Applied Soft Computing 136 (2023) 110054

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

LSTM-Markov based efficient anomaly detection algorithm for IoT environment

Shanmuganathan V. *, Suresh A.

Department of Networking and Communications, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu Dist., Tamil Nadu, 603203, India

ARTICLE INFO

Article history:
Received 11 October 2022
Received in revised form 11 January 2023
Accepted 16 January 2023
Available online 2 February 2023

Keywords:
Anomaly detection
Hybrid IoT environment
KNN
LSTM (Long Short-Term Memory Network)
Smart environment
Smart computing

ABSTRACT

The seamless integration of wireless and IoT device in normal day to day life and for smart homes has enabled more security and privacy needs. The attack or unauthorized access to these devices may result in issues in decision making inside the smart environment. Attacks and anomalies in open IoT system could provide false alarms and cause delay in processing the information's. The problem statement considered is, Anomaly detection systems, on the other hand, can be targets of attacks, h/w s/w failures, and thus fall short of their objectives. Because these are power-hungry devices carrying highly sensitive data, an effective attack and anomaly alert system is critical in an IoT-based environment. The present algorithms require high training and additional memory to identify the anomalies in the network, which is not practically feasible to simple edge-based computing devices. The anomalies and false information inside the network are handled with effective anomaly detection algorithm designed in this paper. An efficient anomaly detection method in real-time sensor is identified through markov and LSTM based network and the outliers in the data is clearly removed through the proposed approach. The proposed approach is tested with the real-time DHT sensor monitoring room temperature and room humidity. The proposed methodology provides 96.03% effective anomalies detection with 92.48% high training accuracy. The methodology showcase improved with 6.54% of effective anomaly rejection and 5.13% of training accuracy when compared with KNN algorithm.

1. Introduction

Internet of things (IoT) is a term refers to a vast connection of disparate devices like cameras, sensors, and drones. These objects can interact and communicate with one another via Internet protocols or proprietary messaging protocols (for example, IPv4/v6, IEEE 802.15.xx protocols, ZigBee, and LoRa) (e.g., CoAP). The ultimate intent is to establish a single objective that encompasses so-called Internet of Things applications (for example, weather forecasting, Building Ventilation, access permissions and Air-Conditioning). IoT technologies have a wide variety of application domains. Several examples include specific crop management, intelligent driving systems, and health. The Internet of Things can make use of almost any automation capable of providing pertinent data about change in parameters or its associated surrounding conditions. The anomalies in monitoring data may result deviated predictions. Anomalies can be classified into two types: those caused by technical or internal system problems (such as failure in communication or disturbances, malfunction), and those caused by data probity and security problems (such as malicious, third-party attacks) [1]. During operation, IoT systems may encounter one or both anomalies concurrently [2]. In this case, device accuracy is jeopardized, which could result in material loss, data theft, or even personal injury or death [3]. As a result, it is tedious to detect and process these anomalies for propermaintenance and proper operation of the system at all times [4,5]. When technical difficulties arise, for instance, the system should be capable of identifying and resolving them autonomously (e.g., selecting and switching to Head for the cluster if the current one fails) [6]. When a security threat occurs, the device should be capable to detect it while lessen the resulting damage (e.g., isolating and disconnecting a malicious node broadcasting incorrect data). As a result, it is critical to have effective surveillance systems that are capable of detecting such events, anomaly threats in real time, quickly raising alerts, and taking necessary measures toenhance systemreliability and accuracy [7,8]. (e.g., incursion for monitoring systems, fire for fire detection applications). The expert system designed for the IoT environment is depicted in Fig. 1.

