Batch - 3

MALLA REDDY
ENGINEERING COLLEGE AND MANAGEMENT SCIENCES
(AN UGC AUTONOMOUS INSTITUTION)

Approved by AICTE , Accredited by NBA (CSE, IT, EEE, ECE)

Kistapur Village, Mediciti Road, Medchal, Hyderabad – 501401, Telangana, India

Department of CSE-Data Science

ANN(Artificial Neural Network) Model for Classification of IOT device states in Medical industry

UNDER THE GUIDANCE OF:

Mr. Sayyed Pasha Mohammad M.Tech,
Assistant Professor
Department of CSE

PRESENTED BY:

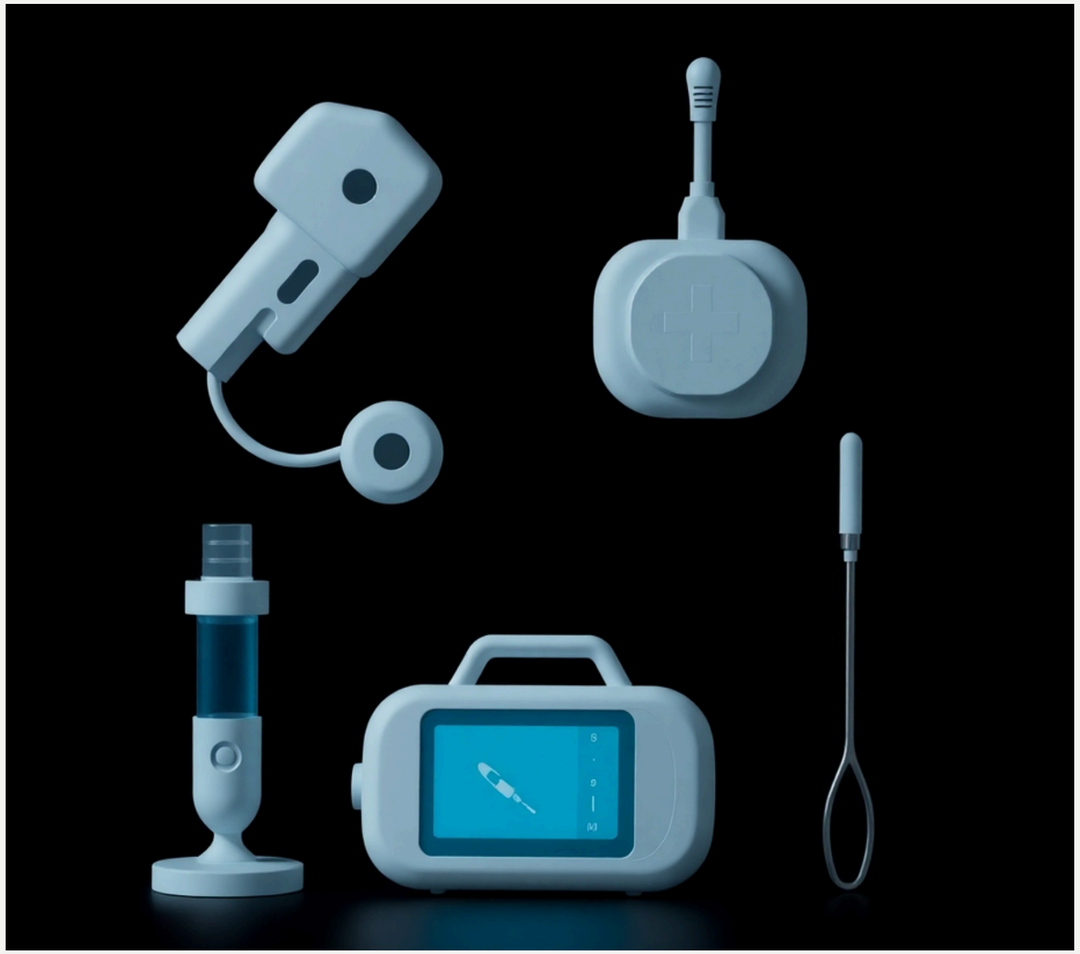
C Arya Abhirvey	21UJ1A6706
R Gokul Varma	21UJ1A6722
A Vedasri	21UJ1A6702
Y Hari Charan	21UJ1A6742

INDEX

1. TITLE
2. INDEX
3. ABSTRACT
4. INTRODUCTION
5. MOTIVATION
6. PROBLEM DEFINITION
7. LITERATURE SURVEY
8. RESEARCH GAPS
9. EXISTING SYSTEM
10. DRAWBACKS
11. PROPOSED SYSTEM
12. EXPLANATION
13. ADVANTAGES
14. SOFTWARE DETAILS
15. SAMPLE CODE
16. PACKAGES USED
17. RESULT ANALYSIS
18. ADVANTAGES
19. APPLICATIONS
20. CONCLUSION
21. FUTURE SCOPE
22. REFERENCES

ABSTRACT

The rapid adoption of IoT devices in healthcare has highlighted the need for accurate device state classification to ensure uninterrupted operations. Traditional monitoring methods are slow, error-prone, and miss transient faults, risking patient safety. This study proposes a hybrid ANN-based model to classify device states as Normal or Anomaly using real-time hospital telemetry data and open-source datasets. Preprocessing includes null removal, normalization, and time-series segmentation. While Gaussian Naïve Bayes offered baseline results, it underperformed on complex data. The proposed ANN+Extra Trees Classifier framework captures deep patterns and improves accuracy, precision, and anomaly detection, enabling faster and safer medical IoT device monitoring.



INTRODUCTION

- The Internet of Things (IoT) has brought a paradigm shift in the healthcare industry by having continuous monitoring, remote diagnostics.
- Medical IoT devices generate vast amounts of data, reflecting their operational states, which are crucial for ensuring reliability, patient safety, and timely interventions
- Traditional rule-based systems lack adaptability and often fail in real-time detection.
- This study proposes an ANN-based classification model to monitor and predict the states of IoT devices.

MOTIVATION

- Critical Role of Devices: Devices like ventilators and monitors must function flawlessly, as malfunctions can cause treatment delays or medical errors.

- Predictive Maintenance by Industry Leaders: Companies like GE Healthcare and Philips use predictive analytics and cloud-based tools to detect faults early and reduce downtime.

- Wider Application in Remote Care: Intelligent monitoring extends to elderly care and remote services, enabling real-time fault detection in wearables and home devices.

- Data-Driven Necessity: With growing device complexity, data analytics is essential for operational efficiency, patient safety, and cost-effective healthcare delivery.

PROBLEM DEFINITION

- Challenges in Dynamic Environments: Medical IoT devices operate in real-time settings but face faults and anomalies that traditional systems struggle to detect.

- Limitations of Manual Methods: Manual inspections and rule-based tools miss transient or subtle issues that degrade device performance without triggering alarms.

- Static Systems Are Inefficient: Reliance on scheduled checks or fixed thresholds ignores evolving device behavior, leading to higher risks and costs.

