Enhancing Smart Cities Through Real-Time Insights and Safety: A Comparative Study of Supervised Machine Learning Algorithms for Anomaly Detection in Emerging Urban Landscapes

U. Noor[1], R. Kanwal[2], Z. Rashid[3]

[1,2] *Department of Computer Sciences, International Islamic University, Islamabad, Pakistan*
[3]*College of Engineering, Seoul National University, South Korea*

[1]umara.zahid@iiu.edu.pk

Abstract- Today, IoT devices contribute to the creation of intelligent environments, encompassing structures like smart buildings, hospitals, banks, houses, and offices. Data acquired through sensors and devices often falls prey to corruption or damage, resulting in anomalous data. These anomalies have a substantial influence on the functionality of smart cities. This research specifically targets anomaly identification within smart cities. The purpose of the study is to examine the performance of machine learning algorithms and identify an optimal machine learning algorithm that is suitable for all settings of IoT-enabled smart cities. The research findings depict that Naïve Bayes achieved the highest average accuracy of 91.75% across all IoT-enabled settings of smart cities.

Keywords- Smart city, IoT Networks, Machine Learning, Anomaly Detection, Naïve Bayes Algorithm

I. INTRODUCTION

A common strategy for addressing the issues brought on by increasing urbanization and sustainability is the creation of smart cities. Smart cities gather and analyze data from several sources to give citizens and government officials insights. They do this by utilizing technology like machine learning algorithms, big data analytics, and Internet of Things (IoT) sensors. An important component of smart cities is anomaly detection, which is the process of finding odd or unexpected patterns or occurrences in data. Because machine learning can recognize trends and identify abnormalities in large-scale, complicated data, it has become a potent tool for anomaly detection in IoT-enabled smart cities. The accuracy and efficiency of smart city applications can be significantly impacted by anomalies in the data [1].

To harness the potential of smart cities, however, challenges must be addressed. Privacy and security vulnerabilities necessitate robust protocols. Algorithmic accuracy must be ensured, scaling capabilities enhanced, and collaboration among researchers, officials, and technology companies must be fostered. Previous studies are focused on a limited set of datasets, including network tracking, WI-FI router signal strength (RSS) for indoor localization, Intel laboratory, Yahoo, Process miner, SWaT, and household, all connected to different aspects of smart cities [1-5]. However, as of now, there isn't a single dataset that covers the complete analysis of an entire smart city. Previous studies on using machine learning for spotting unusual events in smart cities reveal a gap, not enough work has been done using various IoT datasets relevant to these urban environments [4]. While a few researchers have looked into how machine learning might help in future smart cities, they've faced a roadblock. They couldn't experiment with different datasets because these datasets are hard to find [4] [5]. Unfortunately, getting hold of datasets linked to smart cities isn't simple, and that's why some researchers have had to create their own simulated datasets. However, they haven't really shown clearly how they made these datasets in their studies [5]. This points out a need for datasets that are more complete and easy to get for smart cities, along with a common method for making such datasets. Having these datasets and methods could really help researchers create and test algorithms for finding anomalies that match the specific challenges and traits of smart cities. Secondly, the devices in smart cities have limited capability for computation, power consumption, and storage. Therefore, it is indispensable to design smart city applications based on such algorithms that are simple, computationally efficient, and optimized for handling resource constrains.

In the light of aforementioned research gap, the first objective of this research work is to contribute significantly by employing multiple IoT datasets to unearth anomalies within distinct smart city facets, as illustrated in Figure 1. Encompassing areas like Agriculture, Household, Transport, Weather Forecasting, Commercial, Network Tracing, Laboratory, and Healthcare Monitoring, the proposed approach covers a comprehensive spectrum of smart city components. We deploy diverse machine learning techniques, such as Naïve Bayes (NB), Decision Tree (DT), K-nearest neighbors (K-NN), Random Forest (RF), and Gradient Boosting evaluating their performance across metrics such as accuracy, recall, precision, and F1-score. In light of the existing literature's limited scope, which predominantly focuses on isolated smart city components, this research strives for comprehensive coverage.

The second objective of this research work is to identify an optimal machine learning algorithm for anomaly detection in smart cities. This algorithm should prioritize simplicity, avoiding excessive computational complexity, while demonstrating robustness in handling noisy and imbalanced datasets. As we navigate through the subsequent sections, section 2 provides literature review of existing research pertaining to anomaly detection in smart cities.

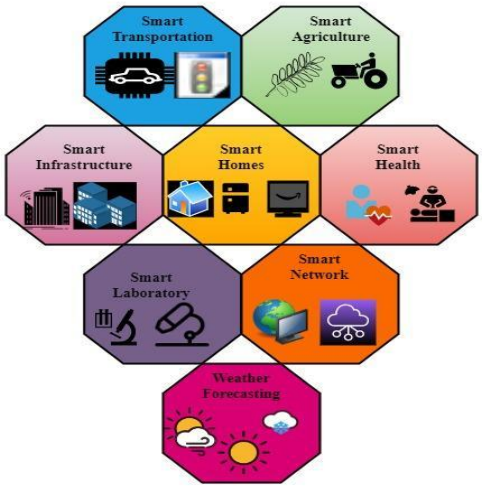

Figure 1. Components of a Smart City

Section 3 outlines the methodology employed in this research, while section 4 presents the research findings and discussion to discuss the outcomes. Finally, section 5 concludes the study, offering insights into potential future research directions within this dynamic field.

II. Literature Review

IoT systems offer wide-ranging utility across domains such as public safety, logistics, home automation, healthcare, and environmental monitoring. However, the susceptibility of crucial infrastructures like smart grids, industrial systems, and transportation networks to attacks raises concerns for cities and nations. In this section, a brief review of the Machine Learning (ML) techniques employed for anomaly detection in individual IoT application is discussed [23]. Notably, Supervised ML algorithms like Naïve Bayes (NB), Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), Logistic Regression, and Artificial Neural Network are applied to classify anomalous data. Additionally, hybrid or ensemble techniques are proposed by certain researchers to enhance detection approaches. Guo [2] introduced a novel unsupervised approach, tailored for anomaly detection in IoT systems using multidimensional time-series data. They introduced a GRU-Gaussian Mixture Variational Autoencoder (VAE) that incorporates a GRU-based deep latent embedding to capture temporal associations within the data. Employing a Gaussian Mixture model with a sequence of Gaussian distributions [20-22] facilitated enhanced understanding of the latent space. By introducing a Bayesian Inference Criterion (BIC) -based model selection approach, accuracy of the Gaussian Mixture latent space is augmented. Experimentation across four publicly available IoT datasets emphasized the effectiveness of the proposed system. Nusaybah Alghanmi [3] proposed the Hybrid Learning Model of Clustering and Classification for automatic labeling and anomaly detection. Firstly, this model classifies data into normal and defected data (which consist of anomalies) by using Hierarchical Affinity Propagation (HAP) unsupervised clustering. Secondly, labeled data is used for training in DTs and classify unseen future data. Precision, recall, and area under the precision recall curve and false positive rate are used and gives results 1.8, 1.8, 1.6, and 1.8 respectively. Two datasets considered here in this research, first is (LWSNDR) that is Labelled wireless sensor network data repository and second is Landsat Satellite are used in this paper. The result shows that HLMCC labeled data automatically and gives higher ranks in contrast of other models. Reddy [4] set the purpose of this research is to present an unique deep learning-based framework using a neural network dense random technique to identifying and categorizing anomalous from into the normal behaviors in the Internet of Things based on the type of attack. When compared to deep learning models, machine learning algorithms have a lower likelihood of exploring performance. It has been observed that deep learning neural network architectures conduct computations more efficiently and produce the appropriate outcomes for category threats. With the objective to figure out seven classified attacks that were found in the data set of traffic traces from the Distributed Smart Space

Orchestration System, this work aims to give a complete evaluation of experimentation performance and deep learning neural network design. The result shows that Deep Neural Network (DNN) achieves 98.29% accuracy, 97% Precision, 98% Recall, 98% F1-Score.

