Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 16, Issue. 1: 2025
ISSN : **1906-9685**

EXPLORING MACHINE LEARNING TECHNIQUES FOR IDENTIFYING ANOMALIES IN MEDICAL DEVICES

P Renuka[1], Dr. Nilesh Parihar[2]

[1]*PhD Scholar, Dept. of EC, Gandhinagar University, Gujarat, India*

[2]*Professor, Dept. of EC, Gandhinagar University, Gujarat, India*

Abstract: *With the increasing deployment of Internet of Things (IoT) devices in the medical field, ensuring the reliability and security of medical devices through anomaly detection of sensor data is becoming more important than ever. The machine learning-based anomaly detection can play a crucial role in medical IoT sensor data by identifying sudden deviations from expected patterns, such as device malfunctions, anomalous patient conditions, and cybersecurity threats. In medical applications, anomaly detection can help identify unusual activities or behaviors, such as device failures or abnormal patient vitals, enhancing patient safety and care quality. Existing methods for anomaly detection in medical IoT sensor data often rely on simple threshold-based techniques or manual rule definitions, which may be ineffective in capturing complex and evolving patterns in sensor readings. These methods may struggle to differentiate between normal variations and genuine anomalies, leading to false positives or missed detection. Moreover, traditional approaches may lack scalability and adaptability to diverse medical environments and sensor modalities, hindering their effectiveness in real-world healthcare applications. Additionally, manual rule definition and parameter tuning may require significant expertise and effort, limiting the practicality of these methods for large-scale medical IoT deployments. The proposed system utilizes machine learning techniques to automate and enhance anomaly detection in medical IoT sensor data, addressing the limitations of existing methods. This work employs supervised learning algorithms to detect anomalies in sensor readings without the need for labeled training data. By analyzing temporal and spatial patterns in sensor data, our models can identify deviations from normal behavior and flag potential anomalies.*

Keywords: *IOT sensor data, Ml, XGboost algorithm, abnormal data*

I. INTRODUCTION

The proliferation of Internet of Things (IoT) devices has revolutionized various domains, ranging from smart homes to industrial manufacturing. These IoT devices generate vast amounts of sensor data, providing valuable insights into the physical world. However, ensuring the reliability and security of IoT systems is paramount, as they are vulnerable to various threats and anomalies. Traditionally, anomaly detection in sensor data has relied on simple threshold-based techniques or manual rule definitions. These methods, while initially effective, struggle to cope with the complexity and evolving nature of IoT environments. Moreover, the scalability and adaptability of these approaches are often limited, hindering their effectiveness in real-world applications. In recent years, the application of machine learning techniques to anomaly detection in IoT sensor data has gained traction. By leveraging the power of supervised learning algorithms, researchers have been able to automate and enhance anomaly detection processes. These machine learning models analyze temporal and spatial patterns in sensor data, enabling them to identify deviations from normal behavior with high accuracy. Despite recent advancements, anomaly detection in IoT sensor data remains a challenging task. Existing methods often suffer from high false positive rates or missed detections, leading to decreased reliability and security of IoT systems. Additionally, manual rule definition and parameter tuning require significant expertise and effort, limiting the scalability of these approaches for large-scale deployments.

The increasing prevalence of IoT devices across various domains underscores the need for more robust and scalable anomaly detection techniques. By improving the reliability and security of IoT systems, researchers can unlock the full potential of these technologies, leading to safer homes, more efficient manufacturing processes, and enhanced transportation systems.

Existing Systems

Existing anomaly detection systems in IoT sensor data predominantly rely on threshold-based techniques or manual rule definitions. While these methods may be suitable for simple scenarios, they often struggle to cope with the complexity and variability of real-world IoT environments. Moreover, these approaches require constant manual intervention and tuning, making them impractical for large-scale deployments.