- Need for Adaptive Solutions: A real-time, data-driven classification method is essential to detect anomalies across diverse devices without manual rules.

LITERATURE SURVEY

No.	Authors	Title	Existing System	Proposed System	Accuracy	Key Findings
1	Akhtar, MD Mobin et al.	Big data classification in IoT healthcare using optimal deep learning	KNN, Fuzzy Classifier	DRCSO-SRN	92.90%	Improved classification accuracy
2	Lee, Jae Dong et al.	M-IDM: A Multi-Classification Based Intrusion Detection Model in Healthcare IoT	Naive Bayes, SVM	M-IDM	98.60%	Higher detection accuracy
3	Vakili, Meysam et al.	Performance comparison of machine and deep learning	RF, DT, KNN	RF, DT, KNN	98%	RF shows high consistency
4	Saif, Sohail et al.	HIIDS: Hybrid intelligent intrusion detection for IoT-based healthcare	KNN, DT	GA	95.39%	GA & DT gave more accuracy
5	Alsalman, Dheyaaldin A	Comparative study of anomaly detection techniques for IoT security	SVM	FusionNet	99%	High precision and recall
6	Khan, Maryam Mahsal et al.	Anomaly detection in IoT-based healthcare	RF, DT, KNN	SL (Advanced Processing)	85%	RF delivers more accuracy
7	Hasan, Mahmudul et al.	Attack and anomaly detection in IoT sensors	LR, SVM	DT, RF	~1–2% lower	LR, SVM gave lower results

LITERATURE SURVEY

No.	Authors	Title	Existing System	Proposed System	Accuracy	Key Findings
8	Dwivedi, Rajendra Kumar et al.	Gaussian ML for anomaly detection in sensor cloud	SVM	GDA	98%	High precision
9	Vishwakarma, Monika et al.	Two-phase IDS with Naive Bayes	IDS	2-Phase IDS	86.90%	Hybrid IDS achieved more
10	Al Abdulwahid, Abdulwahid	Detection of Middlebox-Based Attacks in Healthcare IoT	XNN	ANN	98%	Robust and balanced output
11	Souri, Alireza et al.	Healthcare monitoring for students in IoT	Scalable ML Techniques	SVM	99%	Effective prediction
12	Tiwari, Anurag et al.	Behavioral analysis in smart healthcare	Rule-Based	ANN	Error margin < 5%	Reliable monitoring
13	Mansour, Romany Fouad et al.	AI & IoT-enabled disease diagnosis model	LSTM	CSO-CLSTM	96.50%	Hybrid optimization improved accuracy
14	Woźniak, Marcin et al.	BiLSTM model for imbalanced IoT medical data	BiLSTM	Hybrid BiLSTM	88%	Combining with DT gave high results
15	Yousaf, Iqra et al.	Optimized hyperparameters of CNN	CNN	CNN-HHO	96%	Increased bug severity findings

RESEARCH GAPS

Limited Real-World Deployment:
Most existing ANN studies rely on simulated datasets and fail to validate models in real hospital or clinical environments, reducing real-world applicability.

Lack of Optimization for Edge Devices:
Current ANN models are not sufficiently optimized for medical IoT scenarios that involve low latency and limited computational resources at the edge.

Poor Explainability of ANN Decisions:
ANN-based models often lack interpretability, making them difficult to trust and adopt in critical healthcare decision-making processes.

Insufficient Integration of Domain Knowledge:
Few studies integrate medical domain expertise into ANN models, which limits their accuracy, reliability, and contextual relevance in healthcare IoT applications.

EXISTING SYSTEM

Threshold-Based Monitoring
Uses fixed upper and lower bounds to trigger alerts when values exceed limits.
Easy to implement but lacks adaptability to gradual wear, environmental changes, or patient-specific variations.

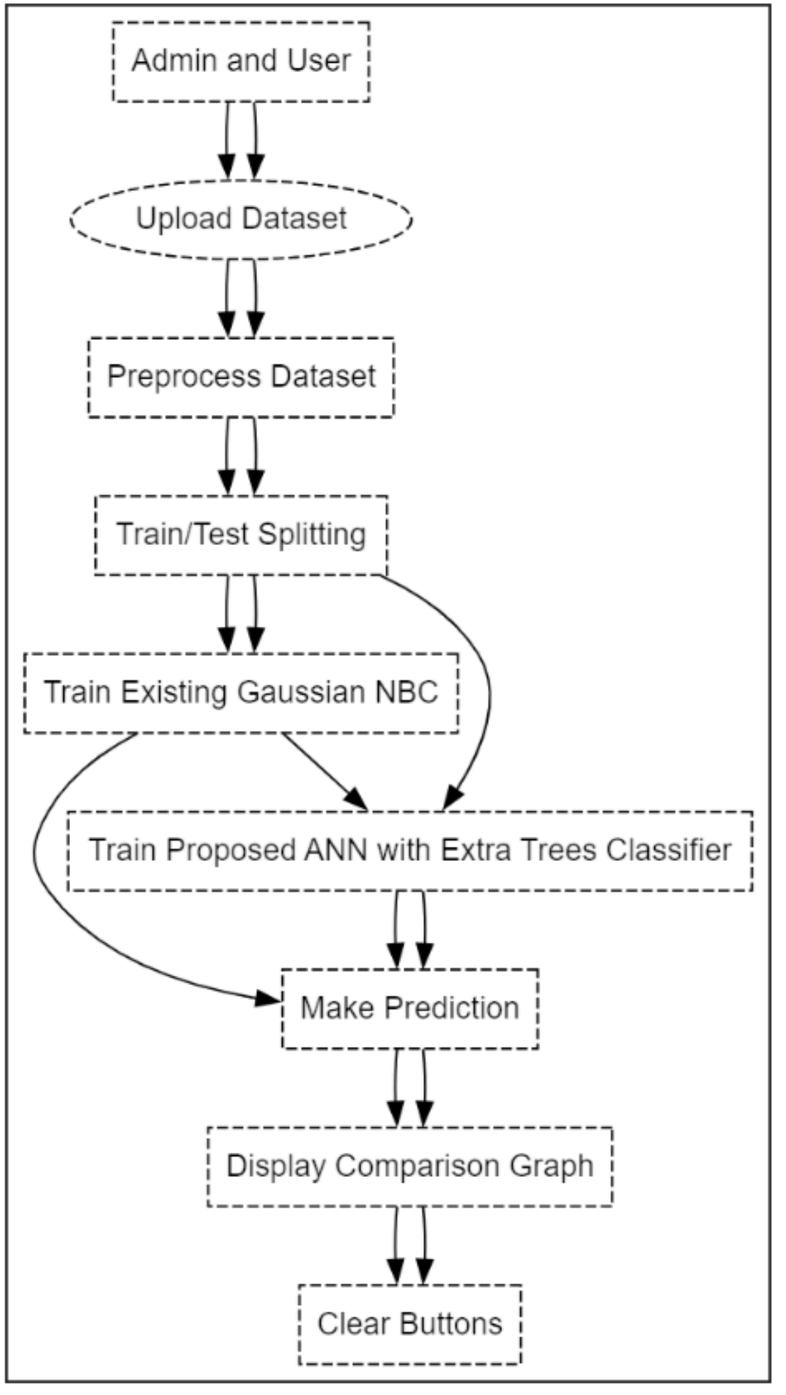
Scheduled Maintenance & Manual Logging
Involves routine checks, documentation, and part replacements based on technician expertise.
Prone to delays, human error, missed anomalies between checks, and inconsistent record-keeping.

