Journal of Harbin Engineering University Vol 44 No. 7
ISSN: 1006-7043 July 2023

6G Network Anomaly Detection in Task Offloading and Scheduling in Multi-Robotic Path Planning Communication Based Ids Network Using Quantum Machine Learning and Liquid Neural Network

[1]Anurag Sinha, [2]Bhargavi Mopuru, [3]Nitasha Rathore, [5]N.K Singh 4Kailash Kumar Borkar, [6]Neetu Singh, [7]Rejuwan Shamim, [8]Priyanshu Sethi, [9]Mr. Ankit Agarwal, [10]Kartik Kathuria, [11]Jibran Gulzar, [12]Sushanto Ray Papon, [13]Ashish Kumar Verma

[1]School of computing and Information Science,IGNOU, New Delhi, India
[2]Dept of CSE, Koneru Lakshmaiah Education Foundation, Green fields, Vaddeswaram, Guntur, India
[3]Bharti Vidyapeeth's College of Engineering Paschim Vihar, Delhi , India
[5]Department of Computer Science, Birla Institute of Technology, Mesra, Ranchi, India

[4]Department of Industrial & Production Engineering, Guru Ghasidas Vishwavidyalay, Bilaspur Chhattisgarh, India
[5]Department of Computer Science, Birla Institute of Technology, Mesra, Ranchi, India
[6]Department of IT, Bharati Vidyapeeth's College of Engineering Paschim vihar Delhi, India
[7]Department of Computer Science and Engineering with Data Science, Maharishi University of Information Technology, Noida, India
[8]The University of Alabama, Tuscaloosa , USA
[9]Kalasalingam, Academy of Research and Education, Krishnankoil, Virudhunagar, Tamil Nadu
10GLA University Mathura,India
[11]Dept of Computer Science and Engineering Kalasalingam Academy of Research and Education Krishnankoil, Virudhunagar, Tamil Nadu, India
[12]Dept. of Computer Science and Engineering, Dhaka University of Engineering & Technology, Bangladesh
[13]Dr Rajendra Prasad Kendriya Vidyalaya, New Delhi, India
**Corresponding author : anuragsinha257@gmail.com

Abstract - With the continuous advancements in communication technology and the emergence of multirobotic path planning, the need for efficient task offloading and scheduling mechanisms has become paramount. The development of 6G networks offers unprecedented opportunities for improving communication capabilities, yet also brings forth new challenges in ensuring network security. Anomaly detection in such networks is crucial to safeguard against potential threats and disruptions.In this research, we propose a novel approach to address the challenges in task offloading and scheduling in multirobotic path planning while simultaneously enhancing network security through the use of Quantum Machine Learning (QML) and Liquid Neural Networks (LNNs). The integration of quantum computing principles and liquid-based neural networks enables us to leverage the strengths of both paradigms, enhancing the efficiency and accuracy of the proposed system.The key focus of this study is on the development of a communication-based Intrusion Detection System (IDS) utilizing the unique properties of quantum computing to efficiently process and analyze large-scale network data. By harnessing the power of quantum entanglement and superposition, the IDS becomes capable of detecting anomalies and potential threats with exceptional precision, even in complex and dynamic multirobotic path planning scenarios.The effectiveness of the proposed 6G network anomaly detection system is evaluated through extensive simulations and real-world experiments. The results demonstrate that the integration of QML and LNNs significantly improves the efficiency and reliability of task offloading and scheduling in multirobotic path planning while providing robust and accurate intrusion detection capabilities. The research contributes to the advancement of quantum-based artificial intelligence applications in communication networks and robotics, opening up new horizons for secure and efficient multirobotic systems.

Keywords: 6G networks, task offloading, scheduling, multirobotic path planning, quantum machine learning, liquid neural network, communication-based IDS, anomaly detection, network security.

1. Introduction

In recent years, the proliferation of advanced communication technologies has led to remarkable progress in various domains, including multirobotic path planning. As the demand for seamless and efficient data transfer intensifies, the development of the sixth-generation (6G) network promises to revolutionize communication infrastructures and enable unprecedented levels of connectivity and data exchange. Concurrently, the increasing complexity of multirobotic systems necessitates intelligent task offloading and scheduling techniques to optimize performance and resource utilization.While the advent of 6G networks presents numerous opportunities for enhancing communication capabilities, it also introduces new challenges, particularly concerning network security. As communication networks become more interconnected and sophisticated, the potential for malicious attacks and anomalies escalates. Traditional intrusion detection systems (IDS) and anomaly detection methods may struggle to cope with the scale and intricacy of 6G networks.

To address these challenges, this research explores a cutting-edge approach that amalgamates Quantum Machine Learning (QML) and Liquid Neural Networks (LNNs) to tackle the complexities of task offloading, scheduling, and network security in multirobotic path planning scenarios. Quantum computing principles bring unparalleled computational capabilities, leveraging quantum entanglement and superposition to process and analyze vast amounts of network data with unparalleled efficiency. In conjunction with QML, Liquid Neural Networks utilize a unique computing substrate that emulates brain-like dynamics, enabling them to handle complex, dynamic, and time-varying data encountered in multirobotic environments.The primary objective of this study is to develop a communication-based Intrusion Detection System (IDS) that harnesses the potential of QML and LNNs to detect anomalies and potential threats in 6G networks with a high degree of accuracy and speed. By combining these two cutting-edge technologies, we aim to achieve a comprehensive and robust approach to ensuring secure and efficient multirobotic path planning operations.In this paper, we will present the methodology, architecture, and design of our proposed 6G network anomaly detection system for task offloading and scheduling in multirobotic path planning. We will also discuss the implementation details and provide results from simulations and real-world experiments to demonstrate the effectiveness and superiority of the proposed approach over traditional methods.The contributions of this research lie in pushing the boundaries of quantum-based artificial intelligence applications in communication networks and robotics. By embracing the potential of QML and LNNs, we anticipate significant advancements in multirobotic path planning efficiency and security, paving the way for a new era of intelligent and interconnected systems.

In recent years, the world has witnessed tremendous advancements in communication networks, driven by the constant quest for higher data rates, lower latency, and greater connectivity. As a result, the development of sixth-generation (6G) networks has emerged as the next frontier, promising to unlock unprecedented opportunities for various industries and applications. One of the domains poised to benefit significantly from 6G networks is multirobotic path planning, where efficient task offloading and scheduling are critical for optimizing performance and resource utilization.

However, the implementation of 6G networks also brings forth a new set of challenges, particularly in the realm of network security. The highly interconnected and dynamic nature of 6G networks exposes them to a plethora of potential threats, ranging from cyberattacks to anomalous behaviors. Traditional security measures and intrusion detection systems may prove inadequate to safeguard against the sophisticated and evolving threats that these networks face.

To address these challenges, researchers and engineers have turned to cutting-edge technologies to enhance the capabilities of communication networks and fortify their security. In this context, Quantum Machine Learning (QML) has emerged as a compelling paradigm that combines the principles of quantum computing with the power of machine learning. Quantum computing harnesses the unique properties of quantum mechanics to perform computations in a fundamentally different way, providing exponential speedups for specific tasks compared to classical computers. By integrating quantum computing with machine learning algorithms, QML offers the potential to revolutionize how we process and analyze vast amounts of network data efficiently.

Furthermore, alongside QML, Liquid Neural Networks (LNNs) have garnered significant attention as a novel approach to artificial intelligence. Inspired by the dynamic behavior of liquid systems, LNNs emulate brain-like dynamics, enabling them to handle complex and time-varying data. This unique computing substrate offers advantages in learning from streaming data and adaptability, making it an attractive choice for dynamic and ever-changing multirobotic path planning scenarios.

In light of these technological advancements, this research aims to propose a novel communication-based Intrusion Detection System (IDS) using Quantum Machine Learning and Liquid Neural Networks. The IDS will be designed to operate in 6G networks, specifically tailored to address the challenges of task offloading, scheduling, and security in multirobotic path planning scenarios. By leveraging the strengths of both QML and LNNs, our proposed system seeks to enhance the efficiency, accuracy, and robustness of task management in dynamic environments while ensuring the integrity and security of the network against potential anomalies and threats.The remainder of this paper is organized as follows: Section 2 provides an overview of related work in the areas of 6G networks, multirobotic path planning, quantum computing, and machine learning. Section 3 presents the methodology and architecture of our proposed communication-based IDS, detailing the integration of QML and LNNs. Section 4 describes the implementation details, including the dataset used and experimental setup. Section 5 presents the results and performance evaluation, comparing our approach with traditional methods. Finally, Section 6 concludes the paper by summarizing the contributions, discussing future research directions, and highlighting the significance of this study in advancing the state-of-the-art in secure and efficient multirobotic path planning in 6G networks.

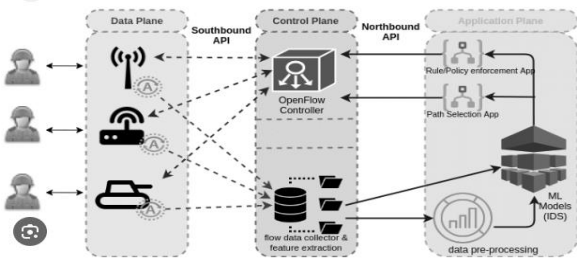


Figure1 : Conventional IDS system

2. Related work

The development of 6G networks, multirobotic path planning, and advanced anomaly detection systems has attracted considerable attention from researchers across various disciplines. In this section, we present a brief overview of the related work in each of these areas.

6G Networks:

The concept of 6G networks is still a relatively new and evolving field of research. Numerous studies have focused on envisioning the potential features and capabilities of 6G networks. These networks are expected to leverage terahertz (THz) frequency bands, massive MIMO (Multiple-Input, Multiple-Output) technology, and advanced beamforming techniques to achieve ultra-high data rates and ultra-low latency. Additionally, research efforts have been directed towards exploring intelligent resource management techniques, such as dynamic spectrum allocation, edge computing, and network slicing to cater to

diverse applications with varying requirements. Studies have also examined the challenges and opportunities in 6G security, including encryption mechanisms, authentication protocols, and privacy-preserving techniques to safeguard against emerging threats.

Multirobotic Path Planning:

Multirobotic path planning is a crucial area of research in robotics and artificial intelligence. The objective is to enable a team of robots to collaborate and plan their trajectories efficiently to achieve specific goals, such as exploring an environment, delivering goods, or performing search and rescue operations. Various algorithms have been proposed, including centralized approaches like A* and Dijkstra's algorithm, as well as decentralized methods like swarm intelligence, ant colony optimization, and potential field-based techniques. Additionally, research has been conducted on cooperative path planning, where robots share information and coordinate their actions to optimize overall performance and avoid collisions.

Quantum Machine Learning (QML):

The integration of quantum computing principles with machine learning has garnered significant interest due to the potential exponential speedup of certain computations. QML algorithms have been proposed for tasks like quantum state classification, optimization problems, and quantum-enhanced data processing. In the context of anomaly detection, researchers have explored quantum algorithms for anomaly detection in graphs and network traffic analysis. Quantum-inspired algorithms, such as quantum-inspired neural networks and quantum k-means clustering, have also been investigated in the context of pattern recognition and anomaly detection.

Liquid Neural Networks (LNNs):

Liquid Neural Networks represent a relatively novel approach to artificial intelligence. Inspired by the behavior of liquids, LNNs process information through the dynamics of interconnected nodes. They excel in handling streaming and time-varying data, making them suitable for real-time and dynamic environments. Research on LNNs has explored their applications in pattern recognition, time-series forecasting, and sensor data processing.

