IoT applications. Their study delves into a complete performance evaluation benchmark that incorporates Cloud, Fog, Edge, and Serverless computing. Jauro et al. (2020) expand on this story by underlining the limits of traditional cloud computing models. The explosion of data created by developing cloud computing architectures has driven the incorporation of deep learning algorithms capable of handling large datasets.

In a parallel endeavor, Jayaraman et al. (2021) shifted the focus to IoT-generated time-series data and the complex interrelationships among sensors and subsystems. Anomaly detection within industrial data is a computationally intensive task, especially when dealing with a high number of sensors. The authors propose leveraging Serverless Computing for parallelization, effectively addressing performance complexities and achieving a significant speed-up. This innovation in computational efficiency holds promise for anomaly detection in industrial IoT data. Meanwhile, Lu et al. (2023) shed light on the potential of edge computing, an emerging paradigm set to revolutionize computation efficiency and data processing within the IoT framework. Intrusion detection within the IoT domain is a critical concern addressed by Belhadi et al. (2023). They present a novel framework integrating deep learning and decomposition techniques, proving superior to existing approaches. Their approach holds promise for enhanced accuracy and efficacy in identifying intrusion groups, contributing significantly to IoT-based intrusion detection.

Lastly, Alotaibi and Barnawi (2023) delve into the interplay between IoT and the upcoming sixth-generation (6G) networks, emphasizing the necessity for fortified security in massive IoT networks. They advocate for innovative architectures leveraging intelligence, softwarization, and infrastructure virtualization to enhance security within the ambit of 6G. Now, let's delve into the literature concerning autoencoders. In recent years, a technological revolution has unfolded, shaping our world and industries. Researchers and innovators, like Fic et al. (2023), have been at the forefront, devising ingenious solutions to tackle real-world challenges using cutting-edge technologies. Fic et al. (2023) provide us with a glimpse into an anomaly detection system tailored for hydraulic power units, employing the power of the Internet of Things (IoT). This system, intricately designed and executed, showcases the potential of integrating IoT into the domain of hydraulic power units, shedding light on its practicality and viability within business constraints.

Similarly, Kannammal et al. (2023) delve into predictive maintenance, an indispensable process for industries, especially focusing on oil rod pumps critical in the realm of oil extraction. Their work takes us into the world of predictive modeling and anomaly detection, presenting an approach that allows preemptive actions to prevent machinery failure. By leveraging IoT and deep learning, they highlight the potential of remote predictive maintenance, promising cost and time savings for companies. Researchers such as Becker et al. (2021) realize the critical need for effective anomaly detection in the broad terrain of IoT, particularly in Edge and Fog settings, Smart Cities, and Industry 4.0. They provide Local-Optimistic Scheduling (LOS), a novel approach for offloading machine learning model training to optimise resource usage while guaranteeing closeness to data sources. Their technique demonstrates the value of decentralized and collaborative solutions, dramatically boosting resource efficiency and opening the path for effective real-time anomaly detection.

Furthermore, Aurangzaib et al. (2022) address the issue of real-time data stream processing, which is crucial in the era of widespread IoT implementation. Their study demonstrates an innovative scalable pipeline built to efficiently handle enormous data volumes. They demonstrate the promise of their technique in increasing real-time anomaly detection in IoT data by dynamically managing resources and employing containerization to improve throughput and dramatically reduce latency. Lastly, Adhikari et al. (2023) shed light on the dynamic landscape of real-time Industrial IoT applications. They emphasize the necessity of computational intelligence techniques to process data efficiently in this ever-evolving domain. Their proposed approach integrates IoT technologies and computational intelligence, highlighting the importance of energy-efficient communication and computation in IoT applications.

3 Research Methodology

3.1 Methodology

The research methodology for Anomaly Detection of IoT Devices in a Serverless Cloud Environment follows the Cross-Industry Standard Process for Data Mining (CRISP-DM) framework, a well-established and widely recognized approach for data mining projects. The workflow is shown in Figure 2. This methodology involves the following steps:

1. **Business Understanding**:
   - Begin by comprehensively understanding the business context and goals for implementing anomaly detection in IoT devices within a serverless cloud environment.
   - Define objectives, identify data sources, and understand the impact of anomaly detection on the IoT ecosystem.
2. **Data Understanding**:
   - Explore and familiarize with the available data sources.
   - Understand the structure, format, and potential variables relevant to anomaly detection.

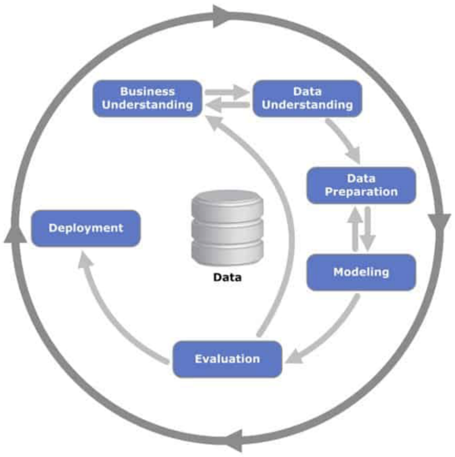


Figure 2: CRISP-DM Methodology (Jensen; 2007)

- Identify the IoT devices, their attributes, and the parameters critical for detecting anomalies.