Bellini [5] proposed the solution for detection of anomalies. Researcher used Gradient Boosting technique using the CatBoost Algorithm [24-25]. Data is created by own researcher on snap4city platform by considering some air pollution and traffic related sensors. The objective of the researcher in this paper is to automatic detection of anomalies. Proposed algorithm achieve the better performance in accuracy. The result shows that proposed algorithm achieves 0.969% accuracy, 0.871% precision, 0.9225% F1 score. Proposed algorithm is not effective for sensors that is revolutionary change with time, and it require periodic training.

Mansoor [6] introduced an IF-Ensemble model utilizing a composition of ensemble, supervised, and unsupervised machine learning techniques for Wi-Fi indoor localization through RSS evaluation. The strategy exhibited a 97.8% accuracy, nearly 2% higher than previous accuracy rates, improving localization accuracy within indoor environments. Mahmudul Hasan [7] explored various machine learning approaches for identifying threats and anomalies in IoT systems. This involved employing ANN, DT, Logistic Regression, RF, and (SVM). Using evaluation metrics like accuracy, recall, precision, F1 score, and area under the receiver operating characteristic curve, this study conducted five-fold cross-validation on Kaggle's dataset. Results indicated significant accuracy for DT, RF, and ANN, with RF outperforming others in various metrics. Chunyong Yin [9] emphasized deep learning's role in anomaly detection. They integrated Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) based recurrent Autoencoder for feature extraction from raw data. To overcome limitations, a two-stage sliding window was implemented in data preprocessing. The model demonstrated improved accuracy, recall, precision, and F1-score. Di Wu [10] introduced the LSTM-Gauss-NBayes model, combining LSTM-ANN and Gaussian Bayes for anomaly detection. Using real-time series datasets, the model exhibited enhanced accuracy, recall, precision, and F1-score compared to existing methods. Ullah [11] proposed a Convolutional Neural Network (CNN) model for IoT network anomaly detection, categorizing binary and multiclass anomalies. A dataset was enhanced to create a comprehensive method, yielding high accuracy for different CNN models. Other research efforts focused on various aspects. Liu [14] targeted 'On' and 'Off' attacks within industrial IoT, while Anthi [15] employed ML classifiers for intrusion detection. Ukil [16] explored IoT-based healthcare analytics, and Pajouh [17] proposed a two-tier classification module for intrusion detection. Diro [18] assessed shallow and deep neural networks, and Usmonov [19] proposed digital watermarks for IoT security.

The research by M.M. Inuwa emphasizes the importance of anomaly detection in IoT systems and the potential of machine learning (ML) to address associated security challenges.

The limitations of traditional techniques is given. The advancements in deep learning, and federated learning is highlighted. However, only two IoT datasets are considered. Based on this the research findings for some major IoT systems in a smart city cannot be generalized [61]. A comparison of IoT anomaly detection in IoT systems using ML Classifiers is provided in table 1.

Table 1. Comprehensive Comparison of ML Classifiers' Performance on Trained and Tested IoT Datasets

Reference	Technique	Dataset	Objectives	Results	Limitations\ Gaps
Guo [2]	GGU-VAE (GRU-based Gaussian Mixture VAE), BIC-based model selection	1. Intel Berkeley Research Lab Dataset 2. Yahoo's anomaly detection dataset. 3. Process Miner's rare event detection dataset. 4. Secure Water Treatment (SWaT) dataset	To propose an unsupervised anomaly detection scheme for multimodal time series data in IoT systems that achieves high performance and cates issues related to high dimensionality, and multimodality.	Accuracy: Dataset 1:98% Dataset 2: 96% Dataset 3: 84% Dataset 4: 87%	The approach relies on generic datasets which may not be specifically tailored to the IoT environment or smart cities, potentially limiting the direct applicability and relevance of the findings to these contexts.
Reddy [4]	Deep neural network dense random technique	The DS2OS dataset, sourced from Kaggle, comprises 357,920 instances of data, each with 13 features.	Improving the effectiveness of anomaly detection within smat city environments	High accuracy (98.29 %), precision (97%), recall (98%), F1-Score (98%)	This paper focuses on future smart cities, specifically exploring network tracking within a single area. However, for a comprehensive understanding of future smat cities, it is essential to incorporate diverse datasets from various IoT fields.
Bellini [5]	Gradient Boosting technique using the CatBoost Algorithm	Custom-created dataset on snap4city platform	Automatic detection of anomalies in IoT infrastructure	High accuracy (97%), moderate precision (87%), F1 score (92%)	Not effective for sensors with revolutionary changes, requires periodic training
Mansoor [6]	IF-Ensemble (Isolation Forest ensemble), supervised and unsupervised ML methods	Wi-Fi indoor localization dataset from UCI repository	Detect outliers for Wi-Fi indoor localization	Accuracy (97.8%)	The approach relies on generic dataset which may not be specifically tailored to the IoT environment or smart cities. Limited to indoor localization, performance dependent on dataset quality and preprocessing
Hasan [7]	Various supervised ML algorithms	The DS2OS dataset, sourced from Kaggle,	Compare ML techniques for attack	Accuracy (99.4%), but	Lack of proposed anomaly detection

	(ANN, DT, LR, RF, SVM)	comprises 357,920 instances of data, each with 13 features.	detection in IoT systems	varied performance in other metrics	algorithm, limited to virtual environment
Chunyong Yin [9]	Deep learning model integrating CNN and LSTM-based recurrent Autoencoder	Yahoo's anomaly detection dataset.	Detect anomalies in time-series data	Accuracy (99.62 %)	The approach relies on generic dataset which may not be specifically tailored to the IoT environment or smart cities, potentially limiting the direct applicability and relevance of the findings to these contexts. Limited hardware resources may impact model optimization
Di Wu [10]	LSTM-Gauss-Nbayes model combining LSTM-NN and Gaussian Naive Bayes	Real-time series datasets (Power, Loop Sensor, Land sensor)	Detect anomalies in real-time series data	Accuracy Power: 96.9% Loop Sensor: 95.2% Land Sensor: 97%	Limited discussion on scalability, may require further optimization for different datasets
Ullah [11]	Convolutional Neural Network (CNN) model for anomaly detection in IoT networks	Custom IoT intrusion detection datasets	Detect and categorize binary and multiclass anomalies in IoT networks	Accuracy nearly 100% for all datasets and classes.	Potential chances of overfitting, limited discussion on scalability and generalizability to different IoT environments
Pajouh [17]	Two-tier classification module with PCA and LDA dimension reduction, Naive Bayes and CF-KNN algorithms for attack detection	NSL-KDD dataset	Identify and categorize harmful practices in network systems	Detection rate: 84.86%	Generic Intrusion detection dataset employed that cannot be generalized for smart cities specifically for IoT backbone networks
Diro [18]	Shallow and deep neural networks for attack detection in Fog-to-things architecture	KDDCUP99, ISCX and NSL-KDD	Comparison of shallow approach with Deep learning. Spot various threats and anomalies in network systems	Accuracy: 98.27	Generic Intrusion detection dataset employed that cannot be generalized for smart cities

III. Methodology for Anomaly Detection Using Machine Learning

This section discusses the methodology employed to execute anomaly detection using machine learning techniques on IoT-based smart city datasets. The goal of this study is to examine the effectiveness of varied machine learning algorithms in recognizing anomalies within an array of data categories. The section presents the research design, data procurement, data preprocessing, feature extraction, algorithm choice, and effectiveness assessment methodologies. Employing a comparative strategy, this research scrutinizes the competence of notable machine learning algorithms in detecting anomalies in smart cities. The steps of the research design are dataset selection, data preprocessing, data sampling, algorithm selection, and effectiveness evaluation, illustrated in Figure 2. The following subsections discuss these steps.