It is not always simple to install and integrate WSNs into existing infrastructures, especially if the setting is temporary [9]. For instance, a surveillance network might only be needed for a brief

* Corresponding author.
E-mail addresses: sv8468@srmist.edu.in (Shanmuganathan V.), suresha2@srmist.edu.in (Suresh A.).

https://doi.org/10.1016/j.asoc.2023.110054
1568-4946/

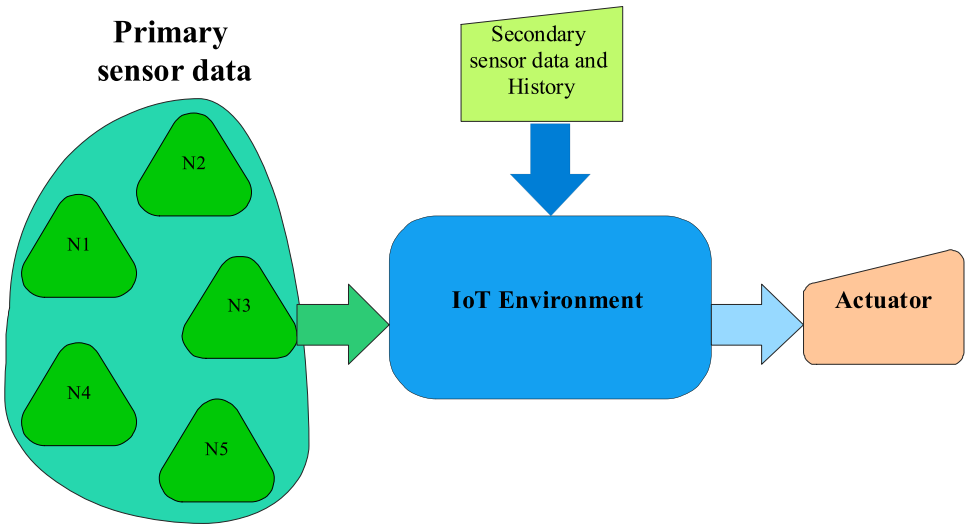

Fig. 1. Expert system architecture.

period of time at a particular location before it needs to be redeployed. Since the majority of applications use battery-powered nodes (sensors), the network lifetime and energy consumption are seen as important performance indicators of the application. Packets are sent to the Base Station (BS) in a multi-hop way to accommodate the nodes' low battery power. A routing algorithm determines the order of the nodes in which the packet is forwarded to the BS.

The issue is made worse due to the growing number of mobile devices and network sizes, which cause an excessive amount of network strain and delays. Additionally, transmitting a lot of data to the cloud requires more bandwidth and uses up a lot of the network's energy. An extension of cloud computing, edge computing is used for QoS provisioning in IoT-edge systems by using terminal edge devices with storage and processing power to offload compute demands to close-by edge servers, reducing network latency. Compared to cloud servers, edge servers have fewer computation resources, making it impossible for them to handle a high volume of computation requests [10]. Furthermore, for heavily loaded systems and numerous requests, the network energy consumption of edge computing may be greater than that of cloud computing. Therefore, it'sessential to reduce latency and power consumption in IoT edge systems to provide QoS for the user. An IoT network can use the associated EN to efficiently and swiftly retrieve pertinent information from the cloud in this fashion. Consequently, an edge-based computation strategy is appropriate for smart city environments to obtain latency-free responses. By deploying the tiny Edge devices, the bandwidth-related problems can also be resolved. The widespread use of such devices will guarantee the availability of services, resources, reduced latency, bandwidth, and cloud-independent data processing [11]. An edge device is able to provide all the services that were previously handled by a cloud server. If the application or cloud server asks for one, a thorough report may be delivered to it. Applications requiring real-time data can operate effectively with an Edge-based paradigm [12]. Providing an efficient data processing algorithms to such edge devices makes IoT environment an efficient one. In the existing algorithms the LSTM approach and other machine learning based schemes are deeply discussed. However, the individual algorithms do not provide effective results for thin client devices and especially IoT based Edge devices. The proposed algorithm combination algorithm provides fast results and better accuracy. The IoT devices are normally power and memory stringent in nature. The proposed algorithm provides better compatibility to the IoT based Edge devices. The main contribution of this article includes (a) providing an efficient anomaly detection system with the combination of LSTM and Markov based approach (b) The accuracy of the algorithm is tested with the real time sensors and IoT boards (c) Effectiveness of the algorithm is compared with the machine learning algorithms is evaluated. The paper is followed with the related work and proposed work algorithm in Section 3. The results and discussion are done in Section 4 and concluded in Section 5.