3. **Data Preparation**:

   - Prepare the data for further analysis by addressing missing values, handling outliers, and performing necessary transformations.
   - Integrate data from various sources to create a consolidated dataset suitable for training and evaluation.
   - Cleanse and preprocess the data to enhance its quality and relevance.

4. **Modeling**:

   - Select appropriate anomaly detection algorithms suitable for IoT devices in a serverless cloud environment.
   - Experiment with various models, considering the unique characteristics of the data and the specific requirements of the application.
   - Evaluate models based on their accuracy, sensitivity, specificity, and other relevant metrics.

5. **Evaluation**:

   - Evaluate the performance of the chosen anomaly detection models using appropriate evaluation metrics.
   - Validate the models on test datasets and fine-tune parameters to achieve optimal results.
   - Compare and contrast different models to identify the most effective approach for anomaly detection.

6. **Deployment**:

   - Implement the selected anomaly detection model in a real-world serverless cloud environment.

- Integrate the model seamlessly to monitor IoT devices and detect anomalies in real time.
- Ensure that the deployment aligns with the serverless architecture and effectively utilizes cloud resources.

3.2 List of Models

A wide range of machine learning and deep learning models were used to generate successful results in the context of anomaly detection for IoT devices in a Serverless cloud environment. The study employs the following models:

Machine Learning Models:

1. **Logistic Regression:** Utilized to model the probability of a binary outcome, essential in binary classification tasks for anomaly detection in IoT device data.
2. **Decision Tree Classifier:** Employed for hierarchical decision-making, effectively identifying anomalies based on specific features within the IoT data.
3. **Random Forest Classifier:** Utilized for ensemble learning, offering robustness and accuracy in anomaly detection by aggregating outputs from multiple decision trees.

Deep Learning Models:

1. **CNN with LSTM (Convolutional Neural Network with Long Short-Term Memory):** Integrated CNN to effectively extract features from sequential IoT data, further leveraging LSTM for capturing long-term dependencies crucial in anomaly detection.
2. **CNN with BiLSTM (Convolutional Neural Network with Bidirectional Long Short-Term Memory):** Combined CNN and Bidirectional LSTM to enhance feature extraction and information flow bidirectionally, leading to improved anomaly detection capabilities for IoT device data.

These models were specifically chosen and tailored to suit the unique challenges and requirements posed by anomaly detection within the Serverless cloud environment for IoT devices. Each model's strengths and capabilities were harnessed to effectively discern anomalies and contribute to the overall security and efficiency of IoT applications.

4 Design Specification

The design specification section describes the entire architecture and technical specifications of the IoT network anomaly detection system. The system architecture is modular in nature, with numerous components combined to enable efficient and precise anomaly detection. Data preparation, model selection, and deployment methodologies designed particularly for IoT contexts are the major components. Starting with data preprocessing, the system employs robust approaches to manage a wide range of IoT network traffic. It includes approaches for data cleaning, normalization, and feature extraction that are tailored to the specific properties of IoT data streams. Additionally, in this phase, uneven class distributions are addressed using oversampling techniques such as SMOTE to provide a balanced dataset for model training.

The system implements a detailed assessment procedure of machine learning and deep learning models throughout the model selection phase. Logistic Regression, Decision Tree Classifier, Random Forest Classifier, CNN with LSTM, and CNN with BiLSTM are all evaluated. The selected CNN with BiLSTM model outperforms the competition, obtaining an accuracy of 83% because of its ability to capture both temporal and spatial correlations in IoT device activity, which is critical for anomaly identification.

Furthermore, the deployment technique is intended to aid in the detection of anomalies in IoT networks in real time. Given the resource limits inherent in IoT devices, the system takes a flexible and scalable approach. This entails optimizing model size and computational complexity in order to enable efficient inference at the edge without sacrificing detection accuracy. The design specification of the system also includes systems for continual monitoring and model retraining. It uses feedback loops to adapt to changing IoT network behaviours and new threats. This iterative method enables the system's adaptation and robustness in IoT contexts against new and emerging abnormalities.

4.1 Dataset Description

The IoT-23 dataset is a useful collection of network traffic statistics from IoT devices. It contains a total of 23 grabs, which are divided between 20 malware captures conducted in IoT devices and 3 benign IoT device traffic captures. The dataset captures ware obtained at the Stratosphere Laboratory, AIC group, FEL, CTU University, Czech Republic, and were initially introduced in January 2020. The major goal of this dataset is to provide a large and labeled archive of genuine IoT malware infections and benign traffic to the academic community, hence encouraging the development of machine learning techniques. The dataset is structured into 23 captures, referred to as scenarios, showcasing diverse IoT network traffic. These scenarios are further categorized into 20 network captures (pcap files) from infected IoT devices. This environment provides a comprehensive understanding of network behaviours and characteristics. The dataset includes detailed information about the network data and protocols found in each scenario, offering an invaluable resource for researchers and practitioners delving into IoT security and malware analysis.