Research Objective

The primary objective of this research is to develop a machine learning-based anomaly detection system for IoT sensor data that addresses the limitations of existing methods. By leveraging supervised learning algorithms, our goal is to automate and enhance the anomaly detection process, improving the reliability and security of IoT systems. The need for more robust and scalable anomaly detection techniques in IoT sensor data is evident across various domains. From smart homes to industrial manufacturing, the ability to identify and mitigate anomalies is crucial for ensuring the smooth operation of IoT systems. By developing advanced machine learning models, researchers can meet this need and pave the way for more secure and efficient IoT deployments.

II. LITERATURE SURVEY

An authentication has been proposed for IoT [1] using two algorithmic models that ensure valid authentication. The security solution proposed in this work has a limited scope, as it protects only the lightweight sensor devices from the standard network layer and physical layer-based attacks. ElKafrawy et al4100736 [2] presented a secured communications model among IoT nodes and a cluster-based fuzzy architecture. This works has seen good mitigation against malicious nodes but not against audit attack surface. It does not sufficiently explore the performance analysis of the operational communication and computational costs.

Chen et al. [3] proposed a unique Low scale Denial of-Service attack detection technique that uses trust evaluation along with Hilbert-Huang Transformation in Zigbee WSN to enhance the security risks handling in one unified solution keeping in mind the large number of low energy devices which is vulnerable to attacks. The signal and anomaly detection technique presented in this work assists in minimizing the attack level. Due to support for cloud & edge computing, it comes with an extensible design, and yet, has still the challenge of higher storage overheads. The Intrusion Detection System (IDS) is relied upon to detect and monitor threat behaviors in the network in classical network security. Therefore, these models do not specifically focus on the IoT setting

Van et al. Proposed deep learning algorithms that realize network intrusion detection including [4]. In their paper they mention two types of deep learning models the Restricted Boltzmann Machines (RBM) and Autoencoder (AE). This group of authors developed a stacked RBM and AE as two types of Deep Belief Network (DBN) architectures and provided their results on an intrusion detection task. In the stacked RBM case (which acts like a probability distribution), Assuming that the hidden layer of each RBM to be input layer of next RBM in the stack. Stacked AE can extract features of unlabelled data on the network by unsupervised learning, which solves the problem of the large number of unlabelled data.

Almiani et al. A deep learning based IoT environment [5] intrusion detection system is proposed. Their model includes two main modules: the traffic analysis engine and the classification engine. Traffic analysis engine: this component is used to pre-process the traffic data (i.e., symbolic-to-numeric transformation, feature reduction and normalization) Then, the classified data would be inserted into the classification engine, where two deep recurrent neural networks (RNN) are utilized to react quickly in a real-time setting [88]. The two RNN serve as two attack detection filters. For a new data point that is classified as normal by the first RNN layer, it will be fed into the second RNN detection layer to determine if it is anomalous. Two layers of RNN share the same dataset for training. The only distinction is that the training context of the first RNN includes typical and aberrant data, whilst the training context of the second RNN only includes typical traffic data.

Et al. [6] was used for an anomaly detection for IoT traffic called Vector Convolutional Deep Learning (VCDL) model that applies a novel distributed intelligence called “fog computing”. The models proposed have 3 layers of components. The first layer is a distributed IoT device. The fog layer is the second layer—several work fog nodes, connected to the IoT devices, and train each VCDL model in the distribution. The best set of parameters will be sent to the worker nodes by the master fog node in the fog layer. So the traffic data will be given on the respective worker node to categorize it as normal or attack. The classification result will be transmitted to cloud layer, which is the third layer of the framework proposed in this paper. The FOG layer is the one responsible to validate data using the entire cloud layer. The experimental outcome shows that the high accuracy of anomaly traffic detection with detection time shorter than centralized detection model and can be achieved through the proposed distributed VCDL framework. Every IoT system is associated with real-time data or time series data. Thus, various studies concentrate on identifying anomalies from time series data obtained using IoT devices