3.1 Data Collection

Smart cities are intricate systems with many technologies, applications, and data sources. Having a single dataset for everything isn't practical. Datasets used for understanding smart cities include open data, sensor data, social media, and business data. But working with these datasets is tricky due to data gaps, privacy concerns, and different data sources. Combining these different datasets is also complex. So, smart city experts usually mix data sources and methods to understand different city aspects. Due to security and privacy concerns, we can't find a real dataset covering everything in a smart city. Instead, *we're using various IoT anomaly detection datasets that focus on different parts of smart cities. These* datasets come from different places and are meant for specific smart city areas, such as weather forecasting, transportation, household, commercial, laboratory, healthcare monitoring, network tracking, and agriculture. In the following subsections, a brief description of each dataset is provided.

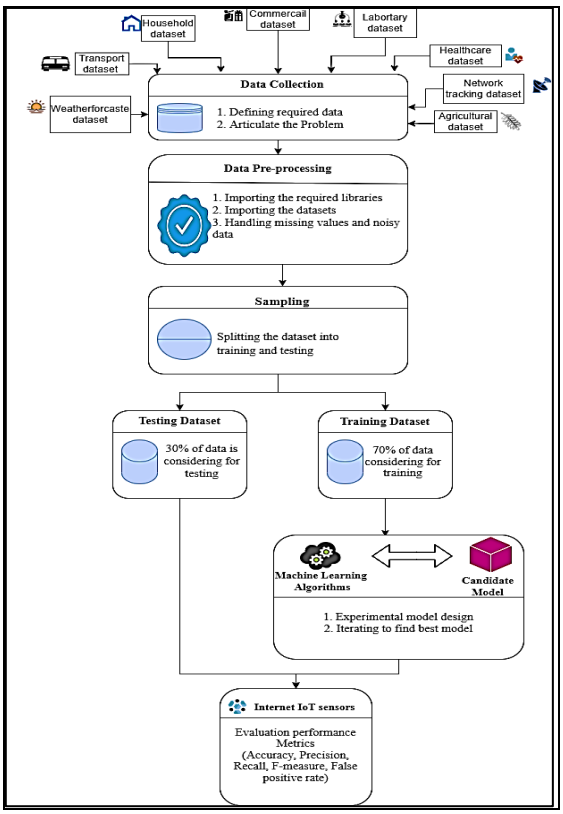

Figure 2. Research design steps of the proposed approach for anomaly detection

3.1.1 Weather Dataset

The IoT weather datasets collects data on weather conditions. The data is collected via sensors deployed at multiple points in a smart city. These features of the dataset are temperature, humidity, wind speed, precipitation, and other weather-related variables. The data is then transmitted to a central server for storage, processing, and analysis. The vital benefit of the IoT weather dataset is the provision of real-time data which fundamental for the city planners and the emergency responders.

They are able to make timely and informed decisions with respect to resource allocation and emergency preparedness. The IoT weather data enables emergency responders to anticipate the severe impact of happenings and react as a result [49].

The more data available, the more the accuracy of the weather forecasts improves. However, there are other factors involved as well that require constantly apprising and sanitizing the forecast models based on real-time data. Such models are beneficial for industries, agriculture lands and transportation services. Moreover, the historical records of IoT weather datasets can support researchers in analyzing weather patterns over long-term which is beneficial regarding observations of climate change [50].

To carry out this research work, the IoT weather dataset is obtained from a cloud store [12]. It consists of 4 features: temperature, pressure, humidity, and label. The label feature symbolizes the normal or anomalous data entry. The normal data is represented as 0 and 1 represents anomalous data.

3.1.2 Transport Datasets

The IoT Transport data is collected from sensors installed in the transportation system of the smart city. The sensors are GPS Trackers that collects information from vehicle and assets based on their GPS activity. This information analyze traffic for safety and security, and supports fleet management, predictive maintenance. The GPS tracking provides real-time monitoring, vehicle location, route optimization, and compliance with rules and regulations. The traffic analysis of aggregated GPS data helps in optimizing traffic flow and in identifying bottlenecks. This analysis can lead to making efficient infrastructures. The parameters of engine performance, and fuel consumption helps in predictive maintenance. The maintenance activities are optimized and the downtime is reduced which results in improved fleet performance [51]. Further IoT GPS tracking dataset provides real-time tracking, and assistance in emergency situations. The risky driving behaviors are identified. The anomalies in the dataset are traffic congestion, traffic violations, and suspicious activity [52].

The dataset under consideration is sourced from a cloud storage platform [12] and consists of 4 distinct features. The features include latitude, longitude, label, and type. The latitude and longitude features represent sensor-derived geo location values. The label feature differentiates between normal (labeled as 0) and anomalous (labeled as 1) data points. Additionally, the type feature describes the specific type of attack associated with anomalous instances. There are 595,686 instances in the dataset. This dataset allows for advanced analysis to uncover valuable insights.

3.1.3 Household Dataset

The IoT devices and sensors provide datasets related to a household setting in a smart city. In this research, IoT Refrigerator dataset is considered [12]. There are four features: fridge temperature, temperature condition (classified as high or low), label, and type.

The dataset provide information about the real time monitoring of temperature conditions of the food items inside the refrigerator. The purpose is to ensure the proper storage of the fragile food items. There are temperature condition labels that provide clear indications of potential issues based on which informed decisions regarding food consumption can be made [53].

Another information in the dataset is about energy efficiency and optimization in the households. The energy-intensive operations, and fluctuations in the temperature can be identified. The anomaly patterns in the dataset are related to sudden spikes and prolonged deviations in the temperature. The timely detection of anomalies ensure scheduling of maintenance that includes repairs, and component replacement. The aim is to achieve optimal performance [54].

3.1.4 IoT Motion Light Commercial Dataset

The IoT Motion Light dataset is about IoT enabled motion lights equipped with motion sensors. The motion of individuals is detected resulting in switching on or off the lights. The applications are both commercial and in residential settings. The purpose is to optimize energy consumption, and security monitoring by developing smart procedures [55]. The anomalies in the dataset are related to the security breaches [56].

The dataset is obtained from a cloud store [12]. It has four features: motion status, light status, label, and type. The motion status is the detection of motion. The light status represent the on or off state of the light. The label feature represents the presence and absence of the anomaly. The type feature tells about the type of attack or anomaly encountered in the dataset.

3.1.5 IoT Thermostat Dataset

The IoT Thermostat dataset is about temperature control and monitoring in the laboratory environments. The aim is to ensure optimal conditions for experiments so that the experiment outcomes can't be effected [57-58]. The dataset is in a cloud store [12]. The features are: current temperature, thermostat status, label, and type. The label features represents the presence and absence of the anomaly as 1 and 0 respectively. The type feature provides information about the type of anomaly.

3.1.6 Healthcare Monitoring Dataset

A real time Enhanced Healthcare Monitoring System (EHMS) is employed for obtaining the

healthcare monitoring dataset. It is also termed as WUSTL-EHMS-2020. There are four features: medical sensors, gateway, network, and control with visualization. The data is collected via sensors connected with patient's body. It is then sent to the gateway. The gateway further send it to the server system, where it is visualized. During the transmission, it is vulnerable to breaches and unauthorized interception. To address this vulnerability, an intrusion Detection System (IDS) is employed [59-60] to monitor real time traffic for anomalies. The dataset incorporated man-in-the-middle, data injection, and spoofing attacks.

The dataset has 44 features out of which there are 35 features related to the network flow metrics, and the rest of the eight features are related to the patients' biometrics. There is one feature that is label feature for anomaly detection [46]. The dataset is stored as csv format by a tool named as Audit Record Generation and Utilization System (ARGUS) [47]. The labeling of the data is based on the Source MAC address, where samples associated with the attacker's laptop MAC addresses are labeled as 1, while the remaining samples are labeled as 0.