4.2 Data Visualization

The data visualization section presents key insights using graphical representations. Figure 3 and Figures 4 illustrate class distribution and duration analysis, revealing data imbalance and feature relationships. Figure 6, a correlation matrix, unveils feature interdependencies. These visualizations aid in understanding data patterns, guiding subsequent analysis and decision-making processes. The count plot (Figures 3) showcases class distribution, emphasizing the need for data balancing. The funnel chart (Figure 4) provides insights into the top durations by label. The correlation matrix (Figure 6) illuminates feature relationships, essential for feature selection and model building.

Figure 3 illustrates a bar chart representing the count of each target class. The labels on the x-axis consist of "benign," "PartOfAHorizontalPortScan," "C&C," and "attack." The y-axis displays the count, ranging from 0 to 25,000. The visualization clearly in-

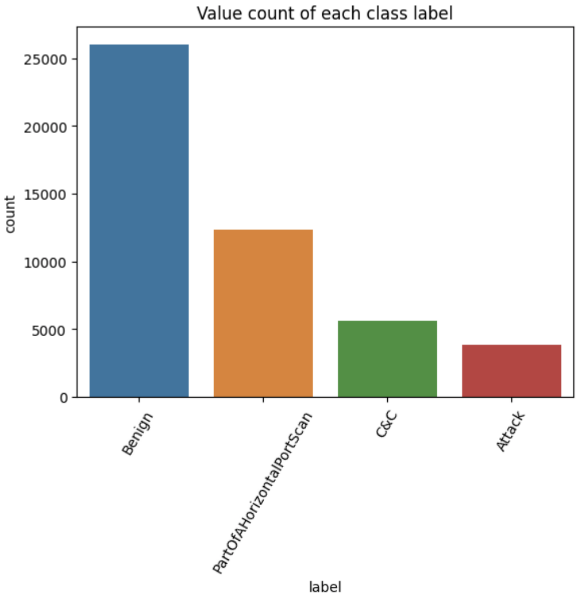


Figure 3: Count Plot of Target Class

dicates an imbalance in the data distribution among the target classes, emphasizing the need for data-balancing techniques to address this disparity effectively.

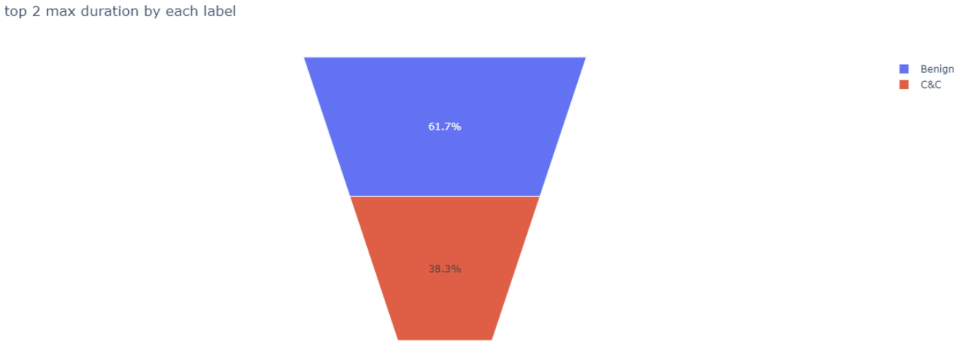


Figure 4: Top Two Maximum Durations by Label in Funnel Chart

In Figure 4, a funnel chart is depicted, showcasing the top two maximum durations for each label. The label ”benign” is represented in blue, constituting 61.7% of the data, while the label ”C&C” is highlighted in red, making up the remaining 38.3%. The funnel chart provides a visual comparison of the durations associated with the top two instances in each label category.

Figure 5 presents a correlation matrix illustrating the relationships between various features. The correlation values range from -0.5 to 1.0, signifying the strength and direction of the associations between the features. A correlation value of -0.5 indicates a negative correlation, suggesting that as one feature increases, the other tends to decrease proportionally. On the other hand, a value of 1.0 represents a perfect positive correlation, signifying that the features move in perfect harmony, increasing together.