2. Related works

An area is covered by a network of sensor nodes called a wireless sensor network to accomplish a specific goal. These sensor nodes are distinguished by their constrained ability to communicate, compute, and most importantly, obtain energy [13]. Chain, Cluster and Grid based architectures are normally used in hierarchical protocols. To request data, a sink transmits interests. The interests-related information is then "pushed" towards the direction of the sink. In addition to transforming or caching data, intermediate nodes also direct interests depending on stored data.

Following the era of Industry 4.0, a lot of research has been done on wireless routing protocols. Energy, link stability, and delay are the main concerns of single path routing algorithms [1]. The received signal strength indicator metrics are used to measure link quality, and the battery level voltages and expenditure formulae are primarily used to indicate energy [17]. Link scheduling is also a major concern when it comes to providing quality of service and extending the life of a network. The Heterogeneous energy Traffic beware Sleep-Awake cluster protocol [2] provided a solution to the joint routing and scheduling problem using a column generation method. To select NEXT HOP from neighboring nodes, the next hop selection process is periodically revised using a probability approach. The random number generator is most commonly used to make probabilistic decisions. The S-AODV [3] routing protocol solves the clustering problem using the battery's residual energy, and frequent key revision improves network security and energy efficiency. The algorithm in [4] works in a random energy heterogeneous traffic environment. The TEAR algorithm discussed does not choose low-energy, high-traffic nodes for the NEXT HOP role. TEAR has the ability to reduce packet

Table 1
Forecasting and outlier detection algorithms comparison.

Name	Methodology	Inference
SVM	This method is mainly used for forecasting univariate time series data such as financial forecasting.	Can only be applied for Univariate time series data.
KNN	The k-nearest neighbor regression method is a nonparametric method that predicts the target's k nearest neighbor.	Distant measurement provides deviated predictions. Detection of outliers is deviated
LSTM [14]	Additional "gates" are used in LSTM-based models to memorize longer sequences of input data.	Deviated prediction of bivariate time series data.
SS-LSTM [9]	Three combinations of LSTM to capture future states	Deviated performance on long term predictions
KNN-TSad [12]	The KNN-TSAD can detect outliers with varied behavior features, including normal and abnormal traits, across time intervals.	Restricted to time intervals
CAE [15]	Concentrates on data out layers and provides better accuracy	Mainly designed for image and other high data sensors
F-RCNN [16]	Image based malware predictions	

dropouts and provide a more efficient routing path. The redundant packet transmissions, on the other hand, are not concentrated, which is a major concern in terms of increasing network lifetime. To avoid redundant data communication, the Sleep-awake energy efficient distribution (SEED) algorithm uses a duty cycling mechanism [5]. Because its radio is in sleep mode for the longest possible time, it has the disadvantages of idle listening. Increased focus on increasing network lifetime may result in a reduction in network throughput. The SEED method schedules cluster members in time division multiplexing mode, leading in a network with more idle hearing. The current routing protocols compute the path cost using link costs, and this lossless mechanism in wireless media abuses the spatial spectrum reusability. The voltage value can be used as a more accurate indicator of the current left-over energy level of a node. After a threshold value, the voltage level(V) in the node decreases exponentially and drastically. The battery recovery time-based enhancement of network lifetime is achieved with BRLE [7]. The battery's discontinuous discharge can extend the network's life and help the battery's health to a certain extent. The residual energy, battery recovery rate, and Markov approach are used to determine the NEXT HOP. The Markov model is a memory-free module that can make better decisions based on the current state of the battery. In the Internet of Vehicle, an authentication-based routing strategy is used, and its challenges are discussed. The security aspects of using public internet, as well as the speed and authenticity of the information, are thoroughly examined. IoV architectures are also investigated in terms of attacks, threats, and network models [8,18]. The above algorithm allows for better decision-making when it comes to increasing network lifetime and also provides a better solution. However, the algorithm key rotation scheme was not specific and was not a promising solution to the heterogeneous environment's existing security issues. By selecting NEXT HOP based on the remaining residual energy through voltage indication and frequently changing the network key with EX-OR methodology, the proposed work provides a better solution than the current work. The operation is more efficient in that it reduces the need for frequent key sharing while also ensuring the network's security. Data forecasting with the different time series data prediction are usedin such edge routing devices [14]. Autoencoders based algorithm with deep learning and other machine learning algorithms are available [19], these are mainly designed for the images and high data sensor devices in the smart environment. Convolutional Autoencoder (CAE) uses reduction in dimensionality and provides data out layers and other anomalies [15]. Image based malware attack and outlayer detection is done through Recurrent Neural Network (F-RCNN) approach [16]. A CNN with the LSTM approach is carried out to find the anomalies detection in the audio data, the results discussed were also impressive for audio based out layer detection [20]. The methodology is mainly used in wireless communication packets. The SVM, KNN, LSTM, SS-LSTM and KNN-TSad based algorithms. These algorithms mainly predict the sensor anomalies with the univariate time series data and in some cases does not support long term predictions. Table 1 showcase the comparison of different time series data prediction algorithms.