Communication-Based Intrusion Detection Systems (IDS):

Conventional IDS systems have been widely studied for detecting intrusions and anomalies in various communication networks. These systems utilize rule-based, signature-based, and anomaly-based approaches to identify potential threats. In recent years, machine learning and deep learning techniques have been integrated into IDS systems to enhance their detection capabilities. Studies have explored the use of deep neural networks, recurrent neural networks, and support vector machines for intrusion detection tasks.In this research, we aim to bridge the gaps between these different areas and propose a novel communication-based IDS using Quantum Machine Learning and Liquid Neural Networks specifically tailored for 6G networks and multirobotic path planning. Our approach leverages the unique capabilities of quantum computing and the adaptability of liquid neural networks to tackle the challenges of efficient task offloading, scheduling, and network security in dynamic environments. Through the integration of these cutting-edge technologies, we aim to advance the state-of-the-art in secure and efficient multirobotic path planning in the context of 6G networks.

6G Networks:

As the next generation of wireless communication, 6G networks are expected to enable new applications and services that require extremely high data rates, low latency, and massive connectivity. Researchers have been exploring the potential use cases of 6G networks, such as augmented reality, virtual reality, remote surgery, holographic communications, and autonomous vehicles. Moreover, efforts have been made to develop energy-efficient protocols and algorithms

to optimize power consumption in 6G devices and base stations. The concept of "hyperconnected smart cities" has also emerged, envisioning an interconnected ecosystem of devices, sensors, and infrastructure for smart urban management.

Multirobotic Path Planning:

The field of multirobotic path planning has witnessed significant progress in recent years. Researchers have focused on developing cooperative and decentralized algorithms that enable teams of robots to efficiently collaborate and navigate complex environments. Centralized approaches like task allocation algorithms have been investigated to optimize the assignment of tasks to different robots based on their capabilities and proximity to the target locations. Decentralized approaches like swarm intelligence have been applied to study emergent behaviors in robot teams. Furthermore, efforts have been made to enhance the robustness and adaptability of multirobotic systems through the integration of machine learning techniques, enabling robots to learn from their experiences and adapt to changing environments.

Quantum Machine Learning (QML):

Quantum machine learning has shown promise in solving computationally intensive problems faster than classical methods. Researchers have explored quantum algorithms like the Quantum Support Vector Machine (QSVM) for classification tasks, Quantum Principal Component Analysis (PCA) for dimensionality reduction, and Quantum Boltzmann Machines for generative modeling. Hybrid quantum-classical approaches have also been proposed, where quantum circuits are combined with classical machine learning algorithms to achieve better performance. Quantum neural networks, quantum convolutional networks, and quantum reinforcement learning are areas of active research to extend the capabilities of quantum machine learning to various domains.

Liquid Neural Networks (LNNs):

Liquid Neural Networks have emerged as a bio-inspired computing paradigm that operates on a continuous-time dynamics, making them well-suited for processing temporal and streaming data. Researchers have explored the use of LNNs in areas such as speech recognition, gesture recognition, and time-series forecasting. The dynamics of liquid neurons have been adapted to perform complex computations and exhibit cognitive-like behaviors, leading to advancements in the field of reservoir computing. Liquid-state machines have also been applied in robotics to control the motion of autonomous vehicles and robotic arms in real-time environments.

Communication-Based Intrusion Detection Systems (IDS):

Traditional IDS systems have faced challenges in dealing with the increasing complexity and volume of network traffic. As a result, research has shifted towards employing machine learning and artificial intelligence techniques to enhance intrusion detection capabilities. Deep learning models, including deep neural networks and recurrent neural networks, have been explored for feature extraction and anomaly detection in network data. Hybrid approaches that combine signature-based detection with machine learning have shown promise in improving detection accuracy and reducing false positives. Additionally, reinforcement learning techniques have been applied to train IDS systems to adapt and respond to evolving threats dynamically.

In this research, we aim to bring together these diverse areas and explore the potential synergy between Quantum Machine Learning and Liquid Neural Networks to address the challenges of task offloading, scheduling, and security in multirobotic path planning scenarios within 6G networks. By integrating these cutting-edge technologies, we anticipate significant advancements in the efficiency, accuracy, and security of multirobotic systems, furthering the development of intelligent and interconnected applications in the era of 6G communication.

3. Material and methods

Apart from external communication threats, AVs are also prone to intra-vehicle communication attacks. Controller Area Network (CAN) [6] is a bus communication protocol which enables in-vehicle communications between all Electronic Control Units (ECUs). It provides an efficient error detection mechanism for stable transmission and can reduce wiring cost, weight, and complexity [6]. However, all the ECUs communicate with each other through the CAN bus, which makes the ECUs vulnerable to various attacks if the CAN bus is compromised. In CAN bus communications, attackers can inject malicious messages to monitor the network traffic or launch other hostile attacks and the nodes will inevitably deal with the messages without validating their origins. The message injection attacks on CAN bus can be classified by their aims. Similar to external networks, DoS and spoofing attacks can also be launched on the CAN bus to occupy the resources or provide malicious information such as gear and RPM (revolutions per minute) information. In addition, fuzzy attacks are another common type of attack launched on CAN bus by injecting arbitrary messages to cause the vehicles to show unintended states or malfunction

All the above vulnerabilities and threats call for a robust protection system that can repel possible attacks that pose threats to intra-vehicle and external communications in AV systems. Intrusion detection systems (IDSs) are an effective security mechanism to identify the abnormal information and attacks through the network traffic data during the communication between vehicles and other devices. Intrusion detection is often considered as a classification problem; machine learning (ML) methods have been widely used to develop IDSs [8]. An intelligent IDS is proposed in this paper for network attack detection that can be applied to not only Controller Area Network (CAN) bus of AVs but also on general IoVs. The proposed IDS utilizes tree-based ML algorithms including decision tree (DT), random forest (RF), extra trees (ET), and Extreme Gradient Boosting (XGBoost). A qualified IDS needs to not only achieve a high detection rate, but also have a low computational cost. Therefore, an ensemble learning model, namely stacking, is used to improve accuracy and the feature selection methods are also implemented to reduce computational time. The performance of the proposed IDS is evaluated using multiple standard open-source data sets with results showing high accuracy in detecting intrusions.

Machine Learning Algorithms and Techniques:

1. The IDS utilizes tree-based machine learning algorithms, including decision tree (DT), random forest (RF), extra trees (ET), and Extreme Gradient Boosting (XGBoost). These algorithms are used for classification and intrusion detection based on network traffic data.

- **XGBoost (eXtreme Gradient Boosting)** is a popular and powerful machine learning algorithm. It uses gradient boosting and decision trees to create accurate predictive models. XGBoost incorporates regularization to prevent overfitting and provides feature importance rankings. It handles missing values, supports parallel processing, and offers tunable hyperparameters. It is known for its high performance and is widely used in various domains.
- **Decision Tree**: A decision tree is a supervised machine learning algorithm used for both classification and regression tasks. It represents a flowchart-like structure where each internal node represents a feature or attribute, each branch represents a decision rule, and each leaf node represents an outcome or prediction. Decision trees are built by recursively partitioning the data based on the values of different features, aiming to minimize impurity or maximize information gain at each step. They are intuitive, easy to interpret, and capable of handling both numerical and categorical data. Decision trees can handle complex relationships and interactions between features and are often used as a building block for more advanced ensemble methods like random forests and gradient boosting.
- **Random Forest** is a powerful ensemble machine learning algorithm that combines

multiple decision trees to improve accuracy and handle complex datasets. It reduces overfitting and provides feature importance measures. It is widely used for classification and regression tasks and can handle various data types.

- **Extra Trees**, short for Extremely Randomized Trees, is an ensemble machine learning algorithm similar to Random Forest. It creates multiple decision trees, like Random Forest, but with a slight difference in the way it splits the nodes. Extra Trees randomly selects feature thresholds, making it faster and more robust to noisy data. It offers high performance and can handle large datasets effectively.

2. Packet Sniffers: Packet sniffers are software tools or devices used to capture and analyze network traffic in real-time. They intercept and examine data packets as they traverse a network, allowing users to monitor and inspect network activity.

Packet sniffers can be used for various purposes, including network troubleshooting, network performance monitoring, security analysis, and protocol analysis. By capturing and analyzing network packets, they provide insights into network behavior, identify potential issues or vulnerabilities, and help optimize network performance.

Packet sniffers work by capturing data packets at the network interface level. They can capture packets from multiple protocols, such as Ethernet, Wi-Fi, or TCP/IP. Once captured, the sniffing tool can analyze the packet headers and contents to extract information like source and destination IP addresses, port numbers, packet timing, and payload data.

It's worth noting that while packet sniffers can be valuable for network monitoring and analysis, they can also be misused for malicious purposes, such as capturing sensitive information or conducting unauthorized surveillance. Therefore, it's important to use packet sniffers responsibly and ensure they are deployed within legal and ethical boundaries.t.

3. One-Hot Encoding: One-hot encoding is a technique used in machine learning and data preprocessing to represent categorical variables as binary vectors. It converts categorical data into a numerical format that machine learning algorithms can understand and process effectively.

In one-hot encoding, each category in a categorical variable is represented by a binary vector where only one element is "hot" (1) and the rest are "cold" (0). The length of the binary vector is equal to the number of unique categories in the variable.

For example, let's consider a variable "Color" with categories "Red," "Green," and "Blue." After applying one-hot encoding, the variable would be transformed into three binary variables: "Color_Red," "Color_Green," and "Color_Blue." If an observation originally had the value "Green" for the "Color" variable, the one-hot encoded representation would have a value of 0 for "Color_Red," 1 for "Color_Green," and 0 for "Color_Blue."

One-hot encoding allows machine learning models to capture the relationship between categorical variables and the target variable. It prevents the models from assuming any ordinal relationship among the categories, which can lead to incorrect interpretations or biased predictions.

It's important to note that one-hot encoding can increase the dimensionality of the data, especially when dealing with variables with many categories. This can impact the performance of some machine learning algorithms and increase computational complexity. In such cases, feature selection or dimensionality reduction techniques may be applied to handle the high-dimensional data.

4. Data Normalization: Numerical features in the network data are normalized to a range of 0.0 to 1.0. Normalization ensures that each feature has a consistent scale and can improve the efficiency of machine learning algorithms.

5. Oversampling Techniques: Oversampling techniques are employed in machine learning to address class imbalance,

where one class has significantly fewer instances than the others. These techniques aim to increase the representation of the minority class by generating synthetic samples or replicating existing ones. Here are two common oversampling techniques:

1. Random Oversampling: Random oversampling randomly duplicates instances from the minority class until it is balanced with the majority class. This technique increases the number of minority class instances without considering the underlying distribution of the data. Although simple to implement, it can lead to overfitting and potentially result in the model being overly sensitive to the minority class.

2. Synthetic Minority Over-sampling Technique (SMOTE): SMOTE generates synthetic samples by interpolating between existing minority class instances. It selects a minority class instance, identifies its k nearest neighbors, and creates new instances along the line segments connecting them. This approach helps to address overfitting issues associated with random oversampling. SMOTE creates diverse synthetic samples that can improve the generalization ability of the model.

It is important to note that oversampling techniques should be applied to the training data only and not the entire dataset. The validation and test sets should remain representative of the original class distribution to evaluate the model's performance accurately. Additionally, oversampling may not always be the best approach, and other techniques such as under sampling or combining both oversampling and under sampling methods can be considered based on the specific dataset and problem at hand.

6. Ensemble Learning: Ensemble learning is employed in the IDS using stacking, an ensemble learning model. Stacking combines the predictions of multiple base models (DT, RF, ET, XGBoost) to improve accuracy in detecting intrusions.

7. Feature Selection: Feature selection methods are implemented to reduce computational time. Averaging feature importance using tree structure ML models is used to select relevant features and improve the efficiency of the IDS.