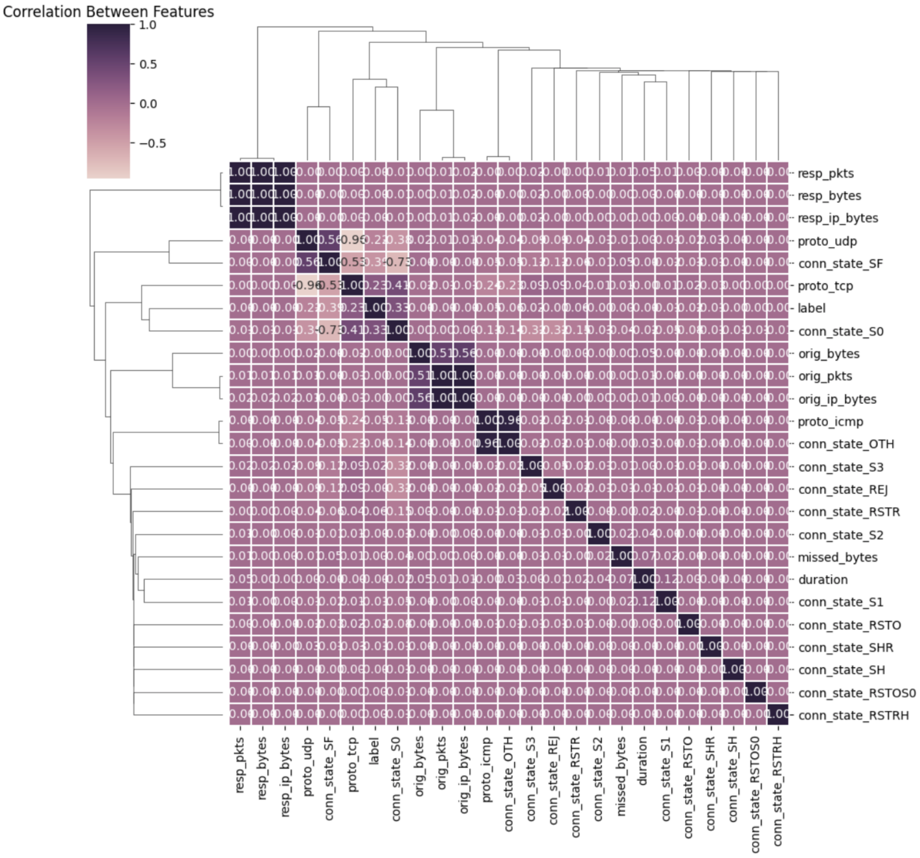


Figure 5: Correlation Matrix of Features

5 Implementation

The Implementation section is a pivotal stage in translating the conceptual framework into tangible reality, where a rich repertoire of tools and libraries converge to materialize the proposed models. In this section, the focus was on the robust implementation of the proposed solution, culminating in the final stages of this process. Python, alongside an array of essential libraries, played a fundamental role in driving the implementation forward. Leveraging the collaborative cloud-based platform Colab, we efficiently executed the implementation tasks.

The implementation primarily involved utilizing scikit-learn for machine learning tasks, employing pandas for streamlined data manipulation, and tapping into plotly, numpy, and matplotlib for generating insightful visualizations aiding in data understanding. TensorFlow and Keras, being pivotal deep learning frameworks, were instrumental in constructing and training intricate neural network models. The final stage witnessed the development and fine-tuning of these models, focusing on achieving optimal performance in anomaly detection for IoT devices within a Serverless cloud environment. A significant aspect of the final stage was transforming the raw data into a format suitable for model training and evaluation.

Data preprocessing techniques were employed to handle missing values, normalize features, and ensure the data was appropriately structured for input into the models. This transformed data formed the basis for subsequent steps, ensuring that the models were fed with quality data conducive to effective learning. Subsequently, code was written

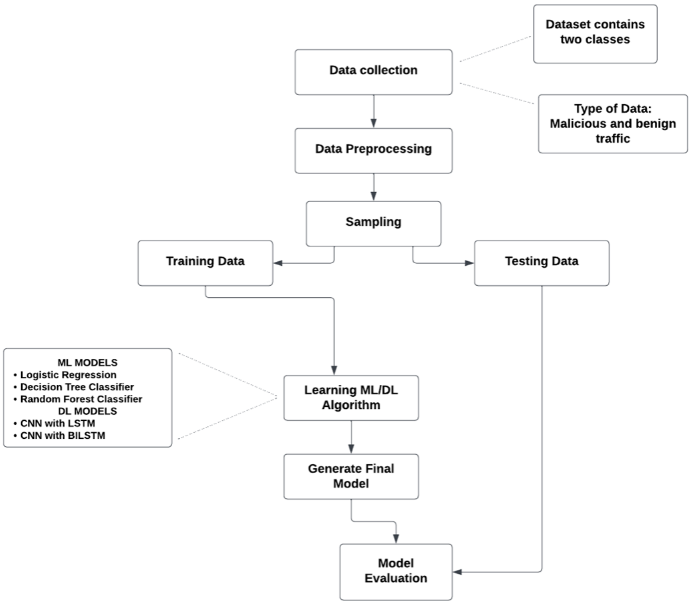


Figure 6: Workflow of the proposed system

to instantiate, configure, and train the machine learning and deep learning models.

For the machine learning models, logistic regression, decision tree classifier, and random forest classifier were implemented and fine-tuned using appropriate parameters to attain optimal results. On the other hand, for the deep learning models, CNN with LSTM and CNN with BiLSTM architectures were defined and trained to harness the power of neural networks in anomaly detection for IoT devices. The execution of the code and model training led to the creation of trained models, each tailored to its specific approach. These models were then assessed using evaluation metrics such as accuracy, precision, recall, and F1-score to gauge their effectiveness in anomaly detection. The outcome of this stage was a comprehensive understanding of how well each model performed and which approach showed the most promising results.