Visual and Auditory Inspection
Relies on healthcare staff to spot visible or audible device malfunctions (e.g., alarms, lights, noises).
Effective in emergencies but reactive, subjective, and unscalable for large device networks.

DRAWBACKS

- Limited Fault Detection: Manual methods fail to identify subtle or emerging issues that evolve gradually over time.
- Inconsistency and Human Error: Subjective judgment and human error lead to unreliable diagnostics and decisions.
- Missed Transient Faults: Periodic inspections often overlook short-term anomalies that occur between checks
- Poor Scalability and Delayed Response: These methods are not suitable for large-scale IoT networks and can delay critical interventions.

PROPOSED SYSTEM

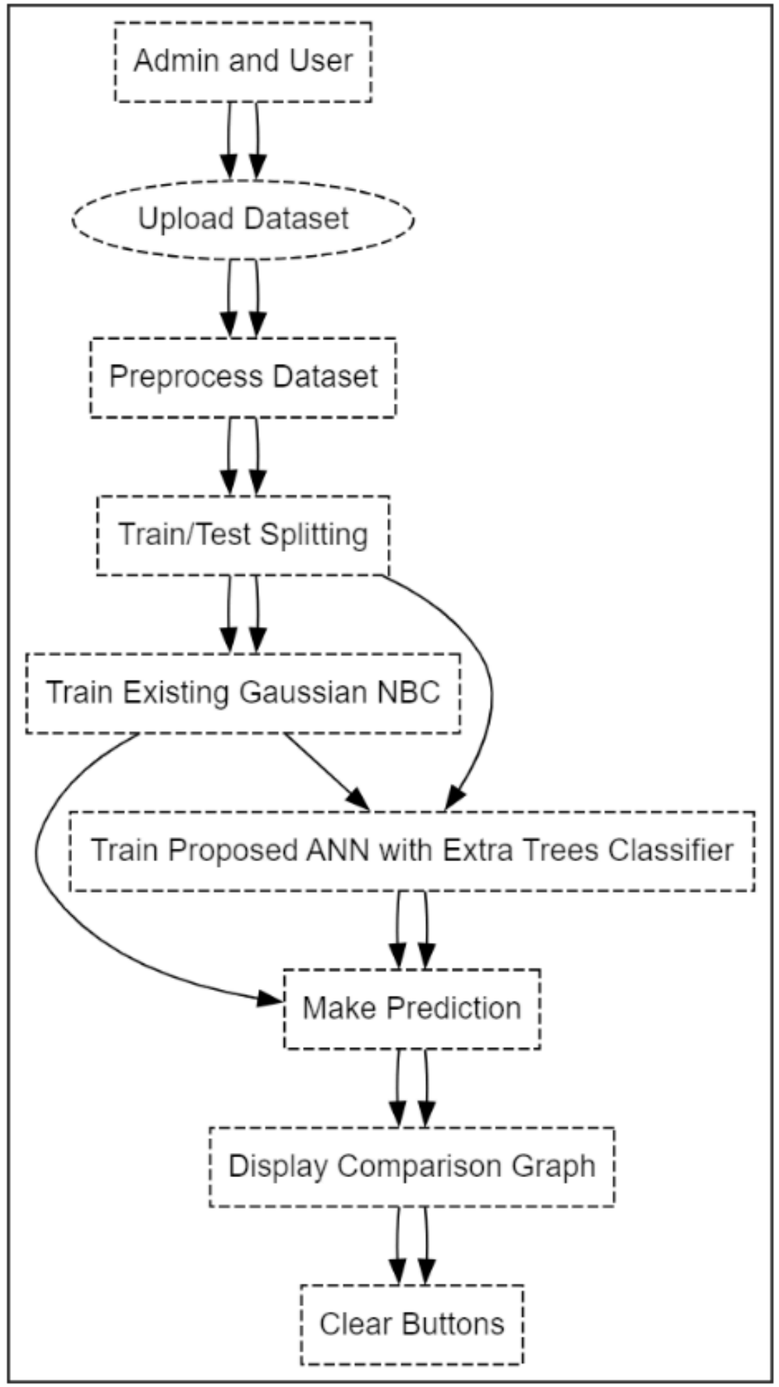

Admin and User
Upload Dataset
Preprocess Dataset
Train/Test Splitting
Train Existing Gaussian NBC
Train Proposed ANN with Extra Trees Classifier
Make Prediction
Display Comparison Graph
Clear Buttons

Admin Input:
The process begins with the admin managing and uploading the IoT dataset. This dataset includes labeled instances of device states collected from medical IoT devices.

IoT Dataset Collection:
The dataset consists of readings such as device metrics, operational parameters, and status indicators, labeled as either normal or anomaly.

Preprocessing:
This step involves cleaning the dataset, handling missing values, normalization, encoding categorical features, and ensuring data quality suitable for training machine learning models.

Train-Test Split:
The dataset is then divided into training and testing sets. The training set is used to develop models, while the test set evaluates model performance and generalization.