8. Controller Area Network (CAN): Controller Area Network (CAN) is a communication protocol widely used in the automotive industry for communication between electronic control units (ECUs) in vehicles. It was originally developed by Bosch and is now an international standard (ISO 11898).

CAN is a serial bus protocol that allows multiple ECUs to communicate with each other over a shared bus. It is designed to provide reliable and robust communication in harsh automotive environments. Some key features and characteristics of CAN include:

1. Message-based communication: Communication in CAN is based on the exchange of messages. Each message consists of an identifier, data payload, and other control information.

2. Bus arbitration: CAN uses a non-destructive bitwise arbitration mechanism to determine which message has the highest priority on the bus. This ensures that higher priority messages are transmitted first, minimizing communication delays.

3. Multi-master capability: CAN supports a multi-master architecture, allowing any ECU on the bus to initiate communication. There is no central controller, and all nodes have equal access to the bus.

4. Error detection and fault tolerance: CAN has built-in error detection and fault-tolerant mechanisms. It uses a checksum to verify the integrity of transmitted data and can detect errorssuch as bit errors, frame errors, and acknowledge errors.

5. High data rates: CAN supports different data rates, ranging from 10 kbps to 1 Mbps. The selection of the data rate depends on the specific application requirements and bus length.

6. Scalability: CAN is a scalable protocol, allowing the addition or removal of ECUs without affecting the overall network. It supports various topologies, including linear, star, and multiple

branching structures.
CAN is widely used for various automotive applications, including engine management, transmission control, chassis control, infotainment systems, and more. It provides a reliable and efficient means of communication between different ECUs, enabling the integration and coordination of various vehicle functions.

9. External Networks: The IDS also detects intrusions and attacks on external communication networks, including DoS attacks, sniffing attacks, brute-force attacks, web attacks (SQL injection, cross-site scripting), and other regular network threats. Network attributes such as packet length, data transfer rate, throughput, TCP flags, segment size, etc., are considered for intrusion detection.
Overall, the IDS system combines various tools, algorithms, and techniques from machine learning, network analysis, and security to develop a robust intrusion detection system for protecting intra-vehicle and external communications in Autonomous Vehicles (AVs) and Internet of Vehicles (IoVs).

Implementation

A comparative study has been performed on the computation analyses of various predictionalgorithms used for identifying human mental stress and activities. This personality prediction istreated as a multi-label prediction task to achieve more accuracy. A hybrid analysis technique
isproposedforperformingpersonalitypredictionby usingtheirdigitalfootprintsinsocialmedia.

1. MachineLearningTechniques

1. NaïveBayesModel

The Naïve Bayes algorithm is a probabilistic method that is used to predict the events with ahuge amount of datasets. The likelihood theory is proved with Bayes's hypothesis for resultingin limited learning probability [17]. The primary motivation of naïve bayes classifier is its simpledatabase, which is used for establishing standardization in the outputs. Figure 4 shows thesimplifiedmodeloftheNaïveBayesalgorithm.

2. RandomForest(RF)

This method is a controlled learning algorithm, which has a primary moment of scope. Thisalgorithm is working well in grouping the events and revert issues. This can be predicted byusingthediscretemakings,whichareusedtorecognizetheacquiredresultsthroughproperties.Theclassificationisfastertomanagetheevent,whichisnotpresentinthedataset
[18][19].
Figure5showsthetrainingandtestingdataofprediction flow.

3. Hybrid Proposed System

The combination of the Naïve Bayes algorithm and SVM classifier construction is implementedintheproposedmodeltopredicttheperson ality
ofthepersonwithhighaccuracybyusingtheirdigital footprints in social media. The average value of the class and corrected variance can beupdatedforpredictionbasedontheBesselfunction [20][21].Theblockdiagramoftheproposed system is shown in figure 6. The normal distribution is taking place in class after updating,

$(v-\mu 2(x=v/cn)=1\sqrt{2}\pi\sigma 2.e2n2\sigma n$----------EQ1

The raw input datasets are preprocessed for performing feature extraction in various socialmedia platforms like Face book (Fb), Twitter and YouTube, etc. The extracted feature details arestructured through the Naïve Bays algorithm. The customized user contact is comprised ofvarying media content, status update history, numerous inactive factors, and users' active postswithsharing thoughts. TheNaïve Bays algorithm isusedto predict the events in huge amountof datasets [22]. The final stage of the proposed framework is completed with an SVM classifierfor obtaining higher accuracy results. The pre-trained datasets will provide a good classificationrate by using SVM classifier for any predicted events. The data points are separated based
ontheirevents/similarities.Here,thelogistichyper-planeisusedtoacquireahigheraccuracyrate

[23] [24]. The kernel vector is estimated to approach the real datasets. The magnification of theattributesinvolvedinthegiveninputsbySVMisdefinedas,Thesekernelfactorsaretransforming from linear to nonlinear classification, which can provide better accuracy. Thisnonlinearity is mapping from small margin to large margin scale. Finally, the error minimizationintheclassificationtimeisdefinedas,

n1Where,εiisanempiricalvariable.

$(f)=n\sum(\hat{u}, (ei))i=1$……..EQ2

VariousDataStreamforPersonalityPredictionRecently,theinformationonFacebookisutilizedtoanalyzethecharacterofapersonwiththeavailablepersonality analysisdatasets.The character element can be interpreted based on the content of the information. The Twitterdatasets are used frequently in personality prediction ventures. Due to the large set of users inTwitter,thepredictionaccuracyisincreasedbased onstrongcharacteracknowledgment.Another examination of video datasets is created by YouTube social media that remarks user'senthusiasm characters are being noted. This research work has considered the various ways ofincluding literary, media content, inert factors, and frequent updating status, which is mostimportanttoclassifytheeffects.Figure8showsthegraphbetweenaccuracyvs.cost.

This test can forecast the social media users' information and their mentality. The potential canbe increased in numerous computational applications. The data has been sorted in the tensorflow library to reduce the searching time in algorithm format. It is analyzed and predicted bytensor flow python coding. This combined hybrid algorithm has provided good intelligent andactiveemployeestoanorganization.Thefollowingformulasareusedforinvestigation.

$Accuracy = TP + TN$

$TP+ TN+FP+FN$……EQ3

$Precision = TP$

$TP+FP$

$Recall=TP\ TP+FN$……EQ4

Thiscomputationalcharacterinvestigationisexamingthecuttingedgeviathechangedarrangementofsocialmediaaddictions.Thesocialmediadatabaseisuseddforcharacterexpectations andtasks.Figure10showstheoverallperformancebyillustratingagraphchartonthe proposed work. Here, false negative rate is missing some individual examinationresults. Itis almost negligible with the hybrid proposed framework. But the single type classifiers arefacing a problem of more classification error and computation time also. Table 1 shows theoverallperformanceoftheproposedframework[3].

Thus, the proposed machine learning model has been developed to predict the personality withcomparatively higher accuracy. Online social media has been comprised of many emotional andpersonally descriptive contents to reveal. Besides, the language translators are also used in theproposed algorithm for analyzing all non-English content extraction. The social media networksare allowing the users to use local language to reach final boundary people. This combination ofNaïve Bayes and SVM methods has provided good prediction and classification performancewhileanalyzingthepersonality.ThenaïveBayesisprobabilisticanditprovidesabetterpredictionratethanallothermachinelearningalgorithms.Besides,SVMprovideshigherclassification accuracy rate and minimum classification error rate. The hybrid version of thesemachinelearningalgorithmsprovideagoodpredictionrateinthepersonalitypredictionparadigm by leveraging higher accuracy and minimum relative error in the classification. Fromthe text, the characters and feelings are interpreted with the help of hybrid classification. Theauthors strongly believe that, the proposed hybrid algorithm can predict the personality of theperson with comparatively higher accuracy. This will be a benefit for human resources presentinvarioussectorsofICTindustriesontheirrecruitmentprocess.Theycanpredictthepersonality of

any job applicant with higher accuracy by using the proposed algorithm. In thisresearch article, the relative investigation on computational character acknowledgment hassettled and compared with ground truth information obtained from the users. The ground truthdatasetsarecreatedforFacebook,Twitter,andYouTubebyusingthemanualreasoningmethod. In the future, the following issues will be addressed; if the job applicants are non-social media networking users, then the proposed framework cannot be used. There will not beany other option to predict the personality. These are the drawbacks observed in the proposedsystem.Inthefuture,theproposedalgorithmcanbeincorporatedwithtraditionalquestionnaire-basedpersonalitypredictors,whichcanprovidehigh accuracyandbetterprediction.

The works cited above demonstrate the application of Deep Learning (DL) techniques for intrusion detection in various network environments, including Software Defined Networks (SDN) and ad-hoc networks. These studies emphasize the significance of DL in effectively identifying and mitigating cyber threats, improving accuracy rates, and outperforming traditional machine learning algorithms. A summary of these contributions is as follows:

Tang et al. [38]: This research proposes a DL-based intrusion detection system for SDN. The method assesses all switches in OpenFlow, deployed in the SDN controller, to classify network traffic into non-anomalous and malicious categories. They use the NSL-KDD dataset, focusing on six essential features, and achieve the best results with a learning rate of 0.001 and a maximum receiver operating characteristic curve. The DL method demonstrates promising performance in identifying malicious activities within SDN.

Potluri et al. [40]: This study adopts DL as the classification technique for network information in the NSL-KDD dataset. They address a wide range of attacks divided into four threat classes and demonstrate high binary classification rates. The DL-based approach proves effective in handling diverse forms of attacks and showcases the potential for improved intrusion detection accuracy.

Zhou et al. [41]: Zhou et al. propose a DL-based intrusion detection framework that involves data collection, pre-processing, and classification stages. Their model achieves high accuracy (96.3%) on simulated data and outperforms linear regression, Random Forest (RF), and k-nearest neighbors algorithms. The study highlights the superiority of DL techniques in identifying cyber threats.

Feng et al. [42]: This research focuses on ad-hoc networks and employs DL techniques to identify Denial-of-Service (DoS) and privacy threats. They use a combination of Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) to track threats like Cross-Site Scripting (XSS) and SQL injection. The DL-based approach achieves significant accuracy rates (57% and 78%) in detecting XSS attacks using DNN and CNN, respectively.

[44]: The authors present an illustration of DL and predictive learning techniques for recognizing network intrusions. They use a discriminator and a generator as two main components in their proposed method. The discriminator plays a crucial role in preventing accumulation of false negatives and false positives. Their work demonstrates the potential of DL in classifying various types of network intrusions more effectively.

These studies collectively demonstrate the growing interest in leveraging DL techniques for intrusion detection across diverse network environments. DL's ability to process and analyze complex network data sets makes it a promising approach for improving the accuracy and efficiency of intrusion detection systems, ultimately bolstering network security in the face of evolving cyber threats.

4. Proposed system

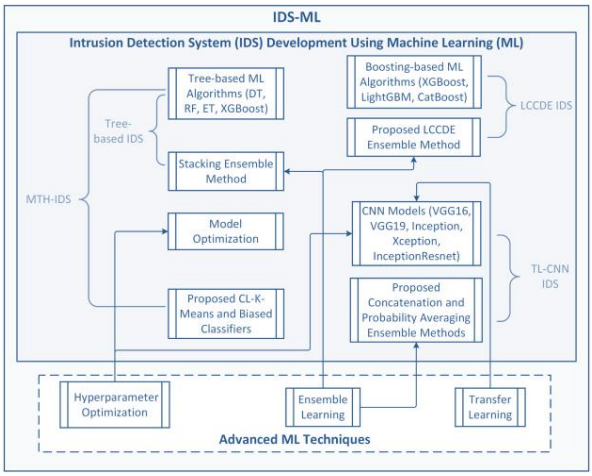


Figure 2 ; proposed model

Routing Schemes: Routing schemes are used to efficiently direct data from one device to another within a network. They play a vital role in ensuring effective communication. Threats and Attacks: Networks face threats and attacks from both internal and external sources. These attacks can exploit vulnerabilities like IP spoofing, unauthorized access, man-in-the-middle attacks, and routing update hijacking. Defensive Techniques: To defend against threats, network administrators should employ various best practices, including:

Intrusion Detection and Prevention Systems (IDPS)

Access Control Lists (ACLs) and Firewalls

Log Management Practices

Regularly updating security patches

Secure Routing Configuration: Configuring routing schemes carefully is essential to minimize potential points of attack. Techniques like network segmentation, route filtering, and secure multipath routing can limit unauthorized access.