In summary, the final stage of implementation showcased the transformation of raw data into refined, model-ready input, the development of machine learning and deep learning models, and the evaluation of these models to discern their effectiveness in anomaly detection for IoT devices within a Serverless cloud environment. The utilization of a plethora of tools and libraries was integral in achieving this successful implementation, contributing to the advancement of IoT security in the specified context.

5.1 Logistic Regression Model

Logistic Regression is an important component in the model selection phase of this research study on IoT anomaly detection. Logistic Regression is used to create a baseline performance and to compare it to more complicated models. Despite its simplicity in comparison to deep learning architectures, Logistic Regression plays an important role in IoT network traffic monitoring by offering insights into the initial prediction power of a linear classifier.

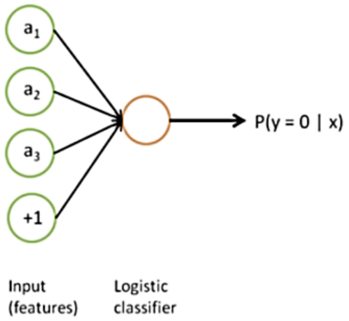


Figure 7: Logistic Regression Architecture

5.2 Decision Tree Classifier Model

Within the IoT network traffic dataset, the Decision Tree Classifier supports in the exploration of non-linear correlations and feature significance.

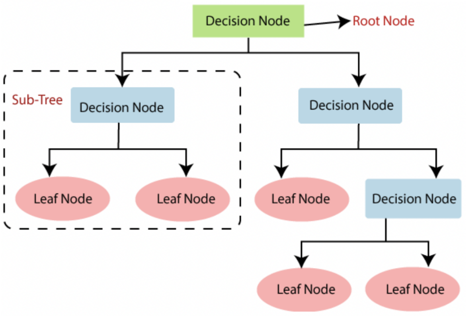


Figure 8: Decision Tree Classifier Architecture (Deshpande et al.; 2021)

5.3 Random Forest Classifier Model

The Random Forest Classifier is an ensemble approach that combines numerous decision trees to improve prediction capacity while retaining interpretability. It uses an ensemble of trees to capture complicated relationships inside the IoT network traffic information, providing a more robust method than single Decision Trees.

5.4 CNN with LSTM

The CNN with LSTM model is an advanced deep learning architecture designed for sequential data processing in IoT network traffic. Its integration entails using convolutional layers for spatial feature extraction and LSTM layers for temporal relationships, allowing it to understand the sequential nature of IoT device activity. Figure 10 presents the architecture of CNN with BiLSTM.

5.5 CNN with BiLSTM

The CNN with BiLSTM emerged as the most effective deep learning architecture. This model stands out for its ability to capture both past and future context, exploiting

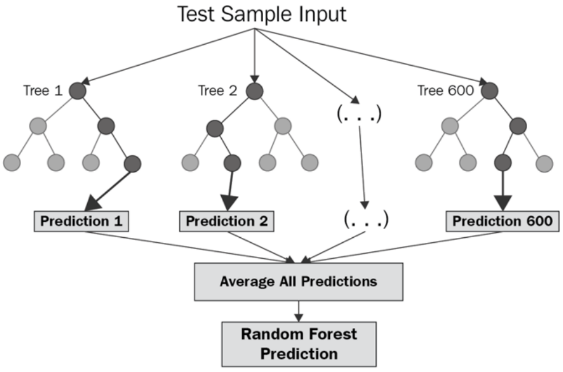


Figure 9: Random Forest Classifier (Ijmtst; 2022)

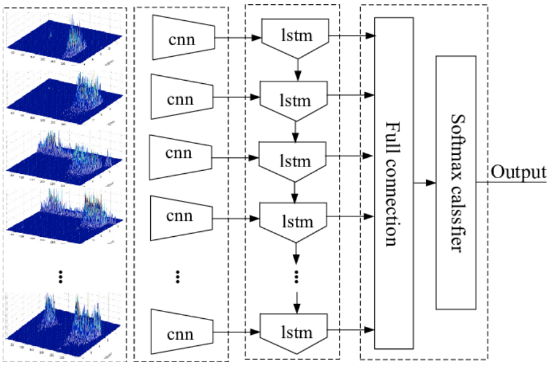


Figure 10: CNN with LSTM architecture (Zhou et al.; 2019)

the BiLSTM layer's bidirectional flow of information to fully comprehend the sequential nature of IoT device activity.

6 Evaluation

6.1 Classification Performance of Machine Learning Models

In the evaluation of this study, three key classifiers were employed: Logistic Regression, Decision Tree Classifier, and Random Forest Classifier. Figure 12 present their ROC curve for class 1 (benign).

In terms of accuracy, Logistic Regression achieved an accuracy score of approximately 0.53, indicating a moderate level of accuracy in classifying the data. The Decision Tree Classifier demonstrated an accuracy of about 0.69, which exceeds Logistic Regression's baseline accuracy, the Decision Tree Classifier provides insights into how various factors contribute to identifying abnormalities in IoT network activity. And the Random Forest Classifier exhibited an accuracy score of around 0.61, which exceeds the accuracy of Logistic Regression while falling short of the performance of the Decision Tree Classifier, the Random Forest Classifier excels in capturing feature relevance and interactions within the data.