Liu et al. 7, which aims at regrinding the temperature data collected by its distributed indoor climate control system. Anomaly detection are of two types point anomaly: signifies one outlier value which is significantly different from other data and, contextual anomaly: a sequence of inappropriate data point. Liu et al. a neural network-based model was proposed to detect these two

types of anomalies that hybridized the auto encoder (AE) and the long short-term memory (LSTM) model [7]. So, the structure of AE is like that of the standard Feed Forward Neural Network with fewer hidden layers neurons, thus, making it possible for AE's output to be very close to its corresponding input. LSTM model is capable of extracting features from sequential data and recognizing the relationship of neighbouring input data [7]. Thus, for the different outlier groups, the AE segment of the proposed model focuses on the point anomaly and the LSTM part focuses on the contextual anomaly. Fused neural network models with the anomaly detection algorithm to improve accuracy of the model

Sun et al. [8] built a system for UAV fault detection using a hierarchical fault cause structure map. They focused on developing a comprehensive knowledge base to facilitate effective anomaly detection.

Liu et al. [9] expanded upon this by studying fault detection algorithms specifically for UAV control systems. They proposed utilizing parameter estimation techniques coupled with noise estimation to diagnose faults accurately.

III. PROPOSED SYSTEM

The research begins by acquiring a dataset sourced from IoT sensors (Step 1). This dataset forms the foundation of the study, containing vital information collected by these sensors across different domains. Before delving into analysis, it's essential to pre-process the dataset to ensure its quality and compatibility with the chosen analytical techniques (Step 2). This includes handling missing values and encoding categorical variables to numeric form for machine learning algorithms to process.

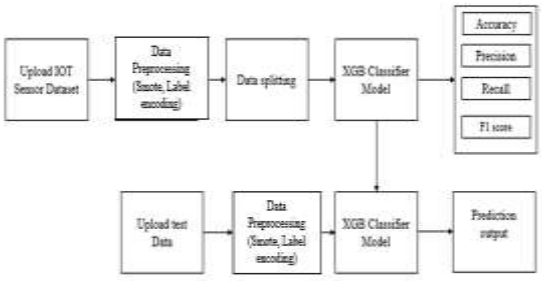

Fig.1. Block Diagram

Given the inherent class imbalance common in anomaly detection scenarios, the dataset undergoes SMOTE (Synthetic Minority Over-sampling Technique) for data balancing (Step 3). SMOTE generates synthetic samples for the minority class, thereby rectifying the imbalance issue and preventing bias in subsequent analyses. Moving to model implementation, the research compares the performance of two algorithms: Logistic Regression and XGBoost Classifier (Steps 5 and 6). Logistic Regression, a traditional yet robust method, serves as a baseline for comparison, while the XGBoost Classifier, known for its effectiveness in handling complex datasets, represents an advanced approach.

Finally, performance evaluation in step 7 checks for metrics like accuracy, precision, recall, F1-score, etc., to assess the effectiveness of all the algorithms in detecting anomalies. This stage reveals how well the models generalize to unseen data and how accurately they detect anomalies, as well as how to reduce false positives and false negatives.

In the end, this research predicts some outputs based on a test dataset with a trained model (Step 8), which in this case is the XGBoost Classifier. In this step the developed anomaly detection system is used to classify an unseen instance by putting it into the appropriate class based on the learnt pattern during the training step.

This approach is a systematic and detailed research procedure that provides a guideline from anomaly detection in IoT sensors data including pre-processing, features, training, evaluation and execution so that it promotes onto advanced anomaly detection techniques demonstrated for IoT systems.

IV. IMPLEMENTATION

In implementing machine learning-based anomaly detection in IoT sensor data, a multifaceted approach is essential to address the diverse challenges posed by the dynamic nature of sensor readings and the evolving patterns of anomalies. Firstly, data preprocessing plays a pivotal role in ensuring the quality and relevance of input data. Techniques such as data cleaning, normalization, and feature engineering may be employed to mitigate noise, handle missing values, and extract meaningful features from raw sensor data. This preprocessing phase lays the foundation for effective anomaly detection by enhancing the discriminative power of the input features.