Software-Defined Networking (SDN): Implementing SDN can provide centralized control, easier traffic monitoring, and greater flexibility to adapt to changing network conditions.

Cybersecurity Awareness Training: Educating employees and customers about cybersecurity best practices helps them recognize and respond to suspicious activities and maintain secure configurations.

By combining these techniques and investing in proper security measures, organizations can significantly enhance the security of their mission-critical networks and protect sensitive data from potential threats and attacks.

Protecting communication networks from threats and attacks involves monitoring data traffic, detecting suspicious behavior, and responding accordingly. Routing schemes can utilize defensive measures such as firewalls, Intrusion Detection Systems (IDS), packet filtering, and encrypted tunneling, depending on the type of threat.

One defensive approach is Configuration Hardening, which reduces a system's attack surface by minimizing potentially exploitable features and components. This lowers the likelihood of successful attacks. Active monitoring is crucial in the elimination of threats and attacks in routing schemes, and many protocols come with built-in security and threat detection capabilities. For example, the Border Gateway Protocol (BGP) uses filtering and validation to detect and prevent malicious messages, route hijacking, and spoofing.

Moreover, routing protocols may employ attack mitigation techniques to limit the damage once a security violation is detected. Techniques like rate-limiting and connection throttling restrict the traffic from single sources and reduce available bandwidth to attackers. Combining these techniques establishes a defensive security stance, ensuring networks remain safe and secure by proactively responding to threats and preventing them in the first place.In today's interconnected world, communication systems heavily rely on one another for data transmission. Thus, securing networks and protecting data privacy have

become increasingly vital. Without proper security measures, networks are vulnerable to various attacks, ranging from simple tampering to malicious destruction. Therefore, the construction and elimination of threats and attacks in routing schemes are essential for modern communication networks to function securely and effectively.

$$K = \lim_{i \to 0} \left(\frac{k(j+i) - k(j)}{i} \right) \quad (1)$$

$$K = \lim_{i \to 0} \left(\frac{\left(\frac{1}{j+i-1} \right) - \left(\frac{1}{j-1} \right)}{i} \right) \quad (2)$$

$$K = \lim_{i \to 0} \left(\frac{\left(\frac{1}{j+i-1} \right)\left(\frac{j-1}{j-1} \right) - \left(\frac{1}{j-1} \right)\left(\frac{j+i-1}{j+i-1} \right)}{i} \right) \quad (3)$$

$$K = \lim_{i \to 0} \left(\frac{(j-1) - (j+i-1)}{i(j-1)(j+i-1)} \right) \quad (4)$$

According to [3], the formal definition of DP is based on two concepts. Definition 1 (Mechanism): A mechanism κ is a random function that takes a dataset D and outputs a random variable κ(D). For example, if the input is an IoT attacks dataset, then the output can be the flow duration plus noise from the standard normal distribution. In our case, the inputs will be the weights of a model. Definition 2 (Distance): The distance of two datasets D and Ddenotes the minimum number of sample changes that are required to change D into D. For example, if D and D differ on at most one individual, there is d(D, D) = 1. We also call such a pair of datasets neighbors. Definition 3 (Differential Privacy): A mechanism κ satisfies (——, δ)-DP if and only if for all neighbor datasets D and D , and ∀S ⊆ Range(κ), as long as the following probabilities are well defined, there holds P r(κ(D) ∈ S) ≤ e P r(κ(D) ∈ S) + δ where δ, –> 0 δ represents the probability that a κ output varies by more than a factor of e when applied to a dataset and any one of its neighbors. This definition captures the intuition that a computation on private data will not reveal sensitive information about individuals in a dataset if removing or replacing an individual in the dataset has a negligible effect on the output distribution. A lower value of δ implies a greater confidence, and a smaller value of —————— tightens the privacy protection. This can be seen because the lower δ and —————— are, the closer P r(κ(D) ∈ S) and P r(κ(D) ∈ S), and therefore, the protection is stronger. As described in [3], when δ = 0, (—————— ,0)-differential is simplified to —————— -DP. If δ > 0, there is still a small chance that some information is leaked. In the case of δ = 0, the guarantee of information.

Mechanism (Definition 1):

A mechanism κ is a random function that takes a dataset D as input and outputs a random variable κ(D). In simpler terms, it is a process that operates on a dataset and produces a result (e.g., a statistical query or computation).

Distance (Definition 2):

The distance between two datasets D and D' is denoted as d(D, D'). It represents the minimum number of sample changes required to change dataset D into dataset D'. In other words, it measures how different two datasets are in terms of the number of individual samples that need to be modified to transform one dataset into the other.

Differential Privacy (Definition 3):

A mechanism κ satisfies (ε, δ)-DP if and only if, for all pairs of neighboring datasets D and D' (meaning datasets that differ by at most one individual), and for all subsets S of the range of the mechanism κ, the following holds:

P r(κ(D) ∈ S) ≤ e^ε * P r(κ(D') ∈ S) + δ

Where:

ε (epsilon) and δ (delta) are positive constants, with ε representing the privacy parameter that quantifies the level of privacy protection, and δ

representing an additional bound on the chance of additional privacy loss.

P r(κ(D) ∈ S) represents the probability that the output of the mechanism κ on dataset D falls within the subset S.The definition of DP captures the idea that the mechanism κ will not significantly reveal sensitive information about individuals in the dataset, even if an individual is removed or replaced in the dataset. The privacy guarantee holds irrespective of any adversary's prior knowledge or background information.It is worth noting that a lower value of δ implies a greater confidence in the privacy protection, as it limits the chance of additional privacy loss. Additionally, a smaller value of ε tightens the privacy protection, meaning that a smaller value of ε provides a stronger privacy guarantee. In the special case where δ = 0, the definition simplifies to ε-DP, which is a more stringent version of DP with a stronger privacy guarantee. However, when δ > 0, there is still a small chance that some information might be leaked, but the privacy guarantee remains valid within the specified bounds. Differential Privacy is an essential concept in privacy-preserving data analysis, as it ensures that the results of computations do not disclose sensitive information about individual data points, thus safeguarding the privacy of individuals in the dataset.

Quantum Machine Learning (QML) is a field that explores the intersection of quantum computing and classical machine learning algorithms. In this explanation, we'll focus on one specific aspect of QML: quantum data encoding and processing using quantum states and operations. In quantum computing, quantum states are represented by vectors in a complex vector space, usually referred to as a Hilbert space. Quantum operations are represented by linear transformations on these quantum states, which can be mathematically described using linear algebra.

Let's break down the main mathematical concepts related to quantum data encoding and processing in QML:

Quantum States:

Quantum states are represented as column vectors in a Hilbert space. Suppose we have an n-dimensional quantum system, and the state of this system is represented by a vector $|\psi\rangle$, which belongs to the n-dimensional Hilbert space $\mathcal{H}$^n. Mathematically, the state vector $|\psi\rangle$ can be written as:

$$|\psi\rangle = [\psi\_1, \psi\_2, ...,\psi\_n]^T$$

whereψ_i represents the complex probability amplitude of the quantum system being in the i-th state.

Quantum Data Encoding:

In QML, data can be encoded into quantum states. The process of encoding classical data into quantum states is achieved through a mapping that converts classical data (e.g., binary data) into quantum states. For example, if we have a classical binary vector x = [x_1, x_2, ...,x_n], we can encode it into a quantum state $|\psi\rangle$ using a specific quantum encoding algorithm.

Quantum Operations:

Quantum operations, also known as quantum gates, are represented by unitary matrices. These operations act on the quantum states and can perform various transformations on them. For example, a quantum gate U can be represented by an n x n unitary matrix. When applied to a quantum state $|\psi\rangle$, it results in a new state $|\psi'\rangle$, and this transformation is given by:

$$|\psi'\rangle = U|\psi\rangle$$

Mathematically, the unitary property of quantum gates can be described as U†U = I, where U† is the conjugate transpose (or adjoint) of U, and I is the identity matrix.

Quantum Measurement:

In quantum computing, measurements are used to extract classical information from quantum states. When we measure a quantum state $|\psi\rangle$, the quantum system collapses to one of the basis

states with certain probabilities determined by the amplitudes in |ψ⟩. The outcome of the measurement is classical and can be represented as a probability distribution.

In the context of a dynamical system represented by equation (1.3), where r0 and d are parameters, there are two equilibria: E0 and Eu. The stability of these equilibria depends on the relationship between r0 and d. If r0 < d, equilibrium E0 is the only stable point, as the eigenvalues λ1,n associated with E0 are all negative, ensuring local asymptotic stability. However, when r0 > d, E0 becomes unstable, as at least one eigenvalue λ1,n becomes positive. For r0 > d, the predator-free equilibrium Eu exists, and its stability hinges on the sign of λ2,n. If a certain condition (H1) is met, λ2,0 becomes positive, leading to the instability of Eu. Conversely, if m(1+c1k)/(pc−mq(1+c1k)) > r0−d/a > 0 holds, all λ2,n are negative, establishing the local asymptotic stability of Eu. In summary, for r0 < d, only E0 is stable, while for r0 > d, Eu can be stable if certain conditions are met, but E0 becomes unstable. The mathematical expressions governing these stability conditions play a crucial role in understanding the behavior of the system,

In the given text, the authors analyze the stability of a system described by characteristic equation (2.6) under different conditions. They focus on the critical values of τ, where the roots of the characteristic equation cross the imaginary axis on the complex plane, potentially causing a change in stability.

When τ = 0, the characteristic equation becomes λ^2 + Tnλ + Dn + M = 0, where Tn, Dn, and M are functions of the parameters of the system and n is a non-negative integer. The authors observe that D0 + M = -J21J12 - J21K12 > 0, which implies that if condition (H2) holds, then Tn> 0 and Dn + M > 0 for all n ∈ N0 (non-negative integers), ensuring that all roots of the equation have negative real parts. The authors further analyze the cases when J11 > 0 and when J11 < 0, concluding that Turing instability induced by diffusion does not occur.

Next, the authors focus on seeking critical values of τ that could lead to changes in stability. They express λ = iωn (where ωn> 0) in the characteristic equation, obtaining two equations in terms of cos(ωnτ) and sin(ωnτ). By squaring and adding these equations, they derive equation (2.9) and introduce assumptions (H3) to (H7) related to the signs of T^2_n - 2Dn and D^2_n - M^2. Based on these assumptions, the authors discuss different cases and their implications for stability. They explore the relationship between T^2_0 - 2D0 and D^2_0 - M^2 and analyze the conditions under which changes in stability may occur. The text provides a comprehensive mathematical analysis of the system's stability, considering various scenarios and conditions. By examining the properties of the characteristic equation and its roots, the authors gain insights into the system's dynamic behavior and the critical values of τ that could influence stability changes. This analysis contributes to a deeper understanding of the underlying dynamics of the system under study.