It is important to note that this dataset [13] was created through the utilization of an actual real-time EHMS testbed, which was divided into network, medical sensors, gateway, and control with visualization components. The dataset provides valuable insights for research and analysis in the domain of healthcare security and intrusion detection.

3.1.7 Network Tracking Dataset

The network tracking dataset is labeled as "DS2OS Anomaly Detection IoT". It is publicly available on Kaggle. It is about communication among different IoT nodes. The nodes are connected through a shared middle ware, i.e., DS2OS. The sensor data is gathered from different IoT devices in a smart home setting. The anomalies are artificially injected for the purpose of evaluation. There are 357,952 records and 13 features. The features include source ID, type, address, location, destination service type, address, location, access node type, address, operation, timestamp, value, and normality.

The dataset covers various attack and anomaly types, including: 5,780 instances of Denial of Service (DoS) attacks, where an attacker overwhelms a resource with excessive traffic. There are 342 instances related to data probing attacks, aimed at manipulating data infrastructure, 875 samples related to malicious control, involving unauthorized system access, 805 examples of malicious operation, involving harmful code execution, 1,547 samples of scanning activities gathering data but potentially modifying it, 532 samples of spying attacks, attempting to collect confidential information, and 122 samples related to wrong setup scenarios due to improper configuration. Additionally, the dataset includes a substantial number of normal samples, totaling 345,899, indicating accurate and legitimate data. A visual representation of these aspects is provided in Table 7. The dataset contains a total of 357,953 instances, as noted.

3.1.8 Landsat Satellite Dataset

The Landsat Satellite dataset encompasses 5,100 records. There are 36 features. For soil categorization It is a valuable resource. The dataset consists of images of earth's surface, captured from satellite. The properties of the observed areas are characterized by the intensity values. The class labels classifies kind of soil as red soil, cotton crop, grey soil, damp grey soil, soil with vegetation stubble, and very damp grey soil. The dataset is converted into binary in a research paper for analysis [45]. In this binary version a new class label representing anomalous and non-anomalous records is added. The red soil, grey soil, damp grey soil, and very damp grey soil represent non-anomalous entries while cotton crop and soil with vegetation stubble are considered anomalous. In the binary dataset values of features are numeric ranging 0 to 255.

3.2 Data Pre-Processing

The data cleaning step deals with what to do when we have missing data. We can either take out the missing parts or put in values that make sense based on the type of data. We also look for values that are really different from the rest of the data, called outliers, and fix them if needed. Dealing with missing data is very important in data preprocessing. When we have missing data, it can mess up the results. In this study, we used two techniques depending on what was needed for the dataset. If we don't have much missing data and it doesn't affect the whole dataset much, we might choose to just take out the parts with missing data. But we have to be careful because this might mean we lose important information. Another way is to replace the missing data with the average value. This works well if the missing data seems random. For example, if we're missing numbers like temperature, we use the average of the numbers we do have. And if it's about categories, we use the one that shows up the most.

The DS2OS open-source dataset includes 357,952 samples with 13 features, mainly consisting of categorical data. However, two features, "Value" and "Accessed Node Type," have missing (null) values. Specifically, the "Accessed Node Type" attribute has 148 null rows, and the "Value" attribute has 2050 null rows. These null values are removed from the dataset. Additionally, the "Value" attribute contains noisy data, such as instances with values like "False" (25,966 instances), "None" (106 instances), "True" (14,460 instances), "Twenty" (200 instances), and "org.ds2os.vsl.core.utils.

AddressParameters" (11 instances). To address this issue, these noisy values are replaced with meaningful numeric representations, ranging from 0 to 21.

Another preprocessing step is dealing with data imbalance. To address data imbalance, various techniques are used, including random under-sampling, random oversampling, SMOTE, adjusting class weights, and using ensemble techniques. In our work, we have employed SMOTE (Synthetic Minority Over-sampling Technique) to tackle the class imbalance issue. SMOTE works by generating imitated data records for the anomalous classes as there fewer instances make class imbalance. The Euclidean Distance formula is used to find the minority anomalous class instances.

3.3 Supervised Machine Learning Algorithms

In this section, the intrinsic properties of ML algorithms used for anomaly detection in smart cities are briefly discussed. The purpose is to provide core insights and intuitive behind suitability of a particular ML approach for a particular IoT-enabled setting in a smart city environment.

The decision tree is a simple and efficient approach for ML model creation. The data preprocessing time is small and the instances do not require normalization and scaling. However, it is prone to overfitting [30-32]. Random Forest algorithm is the ensemble learning version of the decision tree to overcome the issues of overfitting caused by decision trees. It can efficiently handle larger datasets with missing values while maintaining optimal accuracy [27-29]. KNN is also a simple model that classifies new instances based on similarity with its neighbors. It is good in handling missing data. Also, it works efficiently with discrete and continuous attributes [33-35]. Naïve is a simple ML algorithm that is based on the concept of probabilities. It considers independence among features. [36], [38-41].

The gradient boost algorithm combines weak ML models for enhancing the accuracy of the prediction. It is good for both regression and classification activities. The limitation is it is computationally intensive and not good at handling missing values and mixed variable types [42-44]. SVM is also computationally intensive as the training time exponentially rise as the size of dataset scale. Also, its limitation is that it works with less features most of the time it is a preferred approach for binary classification [44].

3.4 Experimental Evaluation

The experiment is conducted using a Intel(R) Core(TM) i5-7200U CPU @2.50GHz 2.70 GHz processor, 20.0 GB RAM, 64-bit operating system, x64-based processor, and windows 10 pro. Table 1 shows the accuracy of all algorithms that are taken by applying to all datasets after using the data balancing technique SMOTE. The table 2 displays the outcomes for both the original datasets (column represented as “O”) and the results after applying SMOTE (column represented as “S”).

In the weather dataset, the random forest technique demonstrated superior performance, achieving the highest accuracy of 97% and outperforming all other methods. In contrast, the Naïve Bayes algorithm yielded a comparatively lower accuracy of 86% compared to the alternative techniques. Transitioning to the GPS Tracker dataset, the K-Nearest Neighbors (K-NN) method showcased optimal results with an accuracy of 94%, surpassing other approaches. Conversely, the Naïve Bayes algorithm exhibited a lower performance level with an accuracy of 82%. In the IoT Fridge dataset, all applied techniques exhibited a uniform accuracy of 85%.

Analyzing the motion light dataset, the machine learning techniques consistently delivered an overall accuracy of 86%. Shifting focus to the thermostat dataset, both the Naïve Bayes and Gradient Boosting methods exhibited heightened accuracy of 87% in contrast to other techniques. On the contrary, the Decision Tree algorithm demonstrated a relatively lower accuracy score of 80%. Within the Wustl-ehms-2020 dataset, all employed machine learning techniques demonstrated commendable performance; notably, the Decision Tree technique distinguished itself by achieving a remarkable accuracy of 97%. Within the DS2OS dataset, the collective performance of the machine learning techniques yielded an impressive accuracy rate of 98%. Particularly noteworthy, all techniques displayed a perfect accuracy of 100% on the satellite dataset, underscoring their exceptional capabilities within this specific context.

In the conducted research, attention is directed towards an additional dataset with class imbalance concerns, specifically a satellite dataset. To effectively address this issue, the application of the SMOTE technique for data balancing is prioritized. Prior to the implementation of data balancing measures, certain techniques yielded flawless accuracy scores of 100%. However, post the integration of SMOTE, there was a minor decline in accuracy to 99%, which still represents a notably high level of accuracy considering the inherent class imbalance within this particular dataset.

Conversely, the influence of data balancing techniques, inclusive of SMOTE, demonstrated variability across diverse datasets. Within certain datasets, the alterations in accuracy remained minimal or inconsequential. It's important to acknowledge that the efficacy of data balancing techniques, such as SMOTE, hinges on the unique characteristics of the dataset and the degree of class imbalance present.