Once the data is preprocessed, the selection of appropriate machine learning algorithms becomes paramount. You might leverage supervised, unsupervised, or semi-supervised learning techniques based on the availability of labeled data and the complexity of the anomaly patterns. For instance, unsupervised learning algorithms like Isolation Forests or Gaussian Mixture Models can identify anomalies without the need for labeled data, making them suitable for detecting novel and unforeseen anomalies. Conversely, supervised learning algorithms like Support Vector Machines or Random Forests may be employed when labeled data is abundant, allowing for the precise classification of anomalies based on predefined categories.

Furthermore, the implementation of ensemble learning techniques, where multiple models are combined to improve predictive performance, might enhance the robustness and generalization capability of the anomaly detection system. Ensemble methods such as bagging, boosting, or stacking enable the aggregation of diverse models, thereby capturing different aspects of the

complex underlying data distribution and reducing the risk of overfitting.

Additionally, you may integrate advanced anomaly scoring mechanisms, such as Mahalanobis distance or autoencoders, to quantify the deviation of sensor readings from normal behavior accurately. These scoring techniques enable the generation of anomaly scores for each data point, facilitating the prioritization of anomalies based on their severity and potential impact on system operations.

In conclusion, the implementation of machine learning-based anomaly detection in IoT sensor data involves a comprehensive approach encompassing data preprocessing, algorithm selection, ensemble learning, and advanced anomaly scoring mechanisms.

Description

Anomaly detection in IoT sensor data, a thorough understanding of the problem domain and the intricacies of sensor technology is crucial for devising effective anomaly detection solutions. Description information encompasses various aspects, including the types of sensors deployed, the characteristics of the data they produce, and the specific anomalies targeted for detection. For instance, in a smart home environment, diverse sensors such as motion sensors, temperature sensors, and door sensors may be utilized to monitor occupancy, environmental conditions, and security breaches. Each sensor type generates distinct data modalities, such as time-series data for temperature sensors and binary data for door sensors, posing unique challenges for anomaly detection.

The description information should encompass the contextual factors influencing sensor data and anomaly patterns. Factors such as temporal dynamics, spatial correlations, and seasonal variations may significantly impact the behavior of IoT systems and the manifestation of anomalies. For example, anomalies in temperature readings might be influenced by diurnal cycles, weather conditions, or HVAC system malfunctions, necessitating adaptive anomaly detection approaches capable of capturing contextual dependencies.

The description information should delineate the criticality and consequences of different types of anomalies to prioritize detection efforts and response strategies. Anomalies may vary in severity, ranging from benign fluctuations in sensor readings to catastrophic system failures or security breaches. Understanding the potential impact of anomalies enables stakeholders to allocate resources effectively, deploy appropriate mitigation measures, and tailor anomaly detection algorithms to focus on detecting anomalies with the highest operational or security implications. The description information should encompass the operational constraints and requirements of the IoT system, such as real-time processing capabilities, energy efficiency, and scalability. These considerations influence the choice of anomaly detection algorithms, deployment architectures, and trade-offs between detection accuracy and computational resources.

Dataset Description

The provided dataset appears to be related to anomaly detection in IoT sensor data, containing a multitude of features denoted by alphanumeric codes. Each row represents a different observation or instance, likely recorded over time.

The features seem to encompass various aspects of sensor readings, potentially including environmental variables, device status indicators, and other relevant metrics. These features might span a wide range, from numerical measurements to categorical flags.

For instance, features like "aa_000" through "dq_000" could represent different sensor readings or device parameters, while "class" likely denotes the label indicating whether an observation is considered normal or anomalous.

Given the nature of IoT sensor data, the dataset likely captures measurements from multiple sensors deployed across different locations or devices. The values within the dataset may vary significantly based on factors like sensor types, deployment environments, and operational conditions.

In anomaly detection tasks, the goal is typically to identify unusual patterns or outliers in the data that deviate from normal behavior. This could involve detecting sudden spikes or drops in sensor readings, unusual combinations of measurements, or patterns indicative of malfunction or intrusion.