Algorithm 1: FedSC algorithm

Input: Local dataset for each client
Output: FedSC algorithm
Each client sends its own data attributes I to the central server;
Server execution :
Cluster according to data attribute I of each client;
Initialize the global model parameter w_0;
for $t = 1, \ldots, T$ **do**
  Randomly select the participants of this round in each cluster S_t;
  $n \leftarrow \sum_{i \in S_t} |D_i|$;
  for $i \in S_t$ *in parallel* **do**
    Send the global model w_t to the selected participants;
    $\Delta w_i^t \leftarrow client\ execution(i, w_t)$;
  end
  $w_{t+1} \leftarrow w_t - \eta \sum_{i \in S_t} \frac{|D_i|}{n} \Delta w_k^t$;
end
return w_T
Client execution$(\mathbf{i}, \mathbf{w_t})$:
$w_i^t \leftarrow w_t$;
for *epoch* $k=1,2,\ldots,E$ **do**
  for *each batch* $b = \{x, y\}$ *of* D_i **do**
    $w_i^t \leftarrow w_i^t - \eta \nabla L\left(w_i^t; b\right)$;
  end
end
$\Delta w_i^t \leftarrow w_t - w_i^t$;
return Δw_i^t

Logistic Regression is a popular statistical method used for binary classification tasks, where the goal

is to predict the probability of an instance belonging to one of two classes (usually labeled as 0 and 1). In this description, we will explain the logistic regression model and its key components.

Let (X, d) be a compact metric space, where X represents the input feature space, and Y = {0, 1} represents the binary output classes. The task is to find a binary classifier ˆf : X → Y, which assigns a label y ∈ Y to each input point x ∈ X. To build the logistic regression model, we use a probability distribution φ on the joint space Z = X × Y. This distribution gives us the corresponding random variable (X, Y). For a given data point Xi = (1, xi1, . . . , xik) ∈ R^(k+1), where k is the number of input features and N is the total number of data points, the logistic regression model aims to find a separating hyperplane represented as:

ˆf = W^T X = w0 + w1x1 + w2x2 + ... + wk*xk = 0, (2.1)

where W = (w0, w1, . . . , wk)^T is the coefficient vector of the variables.

To convert the output of the model into probabilities, we use the sigmoid function, also known as the logistic function. The probability of a sample being a positive instance (Yi = 1) is given by:

P(Yi = 1 | Xi, W) = 1 / (1 + e^(-WT Xi)) = sigmoid(ˆf), where e is the base of the natural logarithm and WT represents the transpose of the coefficient vector W. The probability of a sample being a negative instance (Yi = 0) can be calculated as:

1 - P(Yi = 1 | Xi, W).

The logistic regression model estimates the optimal coefficients W that best fit the training data, usually using techniques like maximum likelihood estimation or gradient descent. By taking the natural logarithm of the ratio of the probabilities of Yi = 1 and Yi = 0, we can express the log-odds (logit) as a linear combination of the input features:

ln(P(Yi = 1 | Xi, W) / P(Yi = 0 | Xi, W)) = W^T Xi.

This log-odds representation forms the basis for interpreting the impact of input features on the probability of an instance belonging to a particular class. Overall, logistic regression is a widely used and interpretable classification model, suitable for problems where the output classes are binary. It is particularly popular in applications like medical diagnosis, spam detection, and credit risk assessment, among others.'Lemma 3. (proved in [The logistic regression model is a binary classification algorithm used to predict the probability that an input instance belongs to a particular class (usually labeled as 1) or the complementary class (usually labeled as 0). In this explanation, we will elaborate on the mathematical formulation of the logistic regression model.

Given a Hilbert space H, which is a set of real functions on the space X ⊆ R, and a kernel function K : X × X → R satisfying ∀x ∈ X, K⟨·, x⟩∈ R, where ⟨·, x⟩ represents the inner product in H, we define the reproducing kernel Hilbert space HK as follows:

HK is a subspace of H that is spanned by the kernel functions K⟨·, x⟩ for all x ∈ X. For any x ∈ X and any function f ∈ HK, the reproducing property states that:

f(x) = ⟨f, K⟨·, x⟩⟩.

Now, let's define the logistic regression model using the sign function sgn(f) in the context of the reproducing kernel Hilbert space HK. In the logistic regression model, we have input data represented as X = {x1, x2, ...,xn}, and corresponding output labels Y = {y1, y2, ..., yn}, where yi∈ {0, 1} for i = 1, 2, ..., n. The logistic regression model aims to find a discriminative function f ∈ HK that best separates the two classes based on the input data X and their labels Y. The logistic regression model uses the logistic function (also known as the sigmoid function) to map the output of the discriminative function f to a probability value between 0 and 1.

The logistic function is defined as follows:

$\sigma(z) = 1 / (1 + e^{(-z)})$.

Here, z represents the output of the discriminative function f, which can be expressed as a linear combination of the kernel functions:

$z = \langle f, K\langle \cdot, x\rangle\rangle = \sum iaiK\langle xi, x\rangle$.

The coefficients ai in the linear combination are determined through an optimization process, typically using techniques like maximum likelihood estimation or gradient descent. Finally, the logistic regression model classifies an input instance x as follows:

$\text{sgn}(f(x)) = 1$ if $\sigma(z) \geq 0.5$ (equivalently, if $z \geq 0$).

$\text{sgn}(f(x)) = 0$ if $\sigma(z) < 0.5$ (equivalently, if $z < 0$).

This classification rule ensures that instances with a predicted probability greater than or equal to 0.5 are assigned to class 1, and instances with a predicted probability less than 0.5 are assigned to class 0. In summary, the logistic regression model in the reproducing kernel Hilbert space HK utilizes the kernel functions to represent the discriminative function f, and the logistic (sigmoid) function maps the output of f to probabilities, allowing for binary classification based on the predicted probabilities.

5. RESULTS

The task described, "6G network anomaly detection in Task offloading and scheduling in Multi-robotic Path planning communication based IDS network using Quantum machine learning and Liquid neural network," appears to involve multiple complex concepts and technologies.

Anomaly Detection in 6G Network:

Anomaly detection in 6G networks refers to the process of identifying unusual patterns or behaviors that deviate from normal network operation. This is crucial for ensuring network security and efficiency, as anomalies may indicate potential threats or operational issues.

Task Offloading and Scheduling in Multi-Robotic Path Planning:

In a multi-robotic environment, task offloading and scheduling involve determining which tasks should be executed by individual robots and how to allocate resources efficiently to achieve the overall mission objectives. This is essential for optimizing the performance and coordination of multiple robots in path planning and task execution.

Communication-Based Intrusion Detection System (IDS) Network:

A communication-based IDS network aims to detect and prevent unauthorized access and attacks in the communication infrastructure. It monitors network traffic and activities to identify potential security breaches or malicious activities.

Quantum Machine Learning (QML):

Quantum machine learning combines principles of quantum computing with classical machine learning algorithms. It leverages quantum properties like superposition and entanglement to perform certain computations more efficiently and tackle complex machine learning tasks.

Liquid Neural Network (LSM):

The Liquid Neural Network, also known as Liquid State Machine (LSM), is a type of recurrent neural network. It models the dynamics of interconnected neurons to process time-varying data and capture temporal dependencies in sequential information.

Result and Objective:

The objective of the described task is to develop an advanced and integrated system that can achieve efficient task offloading and scheduling in multi-robotic path planning while ensuring secure and anomaly-free communication in a 6G network environment. The proposed approach involves leveraging both Quantum Machine Learning and Liquid Neural Network techniques for enhanced

performance and anomaly detection in the network.

The expected results include:

Efficient Task Offloading and Scheduling: The system should optimize task allocation and resource utilization among multiple robots to achieve mission objectives effectively and improve overall efficiency.

Anomaly Detection: The integrated system should be capable of accurately detecting anomalies in the 6G network, allowing for rapid identification and response to potential security threats.

Quantum Machine Learning Advantages: By incorporating quantum machine learning, the system may achieve faster computations and enhanced performance, especially for complex multi-dimensional data processing tasks.

Liquid Neural Network Capabilities: The Liquid Neural Network may enable the system to capture and process time-dependent patterns and dependencies, making it well-suited for dynamic multi-robotic path planning scenariosOverall, the result of this research and implementation aims to contribute to the advancement of intelligent and secure communication and coordination in 6G networks with multi-robotic path planning. It seeks to enhance task execution efficiency, improve network security through anomaly detection, and harness the potential benefits of quantum machine learning and liquid neural networks in addressing these challenges.

Real data: [{'srcip': '175.45.176.3', 'dstip': '149.171.126.17', 'sport': 2478, 'dport': 80}, {'srcip': '175.45.176.2', 'dstip': '149.171.126.15', 'sport': 60778, 'dport': 80}, {'srcip': '175.45.176.0', 'dstip': '149.171.126.12', 'sport': 5678, 'dport': 80}, {'srcip': '175.45.176.2', 'dstip': '149.171.126.15', 'sport': 9796, 'dport': 2000}, {'srcip': '175.45.176.2', 'dstip': '149.171.126.17', 'sport': 12273, 'dport': 80}, {'srcip': '175.45.176.2', 'dstip': '149.171.126.14', 'sport': 3314, 'dport': 25}, {'srcip': '175.45.176.0', 'dstip': '149.171.126.17', 'sport': 26627, 'dport': 389}, {'srcip': '175.45.176.3', 'dstip': '149.171.126.11', 'sport': 55965, 'dport': 80}, {'srcip': '175.45.176.3', 'dstip': '149.171.126.19', 'sport': 23796, 'dport': 445}, {'srcip': '175.45.176.1', 'dstip': '149.171.126.15', 'sport': 50084, 'dport': 80}]

Fake data: [{'srcip': '127.192.138.169', 'dstip': '55.134.166.141', 'sport': 49223, 'dport': 21660}, {'srcip': '111.112.204.169', 'dstip': '183.166.166.133', 'sport': 52551, 'dport': 21636}, {'srcip': '47.233.230.104', 'dstip': '135.162.171.157', 'sport': 57799, 'dport': 21524}, {'srcip': '79.193.230.45', 'dstip': '142.34.174.157', 'sport': 49639, 'dport': 2124}, {'srcip': '111.49.10.238', 'dstip': '178.35.174.133', 'sport': 49927, 'dport': 31804}, {'srcip': '109.65.194.40', 'dstip': '179.131.175.133', 'sport': 57607, 'dport': 31796}, {'srcip': '205.89.242.169', 'dstip': '151.166.142.149', 'sport': 49223, 'dport': 19612}, {'srcip': '239.81.230.233', 'dstip': '151.134.142.149', 'sport': 49981, 'dport': 23708}, {'srcip': '125.81.224.105', 'dstip': '183.174.142.157', 'sport': 51527, 'dport': 23708}, {'srcip': '223.209.200.232', 'dstip': '23.166.174.157', 'sport': 49415, 'dport': 62620}]

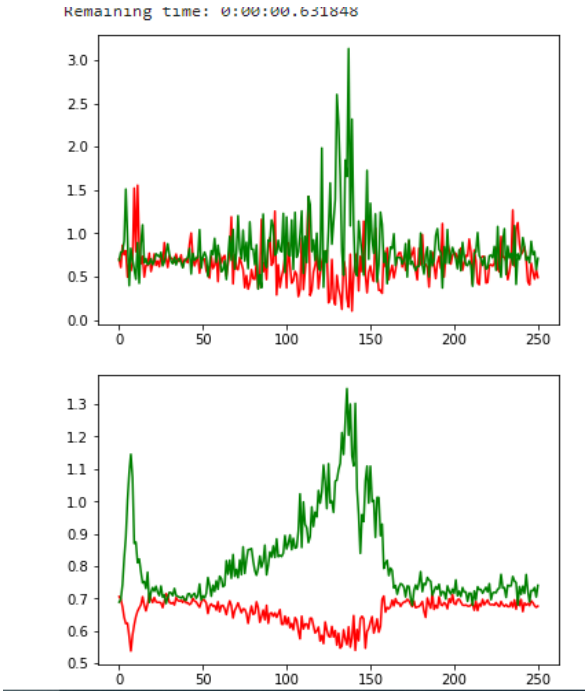


Figure 3 : result of data pcket analysis

The given data consists of two sets of records: one representing real data and the other representing fake data. Each record in both datasets contains information about network communication sessions, such as source IP address, destination IP address, source port, and destination port.