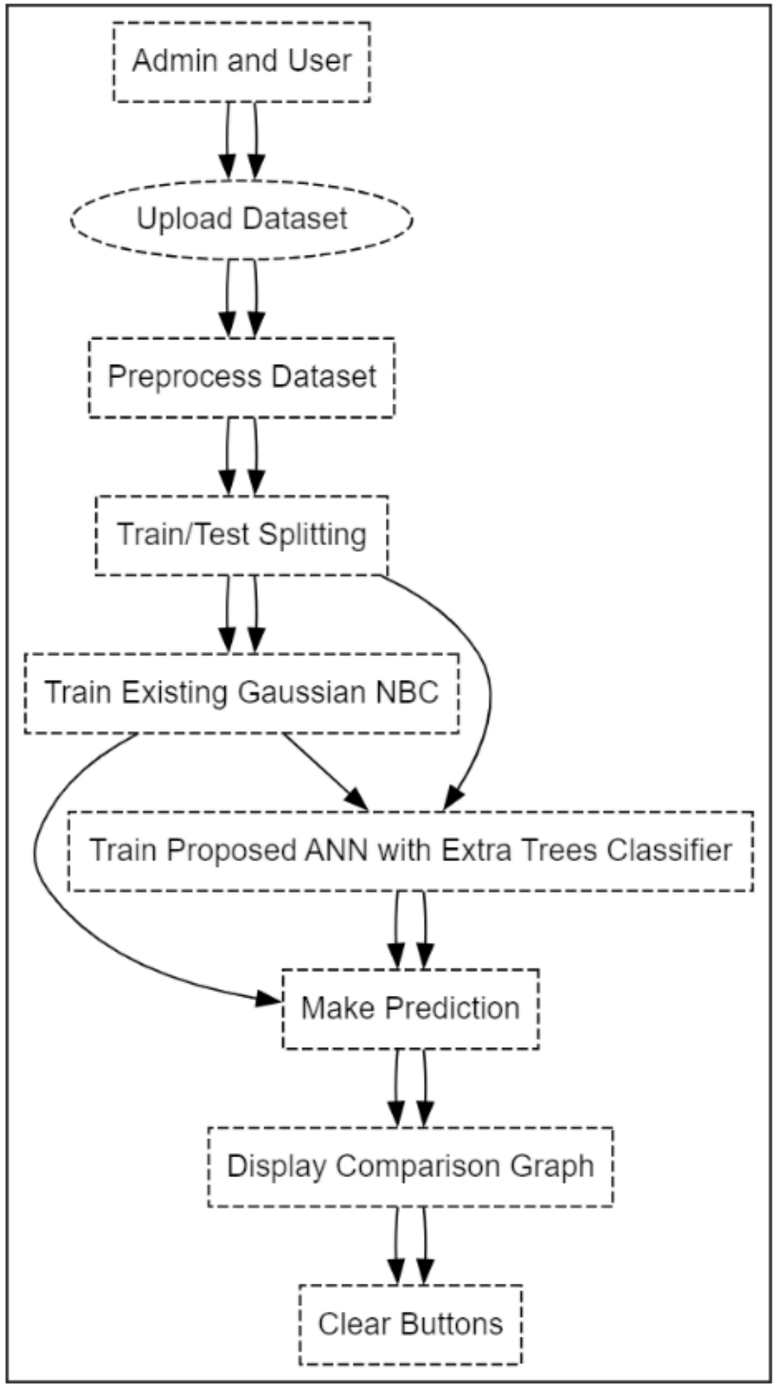


Model Building:
Two models are considered here:
Proposed ANN with ETC (Extra Trees Classifier):
This is the enhanced model that combines the learning capability of ANN with the robustness and ensemble nature of ETC, improving the classification performance.
Existing GNB (Gaussian Naive Bayes):
This serves as a baseline for comparison, already discussed in the existing system.

User Input – Test Data:
New or unseen IoT data from users is processed similarly and used as input for prediction.

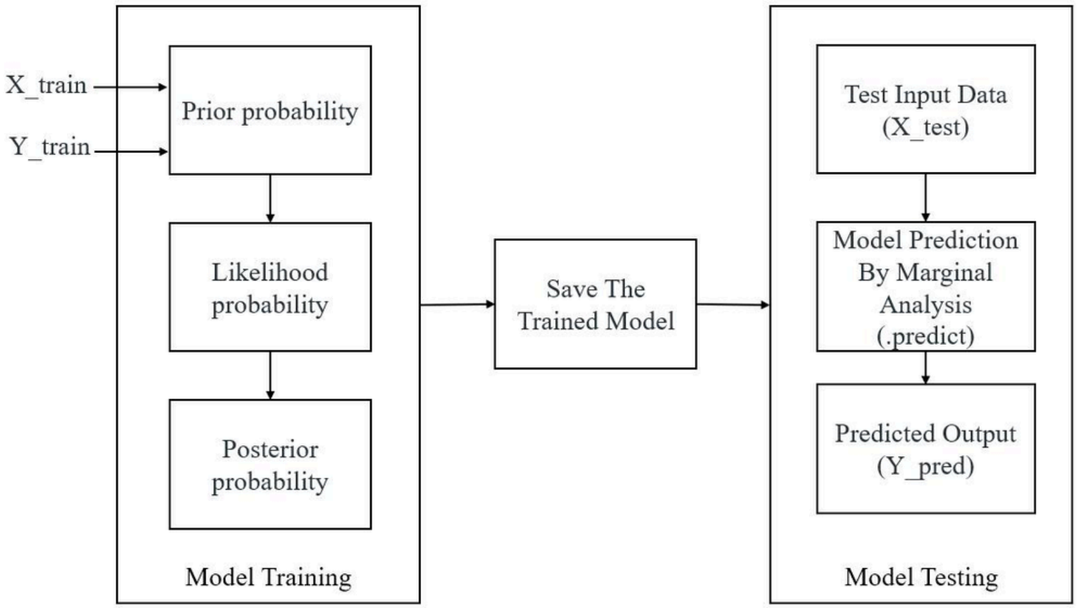
Model Prediction:
Both the proposed and existing models are evaluated. Predictions from both are sent to the prediction module to classify data as Anomaly or Normal.

Performance Evaluation:
The results of the proposed model are further analyzed to evaluate metrics such as accuracy, precision, recall, and F1-score. The proposed system is expected to outperform the existing GNB in detecting anomalies with higher reliability.



GAUSSIAN - NBC

The Gaussian Naive Bayes Classifier (GNBC) assumes that features are independent and follow a Gaussian distribution, which limits its effectiveness in real-world scenarios where features are often correlated and data distributions are skewed or contain outliers. It starts by receiving a dataset split into training and testing sets. The model simplifies computations by assuming conditional independence among features and estimates the mean and standard deviation for each feature within each class. These parameters are used to compute the likelihood of feature values under a Gaussian distribution. Bayes' Theorem is then applied to calculate class probabilities by combining prior probabilities with the likelihoods. The class with the highest posterior probability is selected as the predicted output. While GNBC is computationally efficient, its performance suffers with non-Gaussian data, class imbalance, or high-dimensional noisy datasets. The final predictions are compared against true labels using evaluation metrics like accuracy, precision, recall, and F1-score.



GAUSSIAN - NBC DRAWBACKS

- ·Assumption of Feature Independence: GNBC assumes that all input features are conditionally independent given the class label, which rarely holds true in real-world data. When features are correlated, the classifier's predictions can become inaccurate.