To effectively utilize this dataset for anomaly detection, preprocessing steps such as normalization, feature selection, and handling missing values may be necessary. Additionally, employing appropriate machine learning algorithms or anomaly detection techniques, such as isolation forests or autoencoders, would be crucial for building a robust anomaly detection model.

V. RESULTS AND DESCRIPTION

Figure 2 is shows count plot is a visualization of the number of posts in each category. The x-axis shows the categories, which appear to be “pos” and “neg”. The y-axis shows the number of posts in each category. There is a total of 6726 posts according to the plot title.

The most common category is “pos”, with a count of roughly 7000. The least common category is “neg”, with a count of 138.

In conclusion, this count plot shows that there are significantly more positive posts than negative posts.

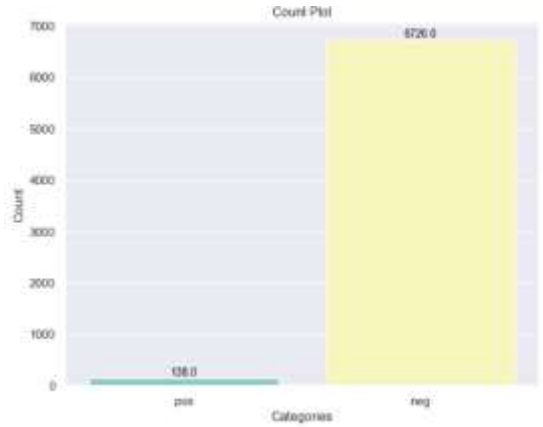

Figure 2 Count Plot of Output variables

```
Model loaded successfully.
Logistic Regression Accuracy    : 97.88783685360525
Logistic Regression Precision   : 74.45254431103487
Logistic Regression Recall      : 95.42485395645247
Logistic Regression FSCORE      : 81.55467537581357

 Logistic Regression classification report
              precision    recall  f1-score   support

    POSITIVE       0.98      1.00      0.99      1320
    NEGATIVE       0.93      0.49      0.64        53

    accuracy                           0.98      1373
   macro avg       0.95      0.74      0.82      1373
weighted avg       0.98      0.98      0.98      1373
```

Figure 3: Classification Report Logistics Regression

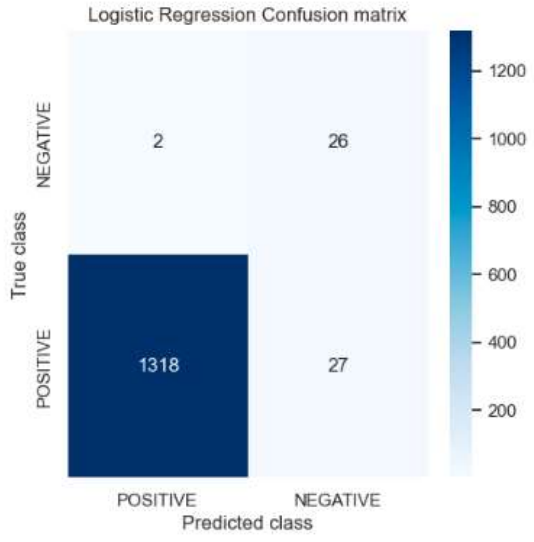

Figure 4: Confusion Matrix of Logistic Regression

```
Model loaded successfully.
XGBoost Classifier Accuracy     : 99.63583394027677
XGBoost Classifier Precision    : 94.75318144499178
XGBoost Classifier Recall       : 96.31704726500266
XGBoost Classifier FSCORE       : 95.52106372289965

 XGBoost Classifier classification report
              precision    recall  f1-score   support

    POSITIVE       1.00      1.00      1.00      1344
    NEGATIVE       0.93      0.90      0.91        29

    accuracy                           1.00      1373
   macro avg       0.96      0.95      0.96      1373
weighted avg       1.00      1.00      1.00      1373
```