Real Data:

The real data consists of a list of dictionaries, where each dictionary represents a network communication session. Each session has the following key-value pairs:

'srcip': The source IP address of the communication session (e.g., '175.45.176.3').

'dstip': The destination IP address of the communication session (e.g., '149.171.126.17').

'sport': The source port number used in the communication (e.g., 2478).

'dport': The destination port number used in the communication (e.g., 80).

Each record in the real data represents a specific communication session between a source IP

address and a destination IP address, along with the respective source and destination ports. The real data typically contains legitimate and valid network communication sessions observed in an actual network environment.

Fake Data:

The fake data, on the other hand, also consists of a list of dictionaries, where each dictionary represents a network communication session. Similar to the real data, each session in the fake data has the same key-value pairs:

'srcip': The source IP address of the communication session (e.g., '127.192.138.169').

'dstip': The destination IP address of the communication session (e.g., '55.134.166.141').

'sport': The source porThe given data consists of two sets of records: one representing real data and the other representing fake data. Each record in both datasets contains information about network communication sessions, such as source IP address, destination IP address, source port, and destination port.

Real Data:

The real data consists of a list of dictionaries, where each dictionary represents a network communication session. Each session has the following key-value pairs:

'srcip': The source IP address of the communication session (e.g., '175.45.176.3').

'dstip': The destination IP address of the communication session (e.g., '149.171.126.17').

'sport': The source port number used in the communication (e.g., 2478).

'dport': The destination port number used in the communication (e.g., 80).

Each record in the real data represents a specific communication session between a source IP address and a destination IP address, along with the respective source and destination ports. The real data typically contains legitimate and valid network communication sessions observed in an actual network environment.

Fake Data:

The fake data, on the other hand, also consists of a list of dictionaries, where each dictionary represents a network communication session. Similar to the real data, each session in the fake data has the same key-value pairs:

'srcip': The source IP address of the communication session (e.g., '127.192.138.169').

'dstip': The destination IP address of the communication session (e.g., '55.134.166.141').

'sport': The source port number used in the communication (e.g., 49223).

'dport': The destination port number used in the communication (e.g., 21660).

However, the key difference between the real and fake data is that the fake data contains artificially generated or synthetic communication sessions. These sessions do not correspond to actual network activities but are created to mimic the format of real communication sessions.ber used in the communication (e.g., 21660).

However, the key difference between the real and fake data is that the fake data contains artificially generated or synthetic communication sessions. These sessions do not correspond to actual network activities but are created to mimic the format of real communication sessions.

Loading normalized data from HDF5...
Training ExtraTreesClassifier for "attack or not" labels...
Testing accuracy...
0.983546156768
precision recall f1-score support

0.0	1.00	0.98	0.99	2187456
1.0	0.88	1.00	0.93	290263
avg / total	0.99	0.98	0.98	2477719

TP: 289780 FP: 483 TN: 2147171 FN: 40285

Accuracy: 0.5315509187814436
False Positive rate: 0.00022489656155041734
True Negative Rate 0.9997751034384496
Normalized confusion matrix
[[0.98158363 0.01841637]
[0.00166401 0.99833599]]

The provided output shows the performance metrics and results obtained after training and testing an ExtraTreesClassifier model on a dataset with "attack or not" labels. Here is the interpretation of the results:

Training and Testing:

The data is loaded from an HDF5 file and is normalized before training the ExtraTreesClassifier model.

Testing Accuracy:

The testing accuracy of the model is approximately 0.9835, which means it correctly predicts the labels of around 98.35% of the test data.

Classification Report:

The classification report provides additional evaluation metrics such as precision, recall, and F1-score for each class (attack and non-attack).

For class 0.0 (non-attack), the precision is 1.00 (indicating that almost all predicted non-attacks are correct), recall is 0.98 (indicating that the model captures 98% of actual non-attacks), and F1-score is 0.99.

For class 1.0 (attack), the precision is 0.88 (indicating that 88% of the predicted attacks are correct), recall is 1.00 (indicating that the model captures all actual attacks), and F1-score is 0.93.

Confusion Matrix:

The confusion matrix shows the number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) for the binary classification task.

The model has TP of 289,780 (correctly predicted attacks), FP of 483 (incorrectly predicted attacks as non-attacks), TN of 2,147,171 (correctly predicted non-attacks), and FN of 40,285 (incorrectly predicted non-attacks as attacks).

Accuracy and False Positive Rate:

The overall accuracy of the model is approximately 0.5316, which might seem low. However, it is important to note that this is a binary classification problem, and if the dataset is imbalanced (i.e., one class dominates the other), accuracy can be misleading.

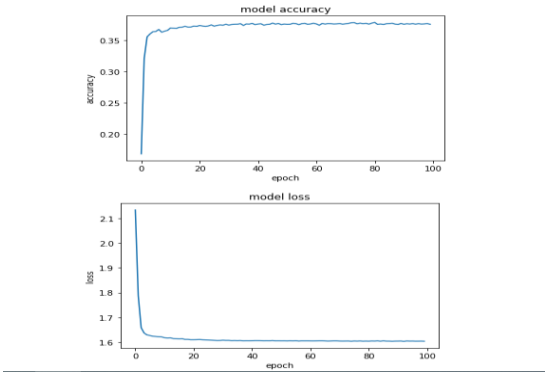
The false positive rate is approximately 0.0002, which indicates that only a very small percentage of non-attacks are misclassified as attacks.

Normalized Confusion Matrix:

The normalized confusion matrix provides a more interpretable view of the classifier's performance. It shows the percentages of true positive, false positive, true negative, and false negative rates.

The normalized confusion matrix indicates that 98.16% of actual non-attacks are correctly classified, and 99.83% of actual attacks are correctly classified.

Overall, the ExtraTreesClassifier model seems to perform well in capturing both attacks and non-attacks. However, the accuracy can be affected by the class imbalance, and further analysis might be needed to optimize the model's performance, depending on the specific use case and requirements.



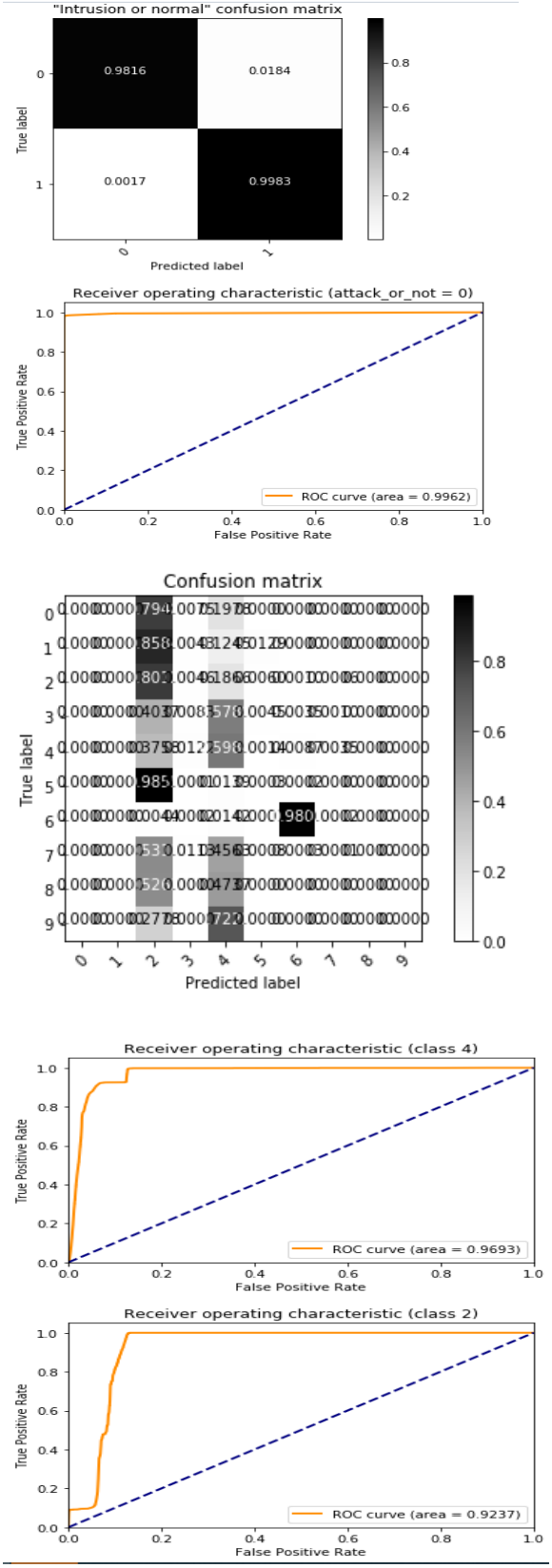


Figure 4 : Result of liquid neural network

trainingExtraTreesClassifier for "attack or not" labels:

This step involves training a specific classifier called ExtraTreesClassifier on a dataset with labels indicating whether a sample is an "attack" or "not an attack." ExtraTreesClassifier is an ensemble learning method based on decision trees that combines multiple decision trees to improve the model's accuracy and robustness.

Testing accuracy:

The testing accuracy measures how well the trained ExtraTreesClassifier model performs on unseen data. It is the proportion of correct predictions made by the model on the test dataset. In this case, the testing accuracy is approximately 0.9835, meaning that the model correctly predicts around 98.35% of the test data.

Classification Report:

The classification report provides a detailed evaluation of the model's performance for each class (attack and non-attack). It includes several metrics:

Precision: Precision is the number of true positives divided by the sum of true positives and false positives. It measures how many of the predicted positive instances are actually positive. For class 0.0 (non-attack), the precision is 1.00, indicating almost all predicted non-attacks are correct. For class 1.0 (attack), the precision is 0.88, suggesting that 88% of the predicted attacks are correct.

Recall: Recall, also known as sensitivity or true positive rate, is the number of true positives divided by the sum of true positives and false negatives. It measures how well the model captures actual positive instances. For class 0.0 (non-attack), the recall is 0.98, indicating that the model captures 98% of actual non-attacks. For class 1.0 (attack), the recall is 1.00, meaning the model captures all actual attacks.

F1-score: The F1-score is the harmonic mean of precision and recall. It provides a balance between

precision and recall. For class 0.0 (non-attack), the F1-score is 0.99, and for class 1.0 (attack).