The existing research gap includes low accuracy in long time prediction and algorithm low accuracy due to the light weight data handled in the IoT environment. Many algorithms in the related work are less effective for the light weight data application. The algorithms also consume more memory and time to identify such anomalies are also high.A combination of LSTM and Markov based approach is used in this paper to predict a long-term time series data and to predict the sensor fault conditions. The deviated performance and for design of IoT based environment live data is deeply discussed in this paper. The algorithms provide deviated results for the live data from the sensors in IoT based environment, however the proposed algorithm a combination of Markov and LSTM provides more accurate results with respect to ground truth data.

3. Effective anomaly detection in real time IoT environment through machine learning approach

IoT devices are used in the majority of monitoring applications because of their evident benefits, including lower costs owing to cable replacement, adaptability to different network topologies, scalability, and cheaper maintenance and commissioning costs. IoT and sensor networks have been used successfully to implement solutions in many fields, such as environmental monitoring, preventing natural disasters, tracking the location of people, assets, or dangerous gases, monitoring the condition of gear, and controlling processes in industrial settings, among many others. The effective anomaly detection in IoT environment is done through sensor data, history and with machine learning anomaly detection [21]. The sensor data are collected in real time from sensors like DHT sensor which provide both temperature and humidity values. The sensor data is connected with the processor which is enabled with the machine learning and anomaly detection algorithm. The sensor's previous data history and data with the adjacent sensors are also considered for anomaly detection. Fig. 2. shows the proposed architecture in which the sensor data such as humidity and temperature are considered and real-time anomaly detection is carried out. The data from the adjacent sensors are shared with the IPv4/IPv6 protocols.

Algorithm 1 elucidates the anomaly detection with the sensor raw data and with the help of LSTM network. The sensor data is acquired with the real time sensor along with the time stamp. The sensor data is cleaned with the help of data averaging algorithm. Once the data is cleaned and unnecessary noises are eliminated,