Table 3 presents the precision of ML algorithms across all datasets. Within the weather dataset,

Naïve Bayes demonstrated the highest precision at 100%, surpassing other techniques. It's unusual for Naïve Bayes to achieve 100% precision in a real-world scenario, especially in a dataset as complex as weather data in features are highly dependent. However, a significant reason behind this is that the weather dataset have a very distinct and separable distribution of features for normal and anomalous instances that have resulted in high precision. Also, Naïve Bayes is a simple and computationally efficient algorithm that works well with limited computational resources. The weather dataset do not require complex decision boundaries to separate the classes, which is why Naïve Bayes have been able to achieve perfect precision without overfitting. Random Forest and Decision Tree algorithms perform well in terms of precision, achieving 87% and 96%, respectively.

Table 2. Accuracy of machine learning algorithms for the IoT smart city datasets

ML Model	IoT_weather Dataset		IoT_Transport Dataset		IoT-Fridge Dataset		IoT_Motion Light Dataset		IoT_Thermostat Dataset		Wustl-ehms-2020		DS2OS Dataset		Satellite Dataset	
	O	S	O	S	O	S	O	S	O	S	O	S	O	S	O	S
Random Forest	97%	97%	93%	93%	85%	85%	86%	86%	80%	73%	94%	93%	98%	98%	100%	99%
K-NN	93%	92%	94%	94%	85%	94%	86%	86%	86%	86%	92%	92%	98%	98%	100%	99%
Naïve Bayes	86%	86%	82%	86%	85%	85%	86%	86%	87%	87%	90%	90%	98%	98%	100%	99%
Decision Tree	96%	96%	92%	92%	85%	85%	86%	86%	80%	80%	97%	97%	98%	98%	100%	99%
SVM	–	–	–	–	–	–	–	–	–	–	91%	90%	98%	98%	100%	00%
Gradient Boosting tree	87%	86%	86%	86%	85%	85%	86%	86%	87%	87%	94%	94%	98%	98%	100%	99%

This is likely because decision tree-based methods are effective for capturing non-linear relationships between weather features and anomalies. Random Forest, being an ensemble of decision trees, further enhances performance through aggregation. K-NN exhibits lower precision (76%), possibly due to the high dimensionality of weather data and the need for careful tuning of the k parameter. In high-dimensional spaces, K-NN may struggle to find relevant neighbors, leading to decreased precision.

Similarly, in the context of the GPS Tracker dataset, Naïve Bayes excelled with a precision of 100%. This is surprising but could be attributed to the simplicity of the Naïve Bayes algorithm and the distribution of features in these datasets. Gradient Boosting displayed a relatively lower precision of around 86%.

When examining the IoT Fridge dataset, Naïve Bayes exhibited the highest precision at 100%, indicating that the dataset's features are well-suited to the assumption of feature independence. Random Forest and Decision Tree algorithms are not performing well, achieving 73% precision, indicating that these algorithms are not effective at capturing complex relationships in IoT fridge data, such as temperature fluctuations and energy consumption patterns.

In the case of the Motion Light IoT dataset, both K-NN and Naïve Bayes techniques yielded exceptional precision outcomes of 100%, outperforming other methods that attained comparatively lower precision scores. Transitioning to the thermostat dataset, Naïve Bayes achieved the highest precision levels at 100%. Conversely, alternative techniques returned precision scores of a lower magnitude.

Evaluation of the Wustl-ehms-2020 dataset unveiled proficient performance across all machine learning techniques, with K-NN garnering a precision score of 98%. Moving on to the DS2OS dataset, both Random Forest and K-NN techniques showcased superior precision at 98%, while the remaining methods achieved a precision score of 97%. Remarkably, in the satellite dataset, all techniques, except for Gradient Boosting which obtained 93%, achieved a precision of 100%.

Table 3. Precision of machine learning algorithms for the IoT smart city datasets

ML Model	IoT_weather Dataset		IoT_GPS Tracker Dataset		IoT-Fridge Dataset		IoT_Motion Light Dataset		IoT_Thermostat Dataset		Wustl-ehms-2020		DS2OS Dataset		Satellite Dataset	
	O	S	O	S	O	S	O	S	O	S	O	S	O	S	O	S
Random Forest	97%	97%	93%	93%	73%	73%	74%	74%	78%	70%	94%	91%	98%	98%	100%	99%
K-NN	80%	76%	98%	82%	98%	82%	100%	100%	98%	14%	98%	93%	98%	98%	100%	99%
Naïve Bayes	100%	00%	100%	00%	100%	00%	100%	00%	100%	00%	100%	96%	97%	97%	100%	99%
Decision Tree	96%	96%	92%	92%	73%	73%	74%	74%	74%	79%	78%	97%	97%	97%	100%	99%
SVM	–	–	–	–	–	–	–	–	–	–	97%	97%	97%	97%	100%	00%
Gradient Boosting tree	87%	96%	86%	00%	73%	73%	74%	74%	76%	76%	95%	95%	97%	97%	93%	99%

The analysis of precision presented in the table highlighted distinct patterns across datasets. Notably, the satellite dataset, characterized by class imbalance, exhibited subtle shifts in precision subsequent to the implementation of the SMOTE technique. Conversely, the majority of other datasets demonstrated minimal fluctuations in precision, with only a subset of techniques displaying minor variations. Intriguingly, the precision outcomes for the remaining datasets proved superior without the integration of SMOTE.

For instance, within the IoT weather dataset, while precision scores remained consistent for most techniques, two methods experienced a decrease in precision, and one exhibited an enhancement. Similarly, in the context of the IoT GPS tracker dataset, the majority of techniques maintained stable precision scores, except for K-NN, which observed a reduction, and GB, which showcased an improvement. In the IoT Fridge dataset, select techniques displayed alterations in precision.

Conclusively, the influence of SMOTE on precision displayed diversity across datasets and techniques. The application of SMOTE was selective, addressing class imbalance within the satellite dataset, whereas other datasets demonstrated either marginal changes or achieved superior precision results without the integration of SMOTE.

Table 4 represents the Recall results of all techniques which is applied on datasets.

Table 4. Recall of machine learning algorithms for the IoT smart city datasets

ML Model	IoT_ weather Dataset		IoT_GPS Tracker Dataset		IoT-Fridge Dataset		IoT_Motion Light Dataset		IoT_Thermostat Dataset		Wustl-ehms-2020		DS2OS Dataset		Satellite Dataset	
	O	S	O	S	O	S	O	S	O	S	O	S	O	S	O	S
Random Forest	97%	00%	94%	93%	85%	85%	96%	96%	80%	80%	94%	94%	98%	98%	100%	99%
K-NN	66%	60%	95%	70%	85%	70%	86%	86%	87%	2%	92%	92%	98%	98%	100%	99%
Naïve Bayes	86%	86%	86%	86%	85%	85%	86%	86%	87%	87%	92%	92%	98%	98%	100%	99%
Decision Tree	96%	96%	92%	92%	85%	85%	86%	86%	80%	80%	97%	97%	98%	98%	89%	96%
SVM	–	–	–	–	–	–	–	–	–	–	92%	92%	98%	98%	99.4 %	9.4%
Gradient Boosting tree	87%	87%	86%	86%	85%	85%	86%	86%	87%	87%	94%	94%	98%	98%	100%	99%

In the weather dataset, Random Forest emerged with the highest recall rate at 97%, outperforming other methods. Conversely, Naïve Bayes yielded a recall rate of 86%, placing it below the recall rates of the remaining techniques. Moving to the GPS Tracker dataset, K-NN exhibited a superior recall rate of 95%, surpassing both Naïve Bayes and Gradient Boosting techniques, which achieved recall values of approximately 86%. Within the Motion Light IoT dataset, the Random Forest technique excelled with a recall rate of 96%, while other techniques achieved a recall rate of 86%. In the thermostat dataset, Naïve Bayes, K-NN, and Gradient Boosting techniques achieved a recall rate of 87%, while Random Forest and Decision Tree techniques displayed lower recall rates. Examining the Wustl-ehms-2020 dataset, the Decision Tree technique garnered the highest recall value at 97%. For the DS2OS dataset, the overall recall performance of machine learning techniques demonstrated consistency, achieving 98%. In the case of the IoT Fridge dataset, all techniques exhibited the same recall rate of 85%. Notably, within the satellite dataset, all techniques attained a recall of 100%, except for Decision Tree and SVM, which achieved recall rates of 89% and 99.4%, respectively.