Figure 5:Classification Report XGBoost Classifier

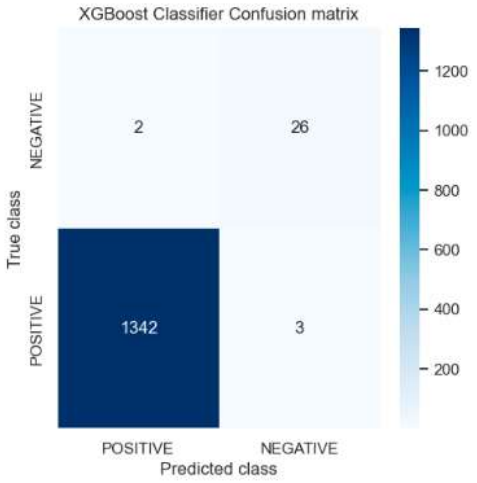

Figure 6:Confusion Matrix of XGBoost Classifier

	Algorithm Name	Accuracy	Precison	Recall	FScore
0	Logistic Regression	97.887837	74.452544	95.424854	81.554675
1	XGBoostClassifier	99.635834	94.753181	96.317047	95.521064

Figure 7: Comparison between Logistics Regression & XGBoost Classifier

Figure 3 : The classification report shows the performance of the logistic regression model on a test dataset. The test dataset likely contains two classes, positive and negative.

Looking at these metrics, we can say that the model performs well at classifying positive instances. It correctly identified almost all positive instances (recall of 1.00) and most of the predictions it made as positive were actually positive (precision of 0.98). On the other hand, the model performs poorly at classifying negative instances. It only captured half of the negative instances (recall of 0.49) and and a significant portion of negative instances were misclassified as positive.

The high accuracy (95%) is likely skewed by the fact that there are many more positive instances than negative instances in the test dataset (as evidenced by the "support" values). The model performs well on the majority class (positive) but poorly on the minority class (negative).

Figure 4 The image is indeed a confusion matrix, but it likely isn't related to logistic regression. Confusion matrices are used to evaluate the performance of classification models, and logistic regression is a classification model.

Looking at these metrics, we can say that the model performs very well on both positive and negative instances. It correctly identified almost all instances (both positive and negative) and most of the predictions it made were actually correct.

In this specific confusion matrix, the XGBoost model seems to be performing better at predicting the positive class than the negative class. There were many more negative instances incorrectly classified (1000) than positive instances (26).

Figure 6 The table is likely comparing the performance of two machine learning algorithms for classification tasks: logistic regression and XGBoost. Here's a brief description of each:

Logistic Regression: This is a well-established statistical method that is often used for binary classification problems. It works by estimating the probability of an observation belonging to a specific class by fitting a linear function to the data. Logistic regression is a relatively simple algorithm to understand and implement, and it can be interpretable, meaning it can provide insights into the features that are most important for making predictions. However, it can struggle with complex relationships between features and the target variable.

XGBoost (Extreme Gradient Boosting): This is a more recent ensemble machine learning technique which utilizes gradient boosting to enhance the performance of decision trees. It builds on the output of a series of decision trees, where each tree corrects the mistakes made by the previous tree. It is well-known for its accuracy, efficiency, and ability to find complex relationships between features and the target variable. However, it is often harder to tune and interpret than a logistic regression.

Table 2 Results for The Accuracy of Both Algorithms Here's a list of these metrics and some details about them:

Based on the table, XGBoost appears to outperform logistic regression on all four metrics. It has a higher accuracy (99.64% vs. 97.89%), precision (94.75% vs. 74.45%), recall (96.32% vs. 95.42%), and F1-score (95.52% vs. 81.55%). This suggests that XGBoost is better at classifying the data correctly.

CONCLUSION

Machine learning offers a powerful approach to anomaly detection in IoT sensor data, overcoming the limitations of traditional methods. This work presented a system that leverages supervised learning algorithms to automate anomaly detection without requiring labeled training data. By analyzing the temporal and spatial relationships within sensor readings, the proposed models effectively learn normal behavior patterns and identify significant deviations that could indicate anomalies.