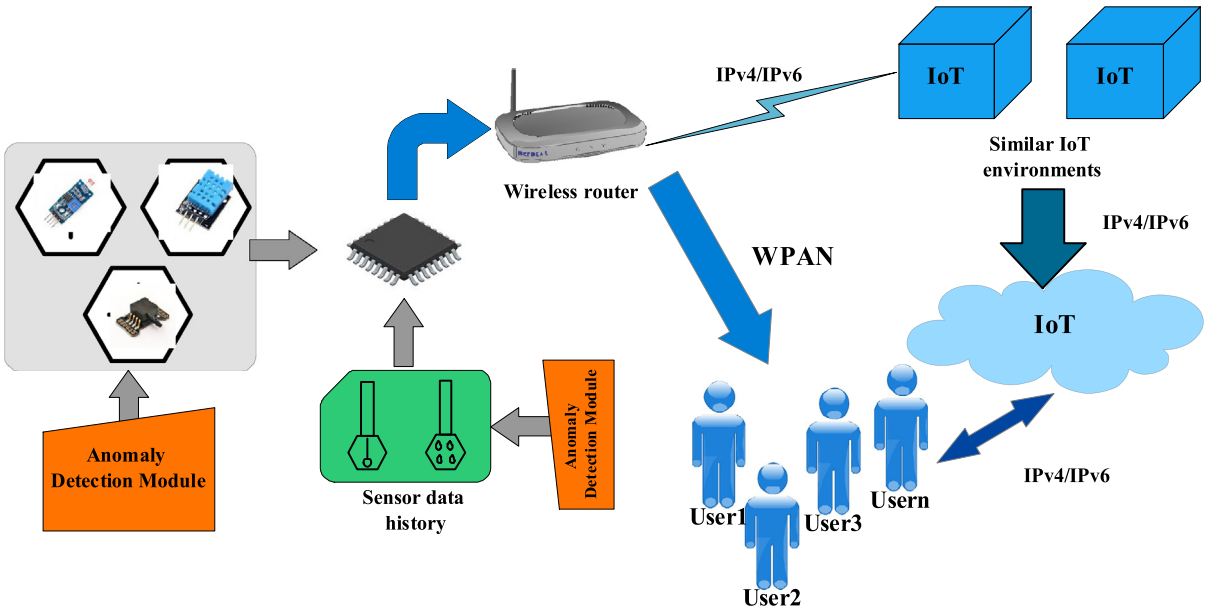

Fig. 2. Proposed system Anomaly Detection Architecture.

the data is compared with the LSTM algorithm to sense anomalies [22]. The Finite state machine model given in Fig. 3 showcase the typical working probability of the sensor. The sensor fault is accurately identified through probability approach. The present working state of the sensor is identified through markov approach to avoid anomalies due to sensor faults. Since markov is the memoryless model and the transition between the states are mainly influenced by the present states. The markov model is considered for realizing the data from the sensor to be worthy [23]. Fig. 3 showcase the working finite state machine of any sensor considered in the IoT environment.

Algorithm
Input : Sensor Data Primary,
Sensor Data Secondary,
Output : Sensor Anomaly
Begin Process
while (Si)
get(Sensor data primary)
if(sensor data=≈Sensor data, Avg)
Check secondary sensor data
for(Si ≈ LSTM)
identifyanomaly;
Alert;
end for
end if
end while

Algorithm 1. Proposed Anamoly detection methodology

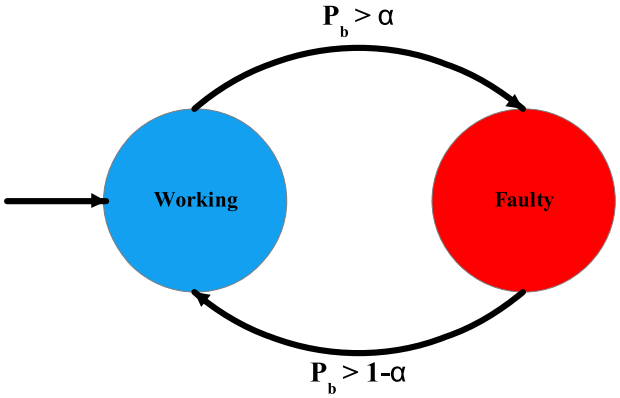

Fig. 3. FSM of sensor.

3.1. Finite state machine

Fig. 3 showcase the sensor working and faulty condition states representation. The prediction of working a faulty sensor is predicted through Markov approach [24]. The Markov is a memoryless model and sensor working principle is also not in depended with the future states, the sensor working is predicted with the help of Markov approach.

3.2. Markov model

The probability of changing from s to t for n steps is as

$$P_{st} = P(P_n = t|P_0 = s) \quad (1)$$

The probability of one-step transition from s to k is denoted by Eq. (2)

$$P_{sk} = P(P_1 = t|P_0 = s) \quad (2)$$

Eqs. (1) and (2) depicts a transition from one state to another that is time homogeneous. Eq. (3) is used to get the step r. For Markov chain having time-homogeneous characteristics with s saturation