Table 1 presents the comparison of classification metrics for anomaly detection models

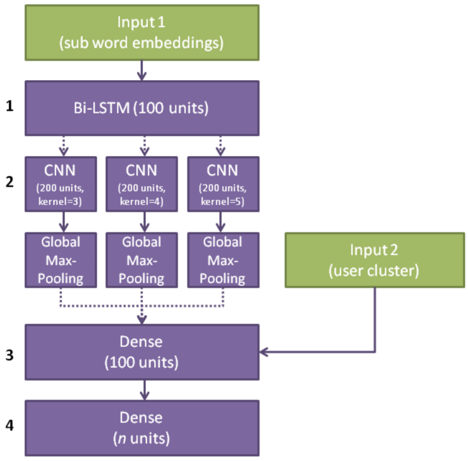
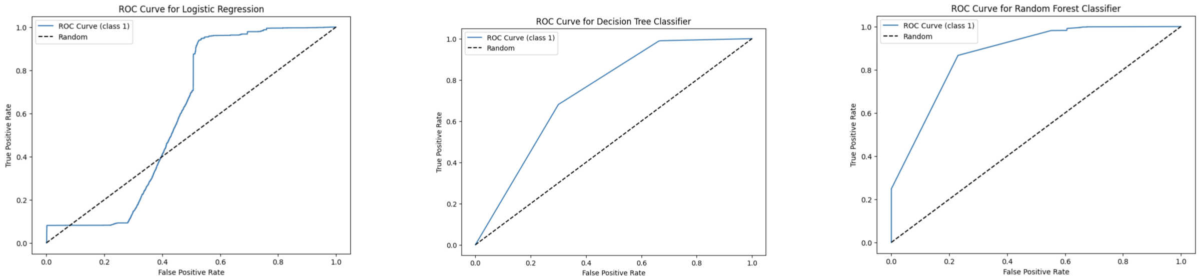


Figure 11: CNN with BiLSTM architecture (Wiedemann et al.; 2018)



(a) Logistic Regression

(b) Decision Tree Classifier

(c) Random Forest Classifier

Figure 12: ROC Curves for ML models

in IoT devices.

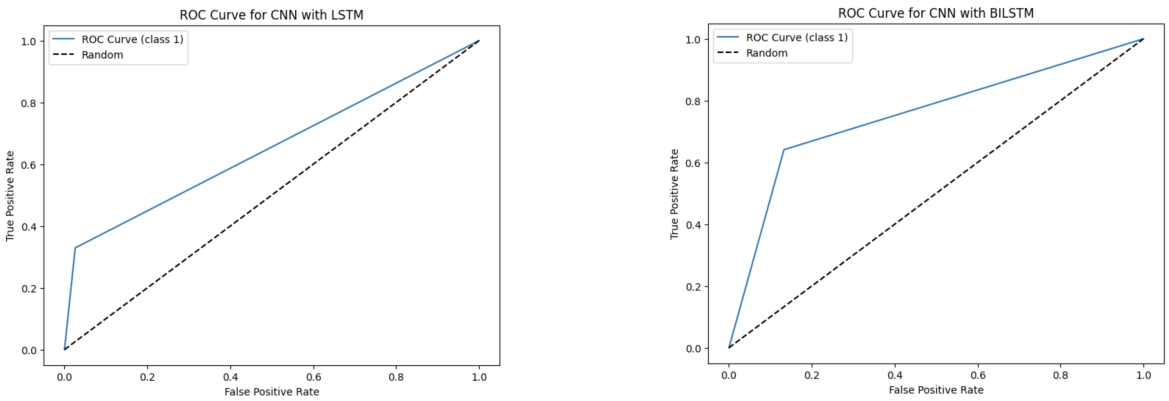
Table 1: Model Performance Metrics

Model	Accuracy
Logistic Regression	0.53
Decision Tree Classifier	0.69
Random Forest Classifier	0.61

6.2 Classification Performance of DL Models

In evaluating the performance of deep learning models for anomaly detection in IoT devices within a Serverless cloud environment, two prominent architectures were employed: CNN with LSTM and CNN with BiLSTM. Figure 13 presents their ROC curve for class 1 (benign).

The CNN with LSTM model achieved an accuracy of approximately 0.66, demonstrating a strong precision for class 0 (1.00) but comparatively lower recall for classes 1, 2, and 3. Although it achieves an accuracy of 0.66, exhibiting competency in learning temporal patterns, it falls short of the CNN with BiLSTM model. The use of the CNN with LSTM model aids in understanding the significance of temporal relationships



(a) CNN with LSTM

(b) CNN with BILSTM

Figure 13: ROC Curves for DL models

within IoT network traffic, laying the groundwork for comparisons with more advanced architectures such as CNN with BiLSTM, which ultimately outperformed it in accurately capturing the complexities of IoT device behaviour for effective anomaly detection in this research.