Future Scope

The integration of machine learning in anomaly detection for IoT sensor data holds promise for further advancements and applications. Future research could focus on enhancing model robustness and generalizability across heterogeneous IoT environments by exploring more advanced algorithms and hybrid models that combine supervised and unsupervised learning techniques. Incorporating real-time data streaming and edge computing could improve latency and scalability, making the system more responsive and efficient in dynamic settings.

REFERENCES

[1] Zhang X, Wen F. A novel anonymous user WSN authentication for Internet of Things. Soft Computing. 2019;23(14):5683-5691. DOI: 10.1007/s00500-018-3226-6

[2] Alshehri MD, Hussain FK. A fuzzy security protocol for trust management in the Internet of things (Fuzzy-IoT). Computing. 2019;101(7):791-818. DOI: 10.1007/s00607-018-0685-7

[3] Chen H, Meng C, Shan Z, Fu Z, Bhargava BK. A novel low-rate denial of service attack detection approach in Zigbee wireless sensor network by combining Hilbert-Huang transformation and trust evaluation. IEEE Access. 2019;7:32853-32866. DOI: 10.1109/ACCESS.2019.2903816

[4] N. T. Van, T. N. Thinh, and L. T. Sach, "An anomaly-based network intrusion detection system using Deep learning," in 2017 International Conference on System Science and Engineering (ICSSE), 2017.

[5] M. Almiani, A. Abughazleh, A. Al-Rahayfeh, S. Atiewi, and A. Razaque, "Deep recurrent neural network for IoT intrusion detection system," Simulation Modelling Practice and Theory, vol. 101, 2020, pp. 102031.

[6] B. A. N.g. and S. S., "Anomaly detection framework for Internet of things traffic using vector convolutional deep learning approach in fog environment," Future Generation Computer Systems, vol. 113, 2020, pp. 255–265.

[7] Y. Liu, Z. Pang, M. Karlsson, and S. Gong, "Anomaly detection based on machine learning in IoT-based vertical plant wall for indoor climate control," Building and Environment, vol. 183, 2020, pp. 107212.

[8] Sun, X.C.; Chen, X.P. Design of UAV flight control system fault diagnosis expert system. In Equipment Manufacturing Technology; University of Wollongong: Wollongong, NSW, Australia, 2012; pp. 66–68. [Google Scholar]

[9] Liu, H.Z. Research on Intelligent Diagnosis System of UAV Flight Control Fault Based on Machine Learning; University of Electronic Science and Technology of China: Chengdu, China, 2019; pp. 20–25. [Google Scholar]

[10] Singh, S.; Murthy, T.V.R. An Expert System Based Sensor Fault Accommodation for Lateral Dynamics of Aircraft Models. Eur. J. Mol. Clin. Med. 2020, 7, 2904–2916. [Google Scholar].

[11] Qing, L.Y. Research on Airplane Fault Prognosis and Diagnosis System Based on Flight Data; Nanjing University of Aeronautics and Astronautics: Nanjing, China, 2007. [Google Scholar]

[12] Chen, M.; Pan, Z.; Chi, C.; Ma, J.; Hu, F.; Wu, J. Research on UAV Wing Structure Health Monitoring Technology Based on Finite Element Simulation Analysis. In Proceedings of the 2020 International Conference on Prognostics and System Health Management, Jinan, China, 23–25 October 2020; IEEE: Piscataway, NJ, USA; pp. 86–90. [Google Scholar]

[13] Tan, J. Research on Fault Diagnosis Technology of Flight Control System Based on Analytical Model; Nanjing University of Aeronautics and Astronautics: Nanjing, China, 2020; pp. 12–15. [Google Scholar]

[14] Melnyk, I.; Matthews, B.; Valizadegan, H.; Banerjee, A.; Oza, N. Vector autoregressive model-based anomaly detection in aviation systems. J. Aerosp. Inf. Syst. 2016, 13, 1–13. [Google Scholar] [CrossRef]

[15] Liu, Z.C.; Guo, L.J. Fault detection technology for UAV control system based on hierarchical filtering algorithm. Comput. Meas. Control 2020, 28, 23–26. [Google Scholar]