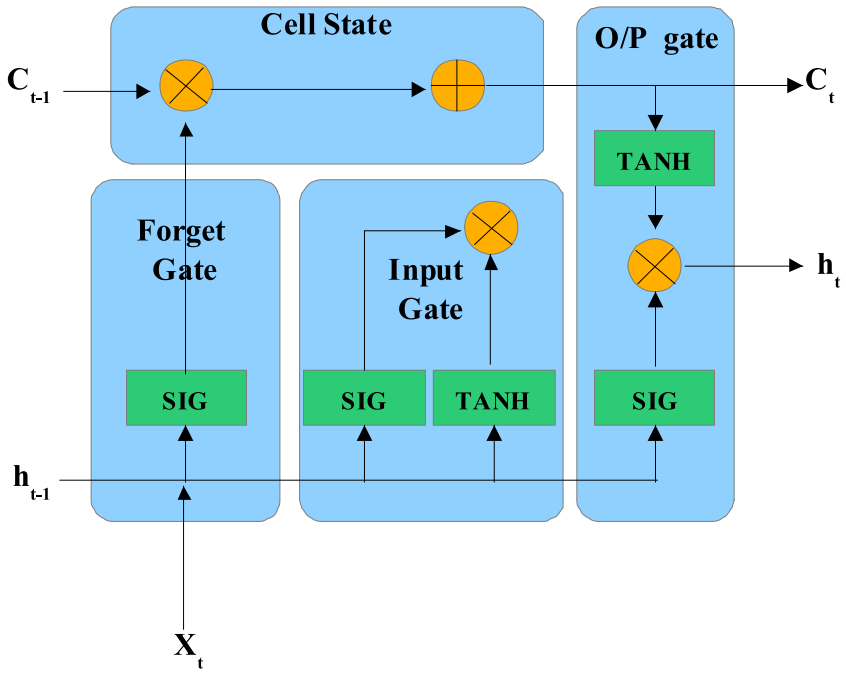

Fig. 4. LSTM anomaly and outlier detection approach.

state

$$P_r(P_n = t) = \sum P_{rt} P_r(P_{n-1} = r) \tag{3}$$

The r steps generalized probability is mentioned in Eq. (4)

$$P_r(P_n = t) = \sum P_{rt} P_r(P_0 = r) \tag{4}$$

The probability P of transition from one state to other is represented by the matrix given in Eq. (5).

$$P = \begin{array}{c} \\ S1 \\ S2 \\ S3 \end{array} \begin{array}{c} \begin{array}{ccc} S1 & S2 & S3 \end{array} \\ \begin{pmatrix} P_{r11} & P_{r12} & P_{r13} \\ P_{r21} & P_{r22} & P_{r23} \\ P_{r31} & P_{r32} & P_{r33} \end{pmatrix} \end{array} \tag{5}$$

3.3. Long short-term memory (LSTM) architecture

The LSTM architecture is mainly used to predict the near future values in the time series data. Fig. 4 elucidates the LSTM based anomaly and outlier detection in the IoT sensor data.

$\mathbf{X_t}$—Current input
h_{t-1}, h_t—Hidden states
C_{t-1}, C_t—Cell states
$\oplus$—Addition
$\otimes$—Multiplication
C_{t-1}, C_t—Cell states

The sequence prediction problems are solved through an approach called recurrent neural network also called as LSTM networks. LSTMs use a gating mechanism that controls the memorizing process. It mainly contains 3 gates Forget gate, Input gate and Output gate. These gates undergo element wise multiplication by sigmoid function that ranges between 0 and 1,

The main functions used are "sig" and "tanh". The "tanh" function is a non-linear activation function [25]. The values returned by this function range from −1 to 1. The "sig" function, which is a sigmoid function, keeps values between 0 and 1 constant [26]. It aids the model's ability to update or forget data.

The forget gate determines which input values should be saved and which should be ignored. The sigmoid function is used to pass the Xt, input state, and previous hidden state, and the output is in the range of 0 to 1 and is passed through point-by-point multiplication with the previous cell state [27]. The input gate obtains the current state $X(t)$ as well as the previous hidden state $h(t-1)$ and passes them through the second sigmoid function. Between 0 and 1 are obtained the values. The same set of data is also passed through the "tanh" function, which performs and keeps the output in the −1 to 1 range [28]. The output of the 2nd sigmoid function as well as the "tanh" function is ready to be multiplied.

The output from the forget gate is multiplied by the previous cell state in the cell state. The input gate's output is then passed through point-by-point addition, which adds both values and stores them in a new cell state $C(t)$. As a result, the output gate has enough data from the forget gate and input gate to create a new cell state [29]. It determines the next hidden state h's output (t). It obtains the required values by passing the values from the current state $X(t)$ and the previous hidden state $h(t-1)$ through the sigmoid function. The new cell state value is then passed through the tanh function, yielding vector values. Finally, these two values are multiplied point by point, with the result being the value of the next hidden state $h(t)$. The forget gate determines in advance which relevant values are required. The input gate determines what information can be extracted from the current and previous hidden states that are relevant. Finally, the output state determines the hidden state that will be used for prediction.