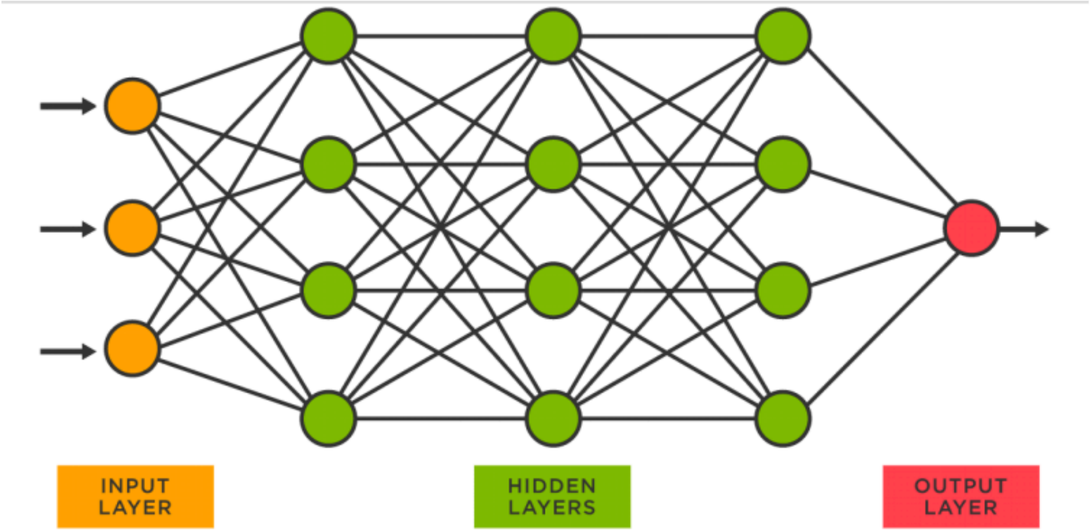
- Assumes Gaussian Distribution: GNBC specifically models features using a normal (Gaussian) distribution. This is a limitation for datasets where feature distributions are non-Gaussian, skewed, or have heavy tails, resulting in poor estimation of likelihoods.

- Sensitive to Irrelevant Features: The model can be negatively impacted by noisy or irrelevant features. Since all features contribute multiplicatively to the final probability, even unimportant features can skew predictions.

- Poor Performance with Imbalanced Data: GNBC tends to be biased towards the majority class in imbalanced datasets because it does not incorporate any mechanism to handle class imbalance directly, which can reduce its effectiveness in such cases.

PROPSED ANN WITH ETC

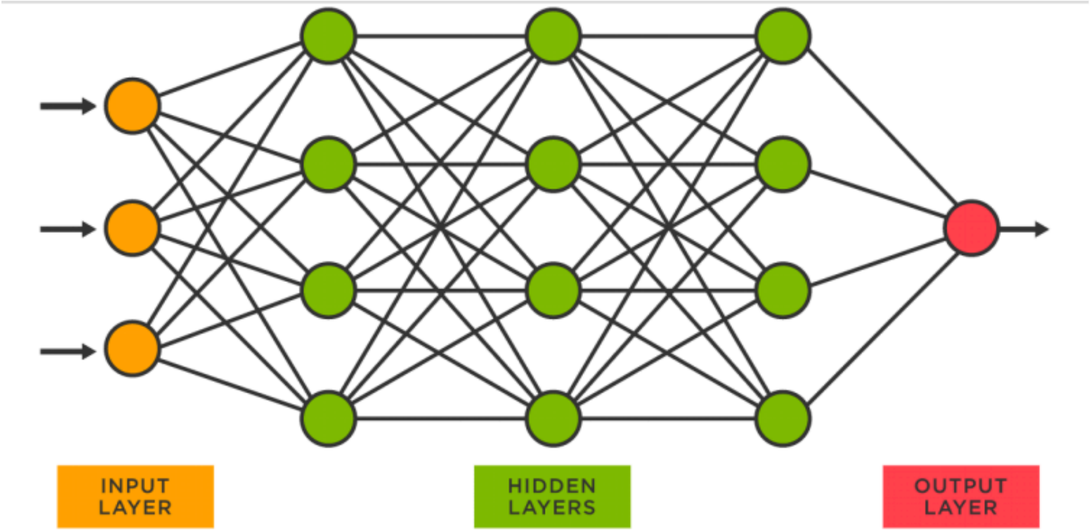
Artificial Neural Networks (ANNs) are effective for feature extraction by automatically learning abstract, task-specific representations from raw input data. This approach eliminates the need for manual feature engineering and is especially useful in complex domains like image classification and intrusion detection. The process starts with the input layer passing raw features to the network. Hidden layers then apply weighted transformations and non-linear activation functions (e.g., ReLU or sigmoid) to uncover complex patterns. As data moves through layers, increasingly abstract features are formed. At a selected hidden layer, neuron activations are extracted as feature vectors. These can be used for classification or input to other models like SVM or Random Forest. The entire network is trained using backpropagation and optimizers such as Adam or SGD, refining features over time.



ADVANTAGES

PROPSED ANN WITH ETC ADVANTAGES

- Automatically learns task-specific features from raw data without manual engineering.
- Capable of capturing complex, non-linear relationships between input variables.
- Provides scalable solutions that can adapt to various data sizes and complexities.
- Offers layered feature abstraction, useful for hierarchical understanding of data.
- Can be fine-tuned or integrated with other classifiers for improved performance.



SOFTWARE DETAILS

The appropriation of requirements and implementation constraints gives the general overview of the project in regard to what the areas of strength and deficit are and how to tackle them.

· Python IDLE 3.7 version (or)
· Anaconda 3.7 (or)
· Jupiter (or)

The project is developed using platforms like Python IDLE 3.7, Anaconda 3.7, Jupyter Notebook, or Google Colab. These tools support coding, testing, and running Python programs effectively. They offer different interfaces and functionalities to suit various development preferences. Anaconda and Jupyter are especially useful for data analysis and visualization.

SAMPLE CODE

```
def Preprocess_Dataset():
    global dataset
    global X,y
    text.delete('1.0', END)

    dataset = dataset.dropna()
    text.insert(END,str(dataset.isnull().sum())+"\n\n")

    non_numeric_columns = dataset.select_dtypes(exclude=['int', 'float']).columns
```

The Preprocess_Dataset function is designed to clean and prepare a dataset for further analysis or modeling. It begins by declaring the use of global variables dataset, X, and y, which allows the function to modify these variables outside its local scope. The function then clears any existing content from a text widget, likely part of a GUI interface. It removes any rows in the dataset that contain missing values using the dropna() method, ensuring the dataset is complete. After this, it calculates how many null values remain in each column (which should now be zero) and prints this information to the text widget for user visibility. Finally, it identifies all columns in the dataset that are non-numeric—excluding data types like integers and floats—storing these in the non_numeric_columns variable, likely for encoding or transformation in later steps of preprocessing.

PACKAGES USED

Pandas: Version 0.25.3 brought significant speed improvements for large-scale data processing, particularly in operations like groupby

tkinter: For creating the GUI (Graphical User Interface).
pandas & numpy: For handling and manipulating datasets.

seaborn & matplotlib.pyplot: For plotting data visualizations.

sklearn (scikit-learn): For preprocessing, splitting data, evaluating performance, and training machine learning models.

joblib: For saving and loading trained machine learning models.