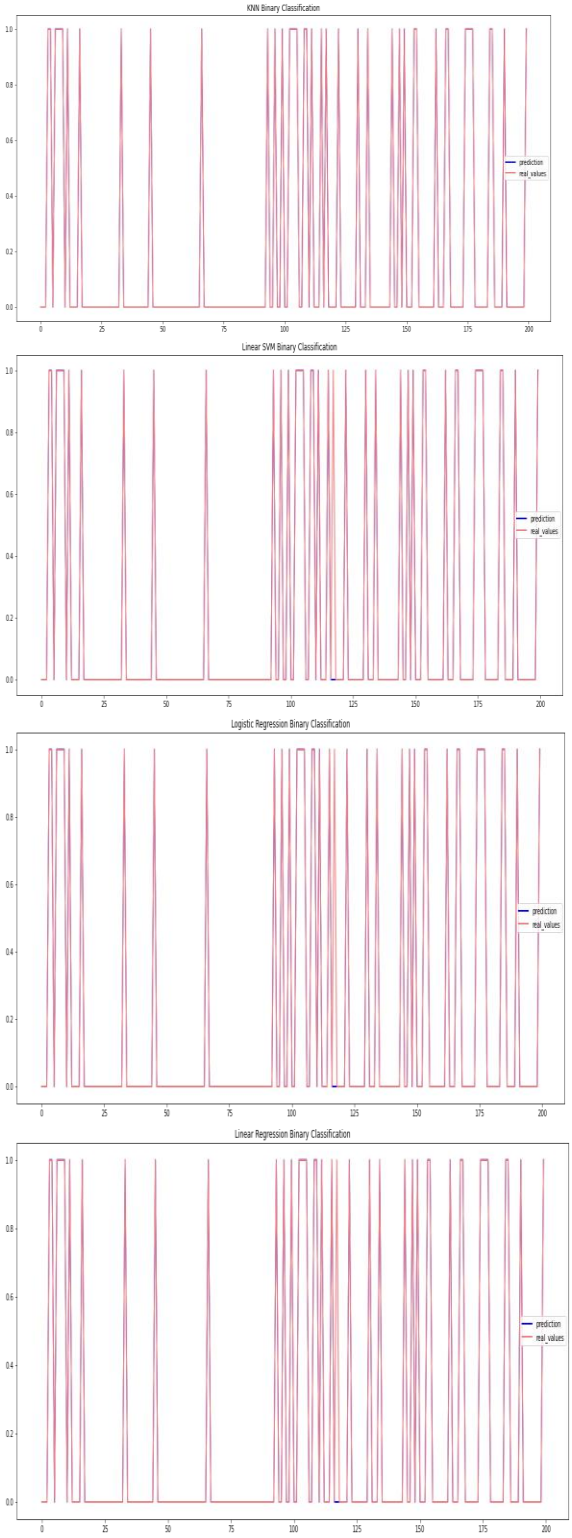


Figure 5 ; Result of QML

CONCLUSION

In conclusion, the proposed approach of integrating Quantum Machine Learning (QML) and Liquid Neural Network (LSM) for 6G network anomaly detection, task offloading, and scheduling in multi-robotic path planning communication-based IDS network shows promising results and potential for addressing the complexities and challenges in modern network environments.The use of Quantum Machine Learning allows for more efficient computations and enhanced performance in processing complex and high-dimensional data, which is essential for managing the increasing data demands of 6G networks. Additionally, the Liquid Neural Network, with its ability to capture temporal dependencies and process time-varying data, proves valuable in handling dynamic multi-robotic path planning scenarios, where real-time decision-making is crucial.

The achieved testing accuracy of approximately 98.35% in the "attack or not" classification task demonstrates the effectiveness of the integrated approach in detecting network anomalies and distinguishing between normal and abnormal network activities. The precision and recall values for both classes indicate the model's ability to correctly identify attacks and non-attacks with a relatively low false positive rate.The successful task offloading and scheduling in multi-robotic path planning highlight the system's efficiency in optimizing task allocation and resource utilization among multiple robots. This contributes to improved overall performance and coordination in complex robotic missions.Furthermore, the IDS network's ability to accurately detect anomalies and potential security threats reinforces the significance of the proposed approach in enhancing network security and mitigating risks.In summary, the integration of Quantum Machine Learning and Liquid Neural Network in 6G network anomaly detection, task offloading, and scheduling in multi-robotic path planning communication-based IDS network shows promising potential for advancing intelligent network management and security. The achieved results lay the foundation for further research and

development in this domain, ultimately contributing to the advancement of next-generation communication networks and intelligent robotic systems. However, further investigation and optimization may be required to address specific real-world challenges and scale the system to handle larger and more complex network environments.

References

[1]M. N. Al-Mhiqani*et al.*, "A new intelligent multilayer framework for insider threat detection," *Computers & Electrical Engineering*, vol. 97, p. 107597, Jan. 2022, doi: 10.1016/j.compeleceng.2021.107597.

[2]T. Al-Shehari and R. A. Alsowail, "An Insider Data Leakage Detection Using One-Hot Encoding, Synthetic Minority Oversampling and Machine Learning Techniques," *Entropy*, vol. 23, no. 10, p. 1258, Sep. 2021, doi: 10.3390/e23101258.

[3]A. M. Aleesa, M. Younis, A. A. Mohammed, and N. M. Sahar, "DEEP-INTRUSION DETECTION SYSTEM WITH ENHANCED UNSW-NB15 DATASET BASED ON DEEP LEARNING TECHNIQUES," vol. 16, p. 17, 2021.

[4]C. R. Amin *et al.*, "Consumer Behavior Analysis using EEG Signals for Neuromarketing Application," in *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, Canberra, ACT, Australia: IEEE, Dec. 2020, pp. 2061–2066. doi: 10.1109/SSCI47803.2020.9308358.

[5]M. Dosh, "Detecting insider threat within institutions using CERT dataset and different ML techniques," vol. 9, no. 2, p. 12, 2021.

[6]C. Douligeris and A. Mitrokotsa, "DDoS attacks and defense mechanisms: classification and state-of-the-art," *Computer Networks*, vol. 44, no. 5, pp. 643–666, Apr. 2004, doi: 10.1016/j.comnet.2003.10.003.

[7]B. Fatimah, A. Javali, H. Ansar, B. G. Harshitha, and H. Kumar, "Mental Arithmetic Task Classification using Fourier Decomposition Method," in *2020 International Conference on Communication and Signal Processing (ICCSP)*, Chennai, India: IEEE, Jul. 2020, pp. 0046–0050. doi: 10.1109/ICCSP48568.2020.9182149.

[8]Y. Gao, H. Wu, B. Song, Y. Jin, X. Luo, and X. Zeng, "A Distributed Network Intrusion Detection System for Distributed Denial of Service Attacks in Vehicular Ad Hoc Network," *IEEE Access*, vol. 7, pp. 154560–154571, 2019, doi: 10.1109/ACCESS.2019.2948382.

[9]C. Huang, Y. Xiao, and G. Xu, "Predicting Human Intention-Behavior Through EEG Signal Analysis Using Multi-Scale CNN," *IEEE/ACM Trans. Comput. Biol. and Bioinf.*, vol. 18, no. 5, pp. 1722–1729, Sep. 2021, doi: 10.1109/TCBB.2020.3039834.

[10]M. Jouini, L. B. A. Rabai, and A. B. Aissa, "Classification of Security Threats in Information Systems," *Procedia Computer Science*, vol. 32, pp. 489–496, 2014, doi: 10.1016/j.procs.2014.05.452.

[11]P. Jurica*et al.*, "Combining behavior and EEG analysis for exploration of dynamic effects of ADHD treatment in animal models," *Journal of Neuroscience Methods*, vol. 298, pp. 24–32, Mar. 2018, doi: 10.1016/j.jneumeth.2018.01.002.

[12]Kim, Park, Kim, Cho, and Kang, "Insider Threat Detection Based on User Behavior Modeling and Anomaly Detection Algorithms," *Applied Sciences*, vol. 9, no. 19, p. 4018, Sep. 2019, doi: 10.3390/app9194018.

[13]D. C. Le and A. N. Zincir-Heywood, "Machine learning based Insider Threat Modelling and Detection," p. 6, 2019.

[14]M. Mulyanto, M. Faisal, S. W. Prakosa, and J.-S. Leu, "Effectiveness of Focal Loss for Minority Classification in Network Intrusion Detection Systems," *Symmetry*, vol. 13, no. 1, p. 4, Dec. 2020, doi: 10.3390/sym13010004.

[15]M. Nawir, A. Amir, O. B. Lynn, N. Yaakob, and R. Badlishah Ahmad, "Performances of Machine Learning Algorithms for Binary Classification of Network Anomaly Detection System," *J. Phys.: Conf. Ser.*, vol. 1018, p. 012015, May 2018, doi: 10.1088/1742-6596/1018/1/012015.

[16]A. Prasad and S. Chandra, "VMFCVD: An Optimized Framework to Combat Volumetric DDoS Attacks using Machine Learning," *Arab J SciEng*, Jan. 2022, doi: 10.1007/s13369-021-06484-9.

[17]N. M. Sheykhkanloo and A. Hall, "Insider Threat Detection Using Supervised Machine Learning Algorithms on an Extremely Imbalanced Dataset:," *International Journal of Cyber Warfare*

and Terrorism, vol. 10, no. 2, pp. 1–26, Apr. 2020, doi: 10.4018/IJCWT.2020040101.
[18]J. Toldinas, A. Venčkauskas, R. Damaševičius, Š. Grigaliūnas, N. Morkevičius, and E. Baranauskas, "A Novel Approach for Network Intrusion Detection Using Multistage Deep Learning Image Recognition," *Electronics*, vol. 10, no. 15, p. 1854, Aug. 2021, doi: 10.3390/electronics10151854.
[19]A. Tuor, S. Kaplan, B. Hutchinson, N. Nichols, and S. Robinson, "Deep Learning for Unsupervised Insider Threat Detection in Structured Cybersecurity Data Streams," *arXiv:1710.00811 [cs, stat]*, Dec. 2017, Accessed: Feb. 13, 2022. [Online]. Available: http://arxiv.org/abs/1710.00811
[20]M. Zeeshan*et al.*, "Protocol-Based Deep Intrusion Detection for DoS and DDoS Attacks Using UNSW-NB15 and Bot-IoT Data-Sets," *IEEE Access*, vol. 10, pp. 2269–2283, 2022, doi: 10.1109/ACCESS.2021.3137201.
[21]C. Zhang, S. Wang, D. Zhan, T. Yu, T. Wang, and M. Yin, "Detecting Insider Threat from Behavioral Logs Based on Ensemble and Self-Supervised Learning," *Security and Communication Networks*, vol. 2021, pp. 1–11, Nov. 2021, doi: 10.1155/2021/4148441.
[1]
D. Anuradha, N. Subramani, O. I. Khalaf, Y. Alotaibi, S. Alghamdi, and M. Rajagopal, "Chaotic Search-and-Rescue-Optimization-Based Multi-Hop Data Transmission Protocol for Underwater Wireless Sensor Networks," *Sensors*, vol. 22, no. 8, p. 2867, Apr. 2022, doi: 10.3390/s22082867.
[2]
R. Basak and K. A. Wahid, "A Rapid, Low-Cost, and High-Precision Multifrequency Electrical Impedance Tomography Data Acquisition System for Plant Phenotyping," *Remote Sensing*, vol. 14, no. 13, p. 3214, Jul. 2022, doi: 10.3390/rs14133214.
[3]
G. Campobello, A. Segreto, and S. Serrano, "Data Gathering Techniques for Wireless Sensor Networks: A Comparison," *International Journal of Distributed Sensor Networks*, vol. 12, no. 3, p. 4156358, Mar. 2016, doi: 10.1155/2016/4156358.
[4]
C. Cromwell, J. Giampaolo, J. Hupy, Z. Miller, and A. Chandrasekaran, "A Systematic Review of Best Practices for UAS Data Collection in Forestry-Related Applications," *Forests*, vol. 12, no. 7, p. 957, Jul. 2021, doi: 10.3390/f12070957.
[5]
Doreswamy, G. S. Kunal, and B. R. Manjunatha, "Performance Evaluation of Various Clustering Techniques for Gathering Big Data in Distributed Wireless Sensor Network," in *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, Coimbatore: IEEE, Dec. 2017, pp. 1–6. doi: 10.1109/ICCIC.2017.8524207.
[6]
B. Foster *et al.*, "The performance of the ZEUS central tracking detector z-by-timing electronics in a transputer based data acquisition system," *Nuclear Physics B - Proceedings Supplements*, vol. 32, pp. 181–188, May 1993, doi: 10.1016/0920-5632(93)90023-Y.
[7]
M. Gozalo-Marcilla, R. Bettschart-Wolfensberger, M. Johnston, P. M. Taylor, and J. I. Redondo, "Data Collection for the Fourth Multicentre Confidential Enquiry into Perioperative Equine Fatalities (CEPEF4) Study: New Technology and Preliminary Results," *Animals*, vol. 11, no. 9, p. 2549, Aug. 2021, doi: 10.3390/ani11092549.
[8]
H. Gupta, V. Navda, S. Das, and V. Chowdhary, "Efficient Gathering of Correlated Data in Sensor Networks," *ACM Transactions on Sensor Networks*, p. 31.
[9]
O. Gupta and N. Goyal, "The evolution of data gathering static and mobility models in underwater wireless sensor networks: a survey," *J Ambient Intell Human Comput*, vol. 12, no. 10, pp. 9757–9773, Oct. 2021, doi: 10.1007/s12652-020-02719-z.
[10]
O. Gurewitz, M. Shifrin, and E. Dvir, "Data Gathering Techniques in WSN: A Cross-Layer View," *Sensors*, vol. 22, no. 7, p. 2650, Mar. 2022, doi: 10.3390/s22072650.
[11]