The time t mapping of xt to ht is shared in Eq. (6).

$$h_t = f(h_{t-1}, x_t) \tag{6}$$

Present time t and past time $t-1$ hidden states are represented as h_t, h_{t-1}. The gradient disappearance and explosion that occur in ordinary RNNs can be avoided with LSTM, a common version of RNNs. The forget gate, input gate, and output gate make up the three gates that make up the LSTM cell unit structure, as shown in Fig. 4. The forget gate is used to regulate how much of the preceding cell state's information is lost [30]. The input gate is used to regulate how much current input information is

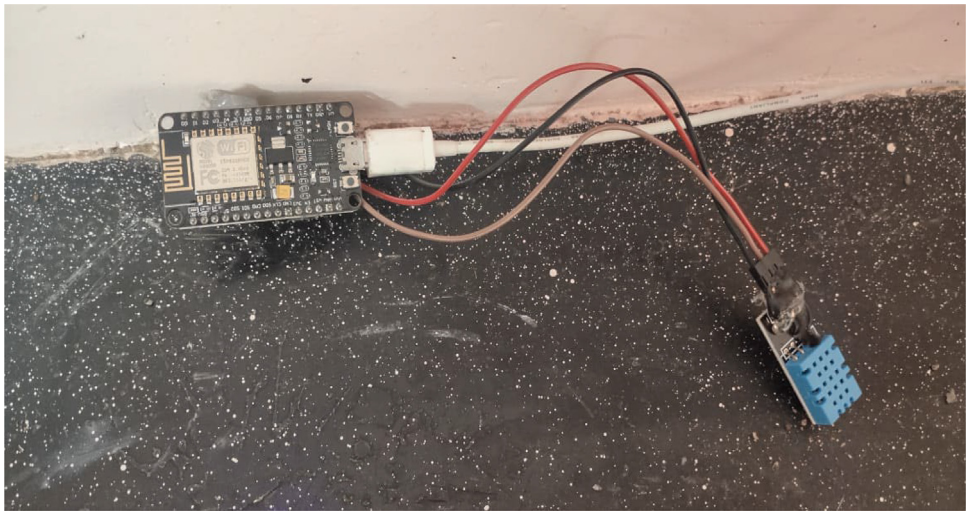

Fig. 5. Real time sensor deployment location 1.

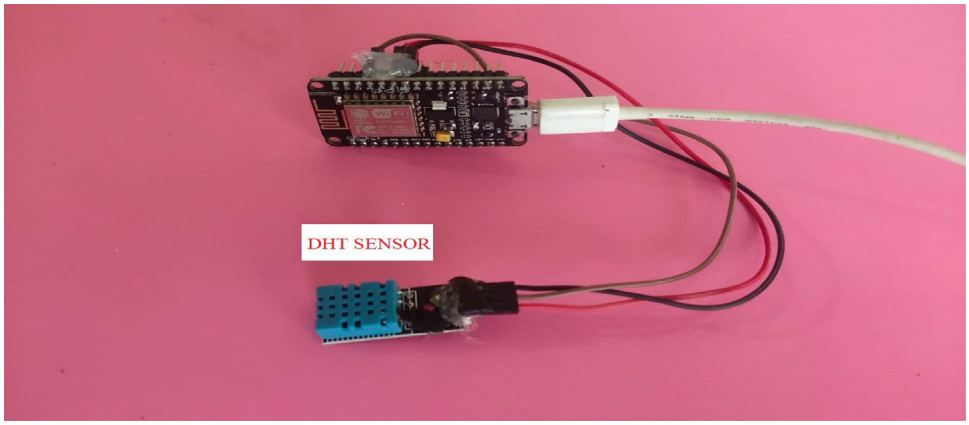

Fig. 6. Real time sensor deployment location 2.

contributed to the current state of the cell. The output gate is used to regulate how much data about the present state of the cell is output. The LSTM cell units' calculation algorithms from the input to the output are as follows. Eqs. (7)–(12) provides the input and output of the proposed LSTM based approach.

$$f_t = \sigma(W_f[h_{t-1}, x_t'] + b_f) \tag{7}$$

$$i_t = \sigma(W_i[h_{t-1}, x_t'] + b_i) \tag{8}$$

$$c_t' = tanh(W_c[h_{t-1}, x_t'] + b_c) \tag{9}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot c_t' \tag{10}$$

$$o_t = \sigma(W_o[h_{t-1}, x_t'] + b_o) \tag{11}$$

$$h_t = o_t \odot tanh(c_t) \tag{12}$$

The input to the LSTM is given from the Markov approach to enhance the prediction accuracy.