On the other hand, the CNN with BiLSTM model showcased an improved accuracy of approximately 0.83, excelling in precision and recall for various classes. Specifically, it demonstrated exceptional recall for class 2 (0.98) and high precision for class 3 (0.98), contributing to an overall higher F1 score. The improved performance of the CNN with BiLSTM indicates its ability to grasp complicated temporal patterns in IoT network traffic, demonstrating a stronger knowledge of sequential data.

In direct comparison, the CNN with BiLSTM outperformed the CNN with LSTM with a notable accuracy advantage of 17%. These results emphasize the significance of employing advanced deep learning models, particularly CNN with BiLSTM, in IoT anomaly detection, with the CNN with BiLSTM model being the most effective in this context.

Table 2 presents the comparison of classification metrics for CNN with LSTM and CNN with BiLSTM models in anomaly detection for IoT devices.

Table 2: Classification Metrics

Model	**Accuracy**
CNN with LSTM	0.66
CNN with BiLSTM	0.83

6.3 Discussion

The discussion revolves around a comprehensive analysis of the findings obtained from all experiments and results, encompassing an in-depth exploration of the models' performances, their strengths, weaknesses, and potential implications for anomaly detection in IoT devices within a Serverless cloud environment. Firstly, the Logistic Regression model displayed moderate accuracy, achieving approximately 53%. The precision, recall, and F1 scores for class 1 were notably imbalanced, indicating a challenge in identifying anomalies. This model's simplicity and efficiency make it a viable initial approach, but its limited ability to capture complex patterns might hinder its performance in intricate IoT anomaly detection scenarios. The Decision Tree Classifier, despite achieving a higher accuracy of around 69%, exhibited imbalanced precision and recall for class 1, highlighting a challenge in identifying anomalies accurately. However, its interpretability and ease of understanding the decision-making process make it valuable for initial insights into anomaly detection. The Random Forest Classifier, with an accuracy of about 61%, showcased a balanced precision-recall trade-off for class 1. This ensemble model outperformed the single Decision Tree Classifier, indicating the advantage of combining multiple decision trees for improved accuracy and precision in anomaly detection.

The CNN with LSTM model demonstrated an accuracy of approximately 66%, indicating its effectiveness in capturing spatial features within IoT data. However, the precision and recall for class 1 were imbalanced, emphasizing the model's struggle to identify anomalies effectively. In contrast, the CNN with BiLSTM model achieved a significantly higher accuracy of around 83%, displaying improved precision and recall for class 1. The bidirectional aspect of BiLSTM enhanced the model's ability to capture intricate temporal patterns, contributing to its superior performance. Comparing the two deep learning models, CNN with BiLSTM emerged as the most effective, surpassing CNN with LSTM in accuracy by 17%.

7 Conclusion and Future Work

In conclusion, this study aimed to address the critical issue of anomaly detection in IoT devices. The research question revolved around identifying the most effective model for this purpose. The objectives included exploring and implementing various models, assessing their performance, and providing valuable insights to enhance anomaly detection.

The research successfully achieved its objectives by comprehensively examining and implementing Logistic Regression, Decision Tree Classifier, Random Forest Classifier, CNN with LSTM, and CNN with BiLSTM models. The key findings revealed that while traditional machine learning models demonstrated moderate performance, deep learning models, particularly CNN with BiLSTM, displayed significantly superior accuracy and efficiency in identifying anomalies in IoT data. The implications of this research are substantial, highlighting the importance of leveraging advanced deep learning architectures for anomaly detection in IoT applications. The superior performance of CNN with BiLSTM emphasizes its potential for enhancing IoT security and reliability. However, limitations were observed, notably in the imbalanced datasets affecting model performance and the need for significant computational resources for training complex models.

In terms of future work, focusing on addressing the data imbalance issue through advanced sampling techniques or novel loss functions can significantly improve model performance. Additionally, incorporating more real-world data and conducting experiments in diverse IoT environments will enhance the generalizability of the models. Exploring edge AI applications and optimizing models for resource-constrained IoT devices is another promising avenue for future research, contributing to the broader domain of IoT anomaly detection.

Finally, exploring potential commercialization opportunities, integrating the proposed models into real-time IoT applications, and offering them as robust anomaly detection solutions in the market would be a meaningful step toward practical implementation and societal impact.

References

Adhikari, M., Menon, V. G., Rawat, D. B. and Li, X. (2023). Guest editorial introduction to the special section on computational intelligence and advanced learning for next-generation industrial iot, *IEEE Transactions on Network Science and Engineering* **10**(5): 2740–2744.

Alghanmi, N., Alotaibi, R. and Buhari, S. M. (2022). Machine learning approaches for anomaly detection in iot: an overview and future research directions, *Wireless Personal Communications* **122**(3): 2309–2324.

Alotaibi, A. and Barnawi, A. (2023). Securing massive iot in 6g: Recent solutions, architectures, future directions, *Internet of Things* p. 100715.

Aurangzaib, R., Iqbal, W., Abdullah, M., Bukhari, F., Ullah, F. and Erradi, A. (2022). Scalable containerized pipeline for real-time big data analytics, *2022 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*, IEEE, pp. 25–32.