NumPy: Version 1.19.5 introduced critical bug fixes and performance enhancements
over older versions, especially for operations involving large datasets

RESULT ANALYSIS

DATA-SET

	duration	protocol_type	service	flag	...	dst_host_srv_count	dst_host_same_srv_rate	dst_host_diff_srv_rate	attack
0	0	tcp	ftp_data	SF	...	25	0.17	0.03	No
1	0	udp	other	SF	...	1	0.00	0.60	No
2	0	tcp	private	S0	...	26	0.10	0.05	Yes
3	0	tcp	http	SF	...	255	1.00	0.00	No
4	0	tcp	http	SF	...	255	1.00	0.00	No

The dataset consists of multiple features that collectively describe network traffic sessions, primarily used for detecting and classifying different types of network attacks. Each record represents a single network connection with attributes capturing various characteristics of that connection.

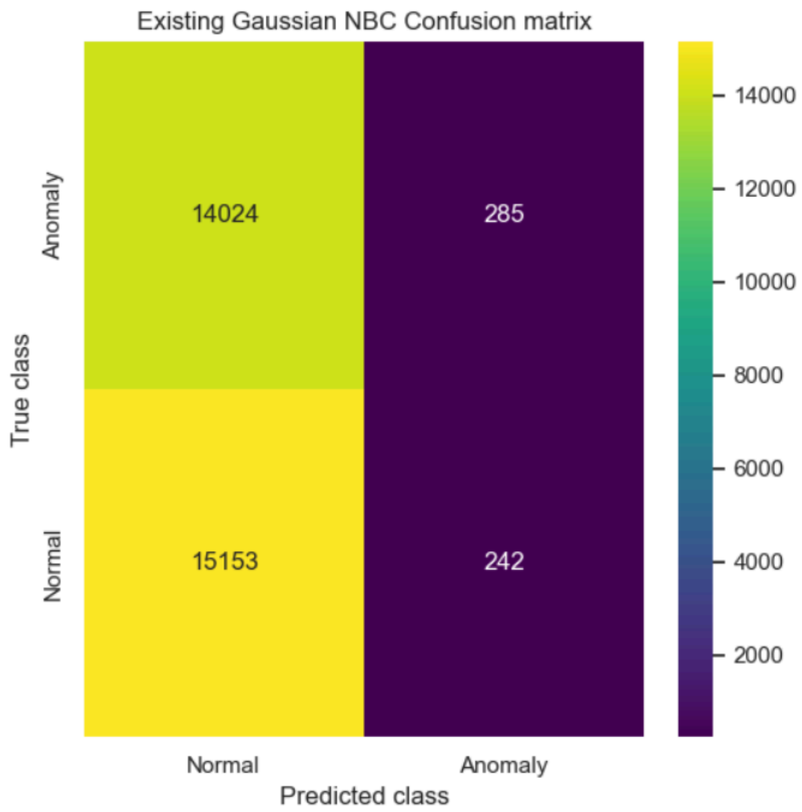
DATA-SET DESCSRIPTION

	duration	protocol_type	service	flag	...	dst_host_srv_count	dst_host_same_srv_rate	dst
0	0	tcp	ftp_data	SF	...	25	0.17	0.03	No
1	0	udp	other	SF	...	1	0.00	0.60	No
2	0	tcp	private	S0	...	26	0.10	0.05	Yes
3	0	tcp	http	SF	...	255	1.00	0.00	No
4	0	tcp	http	SF	...	255	1.00	0.00	No

- duration: Represents the length of the network connection in seconds. It measures how long the connection lasted.
- protocol_type: Indicates the protocol used in the connection, such as TCP, UDP, or ICMP. This categorical feature helps distinguish the nature of the communication.
- service: Specifies the network service on the destination, such as HTTP, FTP, SMTP, etc. This feature identifies which application or protocol the connection is related to.
- flag: Represents the status of the connection, showing different TCP flags like SF (successful connection), REJ (connection rejected), etc. It helps understand connection states.
- src_bytes: Denotes the number of data bytes sent from the source to the destination during the connection.
- dst_bytes: Denotes the number of data bytes sent from the destination back to the source.
- land: A binary feature indicating whether the connection is a “land attack,” where the source and destination IP addresses and ports are the same.
- wrong_fragment: Indicates the count of wrong fragments in the connection, which often points to irregularities or attack attempts
- urgent: Counts urgent packets in the connection, highlighting abnormal or critical communication

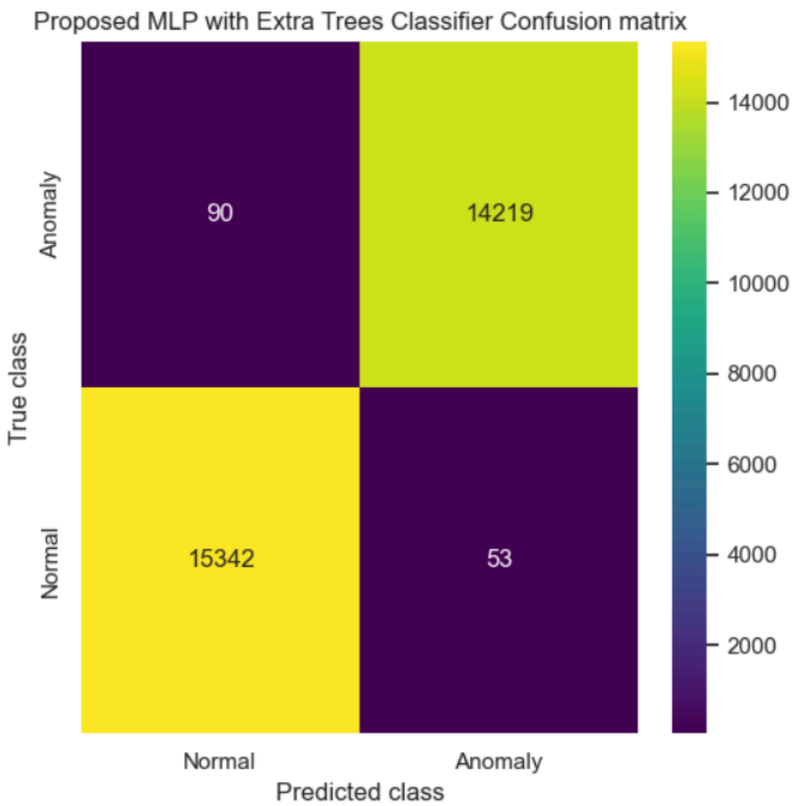
RESULT ANALYSIS

This confusion matrix illustrates the performance of a Gaussian Naive Bayes Classifier (NBC) for anomaly detection. It shows that the model correctly classified 14,024 anomalies and 242 normal instances but misclassified 285 anomalies as normal and a large number—15,153—of normal connections as anomalies. This indicates a high false positive rate, suggesting the model struggles to accurately identify normal traffic.
4o



RESULT ANALYSIS

This confusion matrix displays the performance of a proposed MLP (Multi-Layer Perceptron) combined with an Extra Trees Classifier. The model correctly predicted 14,219 anomalies and 53 normal instances, but it misclassified 90 anomalies as normal and a very high number—15,342—of normal connections as anomalies. Despite improved anomaly detection accuracy compared to the Gaussian NBC model, it still suffers from a significant false positive rate, mislabeling many normal activities as threats.
4o