Even after implementing SMOTE, there was a technique that consistently displayed low recall rates, echoing the findings noted in terms of precision. This suggests that SMOTE's influence on enhancing recall for this specific technique was relatively minor. Consequently, the decision was made to maintain the results achieved prior to employing SMOTE for the rest of the datasets. However, in the case of the satellite dataset, the utilization of SMOTE yielded favorable outcomes, leading to reasonable levels of accuracy, precision, and recall. This progression now calls for the calculation of the F-measure to provide a holistic assessment of the model's performance. Table 5 represents the F-measure results of all techniques which is applied on datasets.

The analysis conducted unveiled a diverse impact of SMOTE data balancing across the datasets. Several datasets demonstrated enhanced accuracy when combining SMOTE with distinct machine learning algorithms. These enhancements became evident through heightened accuracy scores or significant alterations in particular data points.

Table 5: The F-measure score of all techniques on IoT datasets.

ML Model	IoT_weather Dataset		oT_GPS Tracker Dataset		IoT-Fridge Dataset		IoT_Motion Light Dataset		IoT_Thermostat Dataset		Wustl-ehms-2020		DS2OS Dataset		Satellite Dataset	
	O	S	O	S	O	S	O	S	O	S	O	S	O	S	O	S
Random Forest	0.97	0.88	0.93	0.93	0.79	0.79	0.79	0.84	0.79	0.75	0.94	0.92	0.98	0.98	0.94	0.99
K-NN	0.96	0.67	0.96	0.76	0.91	0.76	0.92	0.92	0.92	0.04	0.95	0.92	0.98	0.98	0.94	0.99
Naïve Bayes	0.92	0.92	0.92	0.92	0.92	0.92	0.92	0.92	0.93	0.93	0.94	0.94	0.97	0.97	0.93	0.99
Decision Tree	0.96	0.96	0.92	0.92	0.78	0.79	0.80	0.80	0.79	0.79	0.97	0.97	0.97	0.97	0.94	0.97
SVM	–		–	–	–	–	–	–	–	–	0.94	0.94	0.97	0.97	1	0.99
Gradient Boosting tree	0.81	0.91	0.8	0.92	0.78	0.79	0.79	0.80	0.81	0.81	0.95	0.94	0.97	0.97	0.96	0.99

Nevertheless, it's noteworthy that not all datasets encountered a substantial accuracy shift post-SMOTE. Certain datasets already exhibited commendable performance even without data balancing, leading to comparable accuracy scores between the original imbalanced dataset and the SMOTE-balanced dataset when employing various machine learning algorithms.

IV. Research Findings and Discussion

In this section, we will summarize the key findings of our research, focusing on the performance of different machine learning algorithms for anomaly detection in smart cities. Firstly, a brief overview of the key insights related to the accuracy, precision, recall, and f-measure results obtained for each algorithm across the eight IoT datasets is discussed. Secondly, the performance of the machine learning algorithms is compared. The consistent trends or substantial differences are highlighted. The strengths and weaknesses of machine learning models considering factors such as computational efficiency, scalability, robustness, and interpretability for detecting anomalies in smart cities is discussed. The scientific reasoning for the performance of each algorithm, explaining why certain algorithms may perform better or worse than others in specific scenarios based on their underlying principles, assumptions, and characteristics is discussed. The implications of the outcomes for real-world applications in smart cities are discussed. The limitations of the research work are also mentioned. Finally, the research study is concluded by suggesting a machine learning model as an optimal solution for anomaly detection in smart cities.

4.1 Key Insights

The Machine learning algorithms demonstrate varied performance for *IoT weather dataset*. Naïve Bayes achieves the highest scores of precision recall, and F-measure due to its probabilistic nature and modeling dependencies among features

efficiently. As discussed, there is a complex relationship among features of IoT weather dataset. Due to this, the Random Forest and Decision Tree algorithms also achieves consistent accuracy scores. Naïve Bayes consistently exhibits high precision, recall, and F-measure scores for the *IoT Transport dataset*. It takes the advantage of its probabilistic nature to model dependencies among latitude, longitude, label, and type features of the dataset and predict anomalies. There is a contrast in the performance of K-NN and Gradient Boosting Tree algorithms. K-NN achieves high accuracy but the precision, recall, and F-measure is quiet low as K-NN is sensitive to distance-based measures. On the other hand, Gradient Boosting Tree has a high recall score but lower score for precision which depicts it inability in handling false positives.

For the *IoT Fridge dataset*, Naïve Bayes consistently outperforms with high scores for precision, recall, and F-measure conditions. K-NN exhibits outstanding performance for precision after the data imbalance problem is resolved. Random Forest, Decision Tree, and Gradient Boosting Tree algorithms depict robustness in handling features in IoT fridge dataset by consistent performance for both the original and SMOTE applied datasets.

For the *IoT Motion Light Commercial dataset*, Naïve Bayes, K-NN, and Decision Tree achieve high scores for precision, recall, and F-measure. Random Forest and Gradient Boosting Tree algorithms demonstrate contrasting results as it achieves robust performance with a high recall on the original dataset. The results of precision and f-measure decline when SMOTE is applied.

Naïve Bayes consistently achieves high scores for precision, recall, and F-measure for the *IoT Thermostat dataset*. Decision Tree and Gradient Boosting Tree algorithms also perform well. K-NN exhibits inconsistent behavior with high score for accuracy but lower scores for recall and F-measure when data imbalance dealt. This indicates inconsistent behavior in detecting anomalies effectively.

For the *WUSTL-EHMS-2020 dataset*, Decision Tree achieve the highest scores for precision, recall, and F-measure. Random Forest and Gradient Boosting Tree algorithms and K-NN also perform well.

For the *DS2OS dataset and Landsat Satellite dataset*, Machine learning algorithms demonstrates high performance scores for accuracy, precision, recall, and F-measure.

4.2 Comparative Analysis

The comparative analysis of the ML algorithms for the *weather dataset* reveals that Naïve Bayes can efficiently capture dependencies among weather features. It is capable of handling high-dimensional feature spaces efficiently. This quality makes it an effective solution for anomaly identification in weather dataset where the features are sparse and independent. The nonlinear complex relationships among the features are captured by Random Forest and Decision Tree algorithms due to which they have high score for accuracy but slightly lower score for precision, recall, and F-measure as compared to Naïve Bayes scores. This shows their struggle against imbalanced class distribution. K-NN and Gradient Boosting Tree algorithms have lower scores for precision, recall, and F-measure due to the high-dimensionality among features of the weather dataset.

Similarly, Naïve Bayes outperforms for *the IoT Transport dataset* for all the performance metrics. The Random Forest, and Decision Tree algorithms have almost similar outcomes with minor differences. The comparative analysis tells that the aforementioned algorithms have correctly identified the intricate connections in the transport dataset. However, K-NN depicts inconsistent behavior. This shows that it is sensitive to feature representations and dependence on distance-based measurements. Gradient Boosting Tree works well for recall but have contrasting behavior for precision. This shows the presence of false positive.