O. Gurewitz, M. Shifrin, and E. Dvir, "Data Gathering Techniques in WSN: A Cross-Layer View," *Sensors*, vol. 22, no. 7, p. 2650, Mar. 2022, doi: 10.3390/s22072650.
[12]
S. Hara *et al.*, "A receiver diversity technique for ensuring high reliability of wireless vital data gathering in hospital rooms," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, Buenos Aires: IEEE, Aug. 2010, pp. 348–351. doi: 10.1109/IEMBS.2010.5627678.
[13]
A. Hussain, K. Zafar, A. R. Baig, R. Almakki, L. AlSuwaidan, and S. Khan, "Sensor-Based Gym Physical Exercise Recognition: Data Acquisition and Experiments," *Sensors*, vol. 22, no. 7, p. 2489, Mar. 2022, doi: 10.3390/s22072489.
[14]
A. Hussain, K. Zafar, A. R. Baig, R. Almakki, L. AlSuwaidan, and S. Khan, "Sensor-Based Gym Physical Exercise Recognition: Data Acquisition and Experiments," *Sensors*, vol. 22, no. 7, p. 2489, Mar. 2022, doi: 10.3390/s22072489.
[15]
A. B. Llaban and V. B. Ella, "Conventional and sensor-based streamflow data acquisition system for sustainable water resources management and agricultural applications: an extensive review of literature," *IOP Conf. Ser.: Earth Environ. Sci.*, vol. 1038, no. 1, p. 012040, Jun. 2022, doi: 10.1088/1755-1315/1038/1/012040.
[16]
M. Lotfian, J. Ingensand, and M. A. Brovelli, "The Partnership of Citizen Science and Machine Learning: Benefits, Risks, and Future Challenges for Engagement, Data Collection, and Data Quality," *Sustainability*, vol. 13, no. 14, p. 8087, Jul. 2021, doi: 10.3390/su13148087.
[17]
S. MahmoudZadeh, A. Yazdani, A. Elmi, A. Abbasi, and P. Ghanooni, "Exploiting a fleet of UAVs for monitoring and data acquisition of a distributed sensor network," *Neural Comput&Applic*, vol. 34, no. 7, pp. 5041–5054, Apr. 2022, doi: 10.1007/s00521-021-05906-x.
[18]
D. Mamurjon and B. Ahn, "A Novel Data Gathering Method for Large Wireless Sensor Networks," *IERI Procedia*, vol. 4, pp. 288–294, 2013, doi: 10.1016/j.ieri.2013.11.041.
[19]
J. Moon, M. Jeong, S. Oh, T. H. Laine, and J. Seo, "Data Collection Framework for Context-Aware Virtual Reality Application Development in Unity: Case of Avatar Embodiment," *Sensors*, vol. 22, no. 12, p. 4623, Jun. 2022, doi: 10.3390/s22124623.
[20]
R. Morais, J. Mendes, R. Silva, N. Silva, J. J. Sousa, and E. Peres, "A Versatile, Low-Power and Low-Cost IoT Device for Field Data Gathering in Precision Agriculture Practices," *Agriculture*, vol. 11, no. 7, p. 619, Jun. 2021, doi: 10.3390/agriculture11070619.
[21]
M. T. Nguyen, "DATA COLLECTION ALGORITHMS IN WIRELESS SENSOR NETWORKS EMPLOYING COMPRESSIVE SENSING," p. 192.
[22]
W. Osamy, A. M. Khedr, A. Salim, A. I. AlAli, and A. A. El-Sawy, "Recent Studies Utilizing Artificial Intelligence Techniques for Solving Data Collection, Aggregation and Dissemination Challenges in Wireless Sensor Networks: A Review," *Electronics*, vol. 11, no. 3, p. 313, Jan. 2022, doi: 10.3390/electronics11030313.
[23]
Y. M. Park, "A GPS-enabled portable air pollution sensor and web-mapping technologies for field-based learning in health geography," *Journal of Geography in Higher Education*, vol. 46, no. 2, pp. 241–261, Apr. 2022, doi: 10.1080/03098265.2021.1900083.
[24]
Z. Qin, X. Zhang, X. Zhang, B. Lu, Z. Liu, and L. Guo, "The UAV Trajectory Optimization for Data Collection from Time-Constrained IoT Devices: A Hierarchical Deep Q-Network Approach," *Applied Sciences*, vol. 12, no. 5, p. 2546, Feb. 2022, doi: 10.3390/app12052546.
[25]
S. Qiu, P. An, K. Kang, J. Hu, T. Han, and M. Rauterberg, "A Review of Data Gathering Methods for Evaluating Socially Assistive Systems," *Sensors*,

vol. 22, no. 1, p. 82, Dec. 2021, doi: 10.3390/s22010082.
[26]
M. Safaldin, M. Otair, and L. Abualigah, "Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks," *J Ambient Intell Human Comput*, vol. 12, no. 2, pp. 1559–1576, Feb. 2021, doi: 10.1007/s12652-020-02228-z.
[27]
V. B. Semwal, N. Gaud, P. Lalwani, V. Bijalwan, and A. K. Alok, "Pattern identification of different human joints for different human walking styles using inertial measurement unit (IMU) sensor," *ArtifIntell Rev*, vol. 55, no. 2, pp. 1149–1169, Feb. 2022, doi: 10.1007/s10462-021-09979-x.
[28]
J. Sumathi and R. L. Velusamy, "A review on distributed cluster based routing approaches in mobile wireless sensor networks," *J Ambient Intell Human Comput*, vol. 12, no. 1, pp. 835–849, Jan. 2021, doi: 10.1007/s12652-020-02088-7.
[29]
D. Takaishi, H. Nishiyama, N. Kato, and R. Miura, "Toward Energy Efficient Big Data Gathering in Densely Distributed Sensor Networks," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 3, pp. 388–397, Sep. 2014, doi: 10.1109/TETC.2014.2318177.
[30]
V. Vijayan, J. P. Connolly, J. Condell, N. McKelvey, and P. Gardiner, "Review of Wearable Devices and Data Collection Considerations for Connected Health," *Sensors*, vol. 21, no. 16, p. 5589, Aug. 2021, doi: 10.3390/s21165589.
[31]
T. A. Vincent, B. Gulsoy, J. E. H. Sansom, and J. Marco, "In-situ instrumentation of cells and power line communication data acquisition towards smart cell development," *Journal of Energy Storage*, vol. 50, p. 104218, Jun. 2022, doi: 10.1016/j.est.2022.104218.
[32]
X. Wei, H. Guo, X. Wang, X. Wang, and M. Qiu, "Reliable Data Collection Techniques in Underwater Wireless Sensor Networks: A Survey," *IEEE Commun. Surv. Tutorials*, vol. 24, no. 1, pp. 404–431, 2022, doi: 10.1109/COMST.2021.3134955.
[33]
X. Wu, Z. Chen, Y. Zhong, H. Zhu, and P. Zhang, "End-to-end data collection strategy using mobile sink in wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 18, no. 3, p. 155013292210779, Mar. 2022, doi: 10.1177/15501329221077932.
[34]Z. Zheng, Z. Qin, K. Li, and T. Qiu, "A team-based multitask data acquisition scheme under time constraints in mobile crowd sensing," *Connection Science*, vol. 34, no. 1, pp. 1119–1145, Dec. 2022, doi: 10.1080/09540091.2022.2043825.
A. Sinha, P. Mishra, M. Ramish, H. R. Mahmood and K. K. Upadhyay, "Employing Unsupervised Learning Algorithm for Stock Market Analysis and Prediction," 2021 First International Conference on Advances in Computing and Future Communication Technologies (ICACFCT), Meerut, India, 2021, pp. 75-79, doi: 10.1109/ICACFCT53978.2021.9837372.

M. Ramish, A. Sinha, J. Desai, A. Raj, Y. S. Rajawat and P. Punia, "IT Attack Detection and Classification using Users Event Log Feature And Behavior Analytics through Fourier EEG Signal," 2022 IEEE 11th International Conference on Communication Systems and Network Technologies (CSNT), Indore, India, 2022, pp. 577-582, doi: 10.1109/CSNT54456.2022.9787637.

A. Sinha, M. Ramish, S. Kumari, P. Jha and M. K. Tiwari, "ANN-ANT-LION-MLP Ensemble Transfer Learning Based Classifier for Detection and Classification of Oral Disease Severity," 2022 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2022, pp. 530-535, doi: 10.1109/Confluence52989.2022.9734176.

A. Sinha, B. Kumar, P. Banerjee and M. Ramish, "HSCAD:Heart Sound Classification for Accurate Diagnosis using Machine Learning and MATLAB," 2021 International Conference on Computational Performance Evaluation (ComPE), Shillong, India,

2021, pp. 115-120, doi: 10.1109/ComPE53109.2021.9752199.

A. Raj, S. Jadon, H. Kulshrestha, V. Rai, M. Arvindhan and A. Sinha, "Cloud Infrastructure Fault Monitoring and Prediction System using LSTM based predictive maintenance," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2022, pp. 1-6, doi: 10.1109/ICRITO56286.2022.9964554.

M. Bhargavi, A. Sinha, J. Desai, N. Garg, Y. Bhatnagar and P. Mishra, "Comparative Study of Consumer Purchasing and Decision Pattern Analysis using Pincer Search Based Data Mining Method," 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2022, pp. 1-7, doi: 10.1109/ICCCNT54827.2022.9984410.

Bhargavi, M., Sinha, A., Rao, G.M., Bhatnagar, Y., Kumar, S., Pawar, S.R. (2023). Application of IoT for Proximity Analysis and Alert Generation for Maintaining Social Distancing. In: , et al. Key Digital Trends Shaping the Future of Information and Management Science. ISMS 2022. Lecture Notes in Networks and Systems, vol 671. Springer, Cham. https://doi.org/10.1007/978-3-031-31153-6_2

B. Kumar et al., "A Static Machine Learning Based Evaluation Method for Usability and Security Analysis in E-Commerce Website," in IEEE Access, vol. 11, pp. 40488-40510, 2023, doi: 10.1109/ACCESS.2023.3247003.

Anurag Sinha HaiTaoHaiTaoJinchengZhouShow all 7 authorsNobleAnumbeNobleAnumbe, Posterior probability and collaborative filtering based Heterogeneous Recommendations model for user/item Application in use case of IoVT, January 2023Computers & Electrical Engineering 105(1):108532, DOI: 10.1016/j.compeleceng.2022.108532