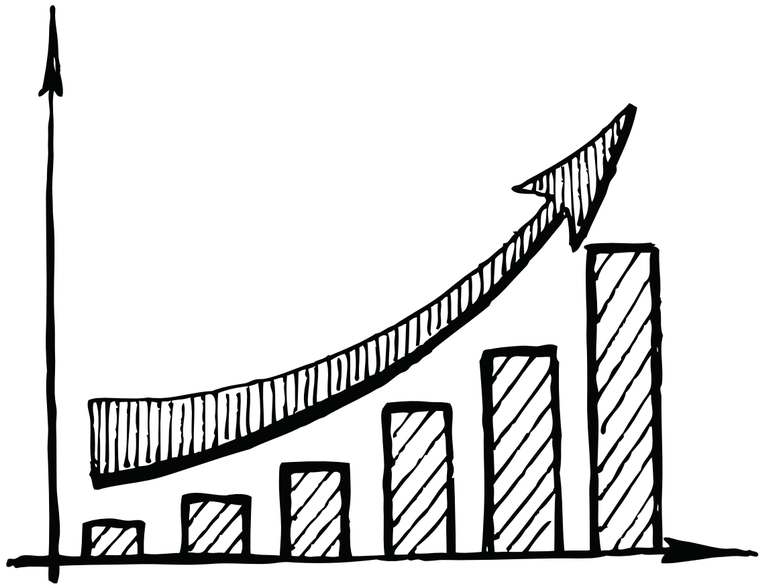
RESULT ANALYSIS

Algorithms Name	Accuracy	Precision	Recall	F1-Score
Gaussian NBC	86.3	86.4	86.3	86.3
ANN with the Extra Trees Classifier	99.53	99.53	99.53	99.53

Table : Performance Comparison for the Gaussian NBC and MLP with the Extra Trees Classifier algorithms.

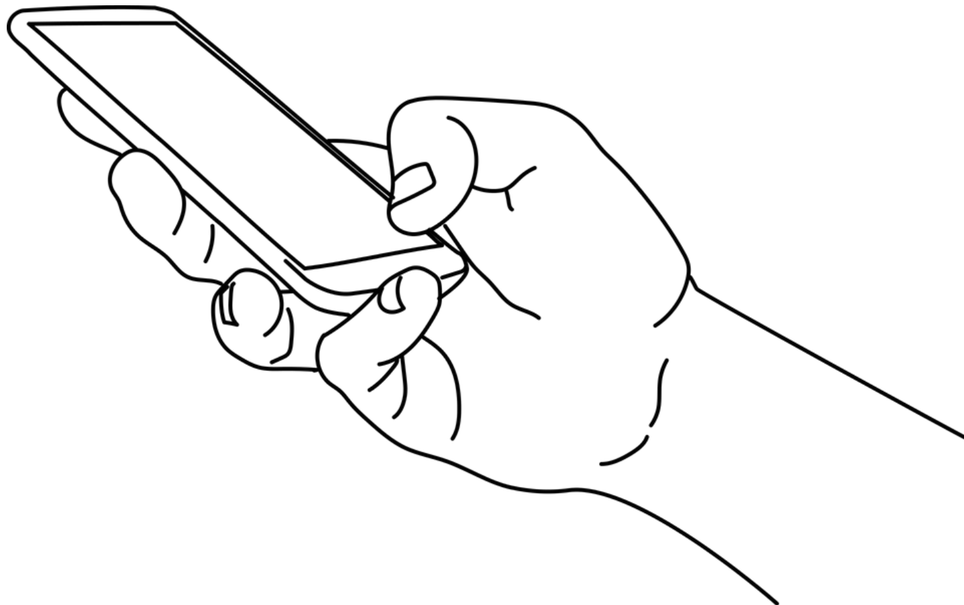
ADVANTAGES

- High Accuracy: Artificial Neural Networks (ANNs) can learn complex, non-linear patterns—ideal for health data with variability.
- Real-Time Analysis: Suitable for real-time monitoring and classification of IoT device states (e.g., wearable sensors, medical monitors).
- Fault Detection: Helps in detecting device malfunctions or unusual behavior early.
- Adaptive Learning: ANN models improve over time with more data, adapting to dynamic medical environments.



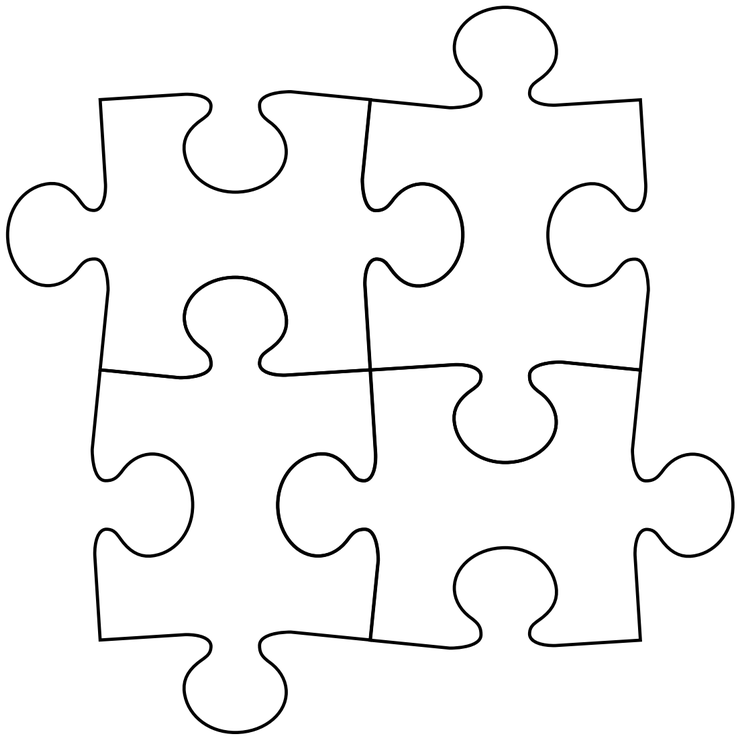
APPLICATIONS

- Remote Patient Monitoring: Classify normal vs. abnormal vitals from IoT-enabled devices.
- Smart Hospital Equipment: Track and classify operational states of critical devices (e.g., ventilators, infusion pumps).
- Predictive Maintenance: Identify early signs of failure in IoT devices.
- Elderly Care Systems: Detect emergencies like falls or cardiac anomalies using wearable devices.