For *fridge temperature conditions*, Naïve Bayes consistently outperforms other ML algorithms. Decision tree, random forest, and gradient boosting also represent consistent behavior for both original and smote-applied datasets which indicates their suitability for anomaly detection. K-NN inconsistent behavior still makes it unsuitable for anomaly detection in smart cities, although precision is improved for smote-applied dataset.

Naïve Bayes, K-NN, and Decision Tree algorithms consistently achieve high precision, recall, and F-measure scores, leveraging their respective strengths in probabilistic modeling, distance-based classification, and hierarchical decision-making to effectively detect anomalies in *motion and light behavior* Random Forest and Gradient Boosting Tree algorithms demonstrate robust performance in recall on the original dataset, highlighting their ability to capture diverse patterns and anomalies in motion and light status, albeit with moderate declines in precision and F-measure with SMOTE applied.

Naïve Bayes consistently outperforms other algorithms, leveraging its probabilistic modeling approach to effectively capture the complex relationships between temperature readings, thermostat status, and anomaly labels for *IoT thermostat dataset*.

Considering the collective findings, Naïve Bayes appears as the most suitable machine learning model for handling IoT datasets in smart city environments due to its effectiveness and simplicity. Limitations of the research include the need for addressing biases in datasets, optimizing model parameters, and evaluating performance under diverse real-world scenarios. Additionally, scalability issues may arise

in large-scale anomaly detection tasks, warranting further investigation.

4.3 Naive Bayes as the optimal model for anomaly detection in Smart Cities

In view of comprehensive discussion of results above, Naïve Bayes stands out as the optimal machine learning model for anomaly detection in smart cities. In the following, the scientific reasoning in support of this outcome is provided.

1. Naïve Bayes is a simple, and computationally efficient algorithm. It treats features independently that do not require extensive pre-processing or feature engineering. This quality makes it suitable for real-time anomaly detection in smart city where IoT networks comprise of devices that have limited computational capability, power consumption, processing power, and memory.
2. There are varied IoT devices in smart cities that results in increasing the dimensionality of the data. Naïve Bayes is suitable for dealing high dimensional and independent feature data. It adapts to unseen data patterns that makes it suitable for detecting novel anomalies.
3. In smart cities incorrect sensor's readings due to network disruptions is very common. Naïve Bayes algorithm provides a reliable operation in smart cities by depicting robustness against noisy data.
4. Naïve Bayes works with the statistical properties of the data instead of capturing or storing the data instances. This ensures that the model does not store any sensitive information about the training data. This quality makes it suitable for smart cities as data privacy and security are a major concern in such environments.

V. Conclusion and Future Work

With the increasing prevalence of IoT gadgets, smart environments such as hospitals, banks, factories, and cities are being transformed. However, the data collected from these IoT devices can often be distorted or degraded due to various factors such as device damage, data errors, issues with pattern matching, or even malicious attacks. This abnormal or anomalous data needs to be identified and addressed, and this is where anomaly detection comes into play. Machine learning techniques have emerged as effective tools for detecting anomalies in data, and their application can significantly improve the performance of any system.

While there is existing literature on anomaly detection in certain elements of smart cities, a comprehensive scientific evaluation of anomaly detection specifically tailored to smart cities' essential and significant components is lacking. Therefore, this study aims to address this gap by examining the effectiveness and efficiency of anomaly detection in smart cities. To achieve this, data is gathered from various sources, representing different aspects of smart cities. Subsequently, a range of machine learning algorithms are applied to these datasets to assess their performance. In future, our research plan is to compile datasets of a complete smart city using Generative Adversarial Network (GAN) algorithms.

References

[1] Syed, A. S., Sierra-Sosa, D., Kumar, A., & Elmaghraby, A. (2021). IoT in smart cities: A survey of technologies, practices and challenges. Smart Cities, 4(2), 429-475.

[2] Guo, Y., Ji, T., Wang, Q., Yu, L., Min, G., & Li, P. (2020). Unsupervised anomaly detection in IoT systems for smart cities. IEEE Transactions on Network Science and Engineering, 7(4), 2231-2242.

[3] Alghanmi, N., Alotaibi, R., Buhari, S. M., (2019). Hlmcc: A hybrid learning anomaly detection model for unlabeled data in the Internet of Things. IEEE Access, 7, 179492–179504.

[4] Reddy, D. K., Behera, H. S., Nayak, J., Vijayakumar, P., Naik, B., & Singh, P. K. (2021). Deep neural network based anomaly detection in Internet of Things network traffic tracking for the applications of future smart cities. Transactions on Emerging Telecommunications Technologies, 32(7), e4121.

[5] Bellini, P., Cenni, D., Nesi, P., & Soderi, M. (2020, September). Anomaly Detection on IOT Data for Smart City. In 2020 IEEE International Conference on Smart Computing (SMARTCOMP) (pp. 416-421). IEEE.

[6] Bhatti, M. A., Riaz, R., Rizvi, S. S., Shokat, S., Riaz, F., & Kwon, S. J. (2020). Outlier detection in indoor localization and internet of things (IoT) using machine learning. Journal of Communications and Networks, 22(3), 236–243.

[7] Hasan, M., Islam, M. M., Zarif, M. I. I., & Hashem, M. (2019). Attack and anomaly detection in IoT sensors in IoT sites using machine learning approaches. Internet of Things, 7, 100059.

[8] Gao, R., Zhang, T., Sun, S., & Liu, Z. (2019, June). Research and improvement of isolation forest in detection of local anomaly points. Journal of Physics: Conference Series, 1237(5), 052023.

[9] Yin, C., Zhang, S., Wang, J., & Xiong, N. N. (2020). Anomaly detection based on convolutional recurrent autoencoder for IoT

time series. IEEE Transactions on Systems, Man, and Cybernetics: Systems.

[10] Wu, D., Jiang, Z., Xie, X., Wei, X., Yu, W., & Li, R. (2019). LSTM learning with Bayesian and Gaussian processing for anomaly detection in industrial IoT. IEEE Transactions on Industrial Informatics, 16(8), 5244–5253.

[11] Ullah, I., & Mahmoud, Q. H. (2021). Design and development of a deep learning-based model for anomaly detection in IoT networks. IEEE Access, 9, 103906-103926.

[12] CloudStor. (n.d.). [https://cloudstor.aarnet.edu.au/plus/s/ds5zW91vdgjEj9i]

[13] Jain, R. (n.d.). [https://www.cse.wustl.edu/~jain/ehms/index.htML]

[14] Liu, X., Liu, Y., Liu, A., & Yang, L. T. (2018). Defending ON–OFF attacks using light probing messages in smart sensors for industrial communication systems. IEEE Transactions on Industrial Informatics, 14(9), 3801-3811.

[15] Anthi, E., Williams, L., & Burnap, P. (2018). Pulse: an adaptive intrusion detection for the internet of things.

[16] Ukil, A., Bandyoapdhyay, S., Puri, C., & Pal, A. (2016, March). IoT healthcare analytics: The importance of anomaly detection. In 2016 IEEE 30th international conference on advanced information networking and applications (AINA) (pp. 994-997). IEEE.

[17] Pajouh, H. H., Javidan, R., Khayami, R., Dehghantanha, A., & Choo, K. K. R. (2016). A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in IoT backbone networks. IEEE Transactions on Emerging Topics in Computing, 7(2), 314-323.

[18] Diro, A. A., & Chilamkurti, N. (2018). Distributed attack detection scheme using deep learning approach for Internet of Things. Future Generation Computer Systems, 82, 761-768.

[19] Usmonov, B., Evsutin, O., Iskhakov, A., Shelupanov, A., Iskhakova, A., & Meshcheryakov, R. (2017, November). The cybersecurity in development of IoT embedded technologies. In 2017 International Conference on Information Science and Communications Technologies (ICISCT) (pp. 1-4). IEEE.

[20] Usmonov, B., Evsutin, O., Iskhakov, A., Shelupanov, A., Iskhakova, A., & Meshcheryakov, R. (2017, November). The cybersecurity in development of IoT embedded technologies. In 2017 International Conference on Information Science and Communications Technologies (ICISCT) (pp. 1-4). IEEE.