Becker, S., Schmidt, F., Thamsen, L., Ferrer, A. J. and Kao, O. (2021). Los: Local-optimistic scheduling of periodic model training for anomaly detection on sensor data streams in meshed edge networks, *2021 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*, IEEE, pp. 41–50.

Belhadi, A., Djenouri, Y., Djenouri, D., Srivastava, G. and Lin, J. C.-W. (2023). Group intrusion detection in the internet of things using a hybrid recurrent neural network, *Cluster Computing* **26**(2): 1147–1158.

Belhadi, A., Djenouri, Y., Srivastava, G. and Lin, J. C.-W. (2021). Reinforcement learning multi-agent system for faults diagnosis of mircoservices in industrial settings, *Computer Communications* **177**: 213–219.

Deshpande, N. M., Gite, S. and Aluvalu, R. (2021). A review of microscopic analysis of blood cells for disease detection with ai perspective, *PeerJ Computer Science* **7**: e460.

Fic, P., Czornik, A. and Rosikowski, P. (2023). Anomaly detection for hydraulic power units—a case study, *Future Internet* **15**(6): 206.

Ijmtst, E. (2022). Air pollution control using data mining, *International Journal for Modern Trends in Science and Technology* **8**: 303–312.

Islam, M. S., Pourmajidi, W., Zhang, L., Steinbacher, J., Erwin, T. and Miranskyy, A. (2021). Anomaly detection in a large-scale cloud platform, *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, IEEE, pp. 150–159.

Jauro, F., Chiroma, H., Gital, A. Y., Almutairi, M., Shafi'i, M. A. and Abawajy, J. H. (2020). Deep learning architectures in emerging cloud computing architectures: Recent development, challenges and next research trend, *Applied Soft Computing* **96**: 106582.

Jayaraman, S., Reddy, C., Khabiri, E., Patel, D., Bhamidipaty, A. and Kalagnanam, J. (2021). Asset modeling using serverless computing, *2021 IEEE International Conference on Big Data (Big Data)*, IEEE, pp. 4084–4090.

Jensen, K. A. (2007). CRISP-DM Process Diagram. Accessed: 2023-12-12.
URL: *https://upload.wikimedia.org/wikipedia/commons/b/b9/CRISP-DM$_P$rocess$_D$iagram.png*

Kannammal, A., Guhanesvar, M. and Venketesz, R. (2023). Predictive maintenance for remote field iot devices—a deep learning and cloud-based approach, *Mobile Computing and Sustainable Informatics: Proceedings of ICMCSI 2023*, Springer, pp. 567–585.

Li, X. and Zou, B. (2021). An automated data engineering pipeline for anomaly detection of iot sensor data, *arXiv preprint arXiv:2109.13828* .

Limprasert, W., Jantana, P. and Liangsiri, A. (2022). Anomaly detection on real-time security log using stream processing, *2022 17th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, IEEE, pp. 1–6.

Lu, S., Lu, J., An, K., Wang, X. and He, Q. (2023). Edge computing on iot for machine signal processing and fault diagnosis: A review, *IEEE Internet of Things Journal* .

Luckow, A., Rattan, K. and Jha, S. (2021). Pilot-edge: Distributed resource management along the edge-to-cloud continuum, *2021 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, IEEE, pp. 874–878.

Poojara, S., Jõeleht, A., Jakovits, P. and Srirama, S. N. (2022). Serverless outlier management for environmental iot data-a case study of puhatumonitoring, *2022 IEEE 8th World Forum on Internet of Things (WF-IoT)*, IEEE, pp. 1–7.

Ramesh, J., Shahriar, S., Al-Ali, A., Osman, A. and Shaaban, M. F. (2022). Machine learning approach for smart distribution transformers load monitoring and management system, *Energies* **15**(21): 7981.

Shankar, K., Wang, P., Xu, R., Mahgoub, A. and Chaterji, S. (2020). Janus: Benchmarking commercial and open-source cloud and edge platforms for object and anomaly detection workloads, *2020 IEEE 13th International Conference on Cloud Computing (CLOUD)*, IEEE, pp. 590–599.

Singh, P., Kaur, A. and Gill, S. S. (2022). Machine learning for cloud, fog, edge and serverless computing environments: comparisons, performance evaluation benchmark and future directions, *International Journal of Grid and Utility Computing* **13**(4): 447–457.

Wiedemann, G., Ruppert, E., Jindal, R. and Biemann, C. (2018). Transfer learning from lda to bilstm-cnn for offensive language detection in twitter.

Yousuf, S. and Kadri, M. B. (2023). A ubiquitous architecture for wheelchair fall anomaly detection using low-cost embedded sensors and isolation forest algorithm, *Computers and Electrical Engineering* **105**: 108518.

Zhou, X., Wu, X., Ding, P., Li, X., He, N., Zhang, G. and Zhang, X. (2019). Research on transformer partial discharge uhf pattern recognition based on cnn-lstm, *Energies* **13**(1): 61.