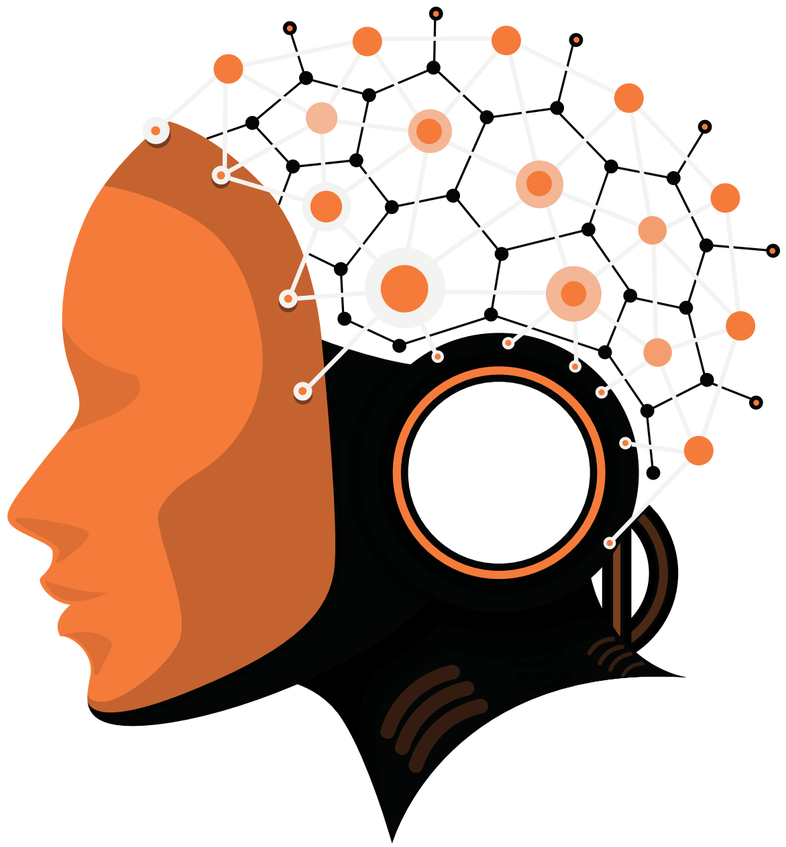
CONCLUSION

- The project developed a machine learning-based intrusion detection system to classify network traffic into normal and attack types using features such as protocol type, service, and byte counts.
- Two classifiers were used: Gaussian Naive Bayes (baseline with 86.3% accuracy) and Extra Trees Classifier, which outperformed with higher accuracy and evaluation metrics.
- Comprehensive preprocessing steps like null value handling, label encoding, and exploratory data analysis (EDA) were crucial for data readiness.
- The project proved that ML algorithms significantly outperform traditional rule-based detection in terms of accuracy, precision, recall, and overall reliability.



FUTURE SCOPE

- Deep learning models like RNN, LSTM, and CNN can be integrated to better capture sequential and spatial patterns in network traffic.
- Real-time detection and cloud deployment will enhance scalability and faster response to threats in dynamic network environments.
- The system can be upgraded to detect encrypted traffic, zero-day attacks, and adopt adaptive learning to handle evolving cyber threats.
- Developing a Graphical User Interface (GUI) will improve the system's usability, making it more accessible for security administrators in real-world setups.



REFERENCES

1] Akhtar, MD Mobin, Danish Ahamad, Abdallah Saleh Ali Shatat, and Ahmad Saleh Ali Shatat. "Big data classification in IOT healthcare application using optimal deep learning." International Journal of Semantic Computing 17, no. 01 (2023): 33-58.

[2] Lee, Jae Dong, Hyo Soung Cha, and Jong Hyuk Park. "M-IDM: A Multi-Classification Based Intrusion Detection Model in Healthcare IoT." Computers, Materials & Continua 67, no. 2 (2021).

[3] Vakili, Meysam, Mohammad Ghamsari, and Masoumeh Rezaei. "Performance analysis and comparison of machine and deep learning algorithms for IoT data classification."2001.09636 (2020).

[4] Saif, Sohail, Priya Das, Suparna Biswas, Manju Khari, and Vimal Shanmuganathan. "HIIDS: Hybrid intelligent intrusion detection system empowered with machine learning and metaheuristic algorithms for application in IoT based healthcare." Microprocessors and Microsystems (2022): 104622.

[5] Alsalman, Dheyaaldin. "A comparative study of anomaly detection techniques for IoT security using adaptive machine learning for IoT threats." IEEE Access 12 (2024): 14719-14730.

REFERENCES

6] Khan, Maryam Mahsal, and Mohammed Alkhathami. "Anomaly detection in IoT-based healthcare: machine learning for enhanced security." Scientific reports 14, no. 1 (2024): 5872.

[7] Hasan, Mahmudul, Md Milon Islam, Md Ishrak Islam Zarif, and M. M. A. Hashem. "Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches." Internet of Things 7 (2019): 100059.

[8] Dwivedi, Rajendra Kumar, Rakesh Kumar, and Rajkumar Buyya. "Gaussian distribution-based machine learning scheme for anomaly detection in healthcare sensor cloud." International Journal of Cloud Applications and Computing (IJCAC) 11, no. 1 (2021): 52-72.
132

[9] Vishwakarma, Monika, and Nishtha Kesswani. "A new two-phase intrusion detection system with Naïve Bayes machine learning for data classification and elliptic envelop method for anomaly detection." Decision Analytics Journal 7 (2023): 100233.

[10] Al Abdulwahid, Abdulwahid. "Detection of Middlebox- Based Attacks in healthcare Internet Of Things Using Multiple Machine Learning Models." Computational Intelligence and Neuroscience 2022, no. 1 (2022): 203795

REFERENCES

[11] Souri, Alireza, Marwan Yassin Ghafour, Aram Mahmood Ahmed, Fatemeh Safara, Ali Yamini, and Mahdi Hoseyninezhad. "A new machine learning-based healthcare monitoring model for student's condition diagnosis in Internet of Things environment." Soft Computing 24, no. 22 (2020): 17111-17121.

[12] Tiwari, Anurag, Viney Dhiman, Mohamed AM Iesa, Haider Alsarhan, Abolfazl Mehbodniya, and Mohammad Shabaz. "Patient behavioral analysis with smart healthcare and IoT." Behavioural Neurology 2021, no. 1 (2021): 4028761.

[13] Mansour, Romany Fouad, Adnen El Amraoui, Issam Nouaouri, Vicente García Díaz, Deepak Gupta, and Sachin Kumar. "Artificial intelligence and internet of things enabled disease diagnosis model for smart healthcare systems." IEEE Access 9 (2021): 45137-45146.

[14] Woźniak, Marcin, Michał Wieczorek, and Jakub Siłka. "BiLSTM deep neural network model for imbalanced medical data of IoT systems." Future Generation Computer Systems 141 (2023): 489-499.

[15] Yousaf, Iqra, Fareeha Anwar, Salma Imtiaz, Ahmad S. Almadhor, Farruh Ishmanov, and Sung Won Kim. "An Optimized Hyperparameter of Convolutional Neural Network Algorithm for Bug Severity Prediction in Alzheimer's- Based IoT System." Computational Intelligence and Neuroscience 2022, no. 1 (2022): 7210928.

THANK YOU.