[21] Guo, Y., Liao, W., Wang, Q., Yu, L., Ji, T., & Li, P. (2018). Multidimensional time series anomaly detection: A GRU-based Gaussian mixture variational autoencoder approach. In Proceedings of the Asian Conference on Machine Learning (ACML) (pp. 97-112).

[22] Liao, W., Guo, Y., Chen, X., & Li, P. (2018). A unified unsupervised Gaussian mixture variational autoencoder for high dimensional outlier detection. In Proceedings of the IEEE International Conference on Big Data (Big Data) (pp. 1208-1217).

[23] Gaddam, S. R., Phoha, V. V., & Balagani, K. S. (2007). K-means+ID3: A novel method for supervised anomaly detection by cascading k-means clustering and ID3 decision tree learning methods. IEEE Transactions on Knowledge and Data Engineering, 19(3), 345-354.

[24] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2018). CatBoost: Unbiased boosting with categorical features. Advances in Neural Information Processing Systems, 31.

[25] Dorogush, A. V., Ershov, V., & Gulin, A. (2018). CatBoost: Gradient boosting with categorical features support. arXiv preprint arXiv:1810.11363.

[26] Machine Learning Approaches in Smart Cities. (n.d.). [https://www.researchgate.net/publication/345226030_Machine_Learning_Approaches_in_Smart_Cities]

[27] Understanding Random Forest. (n.d.). [https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/]

[28] Machine Learning - Random Forest Algorithm. (n.d.). [https://www.javatpoint.com/machine-learning-random-forest-algorithm]

[29] Applications of Random Forest. (n.d.). [https://iq.opengenus.org/applications-of-random-forest/]

[30] Decision Tree Classification Algorithm. (n.d.). [https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm] (https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm)

[31] Decision Tree: Advantages and Disadvantages. (n.d.). [https://www.educba.com/decision-tree-advantages-and-disadvantages/] (https://www.educba.com/decision-tree-advantages-and-disadvantages/)

[32] Advantages of Decision Tree. (n.d.). [https://www.simplilearn.com/advantages-of-decision-tree-article] (https://www.simplilearn.com/advantages-of-decision-tree-article)

[33] Kataria, A., & Singh, M. D. (2013). A review of data classification using k-nearest neighbour algorithm. International Journal of Emerging Technology and Advanced Engineering, 3(6), 354-360.

[34] K-Nearest Neighbor Algorithm for Machine Learning. (n.d.). [https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning] (https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning)

[35] Batista, G. E., & Monard, M. C. (2002). A study of K-nearest neighbour as an imputation method. HIS, 87(251-260), 48.

[36] Naive Bayes Classifier. (n.d.). [https://www.javatpoint.com/machine-learning-naive-bayes-classifier] (https://www.javatpoint.com/machine-learning-naive-bayes-classifier)

[37] Naive Bayes. (n.d.). [https://gerardnico.com/data_mining/naive_bayes]

[38] Galit Shmueli, Nitin R. Patel, Peter C. Bruce. Data Mining for Business Intelligence: Concepts, Techniques, and Applications in Microsoft Office Excel with XLMiner [ebook]

[39] Yadav, K., & Thareja, R. (2019). Comparing the performance of naive bayes and decision tree classification using R. International Journal of Intelligent Systems and Applications, 11(12).

[40] Obulesu, O., Mahendra, M., & ThrilokReddy, M. (2018, July). Machine learning techniques and tools: A survey. In 2018 International Conference on Inventive Research in Computing Applications (ICIRCA) (pp. 605-611). IEEE.

[41] Ranjitha, K. V. (2018, December). Classification and optimization scheme for text data using machine learning Naïve Bayes classifier. In 2018 IEEE world symposium on communication engineering (WSCE) (pp. 33-36). IEEE.

[42] How the Gradient Boosting Algorithm Works. (n.d.). [https://www.analyticsvidhya.com/blog/2021/04/how-the-gradient-boosting-algorithm-works/]

[43] All You Need to Know About Gradient Boosting Algorithm – Part 1 (Regression). (n.d.). [https://towardsdatascience.com/all-you-need-to-know-about-gradient-boosting-algorithm-part-1-regression-2520a34a502]

[44] GBM in Machine Learning. (n.d.). [https://www.javatpoint.com/gbm-in-machine-learning]

[45] Goldstein, M., & Uchida, S. (2016). A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data. PloS One, 11(4), e0152173.

[46] Hady, A. A., Ghubaish, A., Salman, T., Unal, D., & Jain, R. (2020). Intrusion Detection System for Healthcare Systems Using Medical and Network Data: A Comparison Study. IEEE Access, 8, 106576-106584.

[47] Argus. (n.d.). [https://openargus.org] (https://openargus.org)

[48] Machine Learning - Datasets - UCR Time Series Classification Archive. (n.d.). [https://www.kaggle.com/datasets/francoisxa/ds2ostraffictraces?select=mainSimulationAccessTraces.csv]

[49] Jakaria, A. H. M., Hossain, M. M., & Rahman, M. A. (2020). Smart weather forecasting using machine learning: a case study in tennessee. arXiv preprint arXiv:2008.10789.

[50] Subashini, M. J., Sudarmani, R., Gobika, S., & Varshini, R. (2021). Development of Smart Flood Monitoring and Early Warning System using Weather Forecasting Data and Wireless Sensor Networks-A Review. 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 132-135.

[51] Saarika, P. S., Sandhya, K., & Sudha, T. (2017). Smart transportation system using IoT. 2017 International Conference on Smart Technologies for Smart Nation (SmartTechCon), 1104-1107.

[52] Jan, B., Farman, H., Khan, M., Talha, M., & Din, I. U. (2019). Designing a smart transportation system: an internet of things and big data approach. IEEE Wireless Communications, 26(4), 73-79.

[53] Ahmed, M. M., Qays, M. O., Abu-Siada, A., Muyeen, S. M., & Hossain, M. L. (2021). Cost-effective design of IoT-based smart household distribution system. Designs, 5(3), 55.

[54] Chou, J. S., Hsu, Y. C., & Lin, L. T. (2014). Smart meter monitoring and data mining techniques for predicting refrigeration system performance. Expert Systems with Applications, 41(5), 2144-2156.

[55] Baharudin, N. H., Mansur, T. M. N. T., Ali, R., & Sobri, N. F. A. (2021). Smart lighting system control strategies for commercial buildings: A review. International Journal of Advanced Technology and Engineering Exploration, 8(74), 45.

[56] Minoli, D., Sohraby, K., & Occhiogrosso, B. (2017). IoT considerations, requirements, and architectures for smart buildings—Energy optimization and next-generation building management systems. IEEE Internet of Things Journal, 4(1), 269-283.

[57] Schäuble, D., Marian, A., & Cremonese, L. (2020). Conditions for a cost-effective application of smart thermostat systems in residential buildings. Applied Energy, 262, 114526.

[58] Özgür, L., Akram, V. K., Challenger, M., & Dağdeviren, O. (2018). An IoT based smart thermostat. 2018 5th International Conference on Electrical and Electronic Engineering (ICEEE), 252-256.

[59] Islam, M. M., Rahaman, A., & Islam, M. R. (2020). Development of smart healthcare monitoring system in IoT environment. SN Computer Science, 1, 1-11.

[60] BK, B., & Muralidhara, K. N. (2015). Secured smart healthcare monitoring system based on IoT. International Journal on Recent and Innovation Trends in Computing and Communication, 3(7), 4958-4961.

[61] M.M. Inuwa and R. Das, A comparative analysis of various machine learning methods for anomaly detection in cyber attacks on IoT networks, Internet of Things (2024), doi: https ://doi.org/10.1016/j.iot.2024.101162.