4. Results and discussion

Detection of anomaly is mainly used in so many data mining applications to perfectly detect the unusual activities by many data sources. The detection of anomalies is in great demand and with the ML based algorithms produce more perfect findings. Anomaly detection is no longer confined to detecting fraudulent customer behavior; it is increasingly being used extensively in industrial applications.

The anomaly detection approach is employed in machining industries where sensors are used to predict anomalous machine activity based on data obtained from sensors. This information was gathered from the sensors of a major industrial machine's interior component. Figs. 5 and 6 showcase the real-time sensor deployment in the Location 1 and 2. The DHT sensors provide, Temperature and humidity data of the adjacent locations. Figs. 5 and 6 showcase the real-time module with nodemcu and sensors collecting temperature and humidity data via mqtt protocol.

4.1. Real time sensor data collection

A hidden state is built into the structure of the recurrent neural network. The input at time t depends on the hidden state at time t1, and the hidden state at time t2 determines the hidden state at time t1. Most of the time, this hidden state is set to 0 at time 0. For each training and test data point, the unrolling function below will make a list of 50 data points that came before it. The accuracy of the test data and train data in relation to the iterations are shown in Fig. 7.

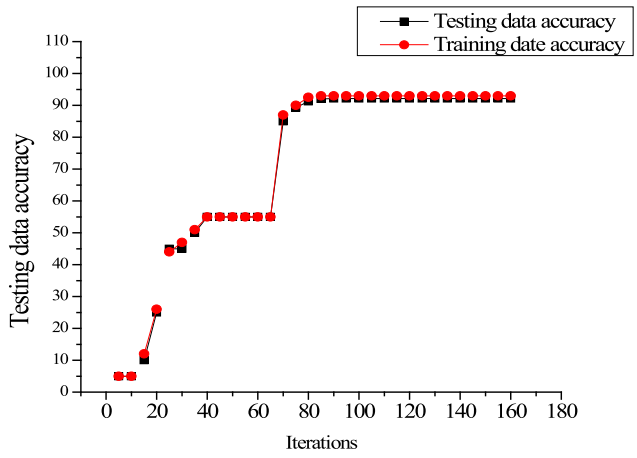

Fig. 7. Train and test accuracy of the real time sensor data.

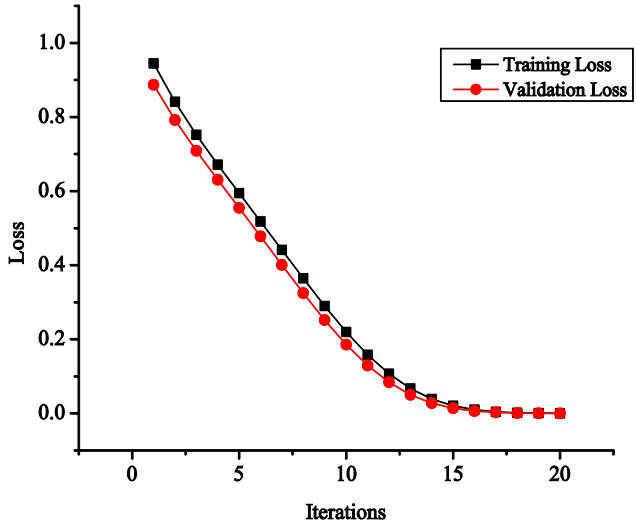

Fig. 8. Training loss and validation of real time sensor data.

Fig. 8 provides the training loss and validation loss of the sensor data acquired from the esp-nodemcu connected with the DHT sensor. About 1lakh data points is collected through the nodemcu setup and the anomalies in the live data is identified. The testing accuracy increases within 100 data iterations points in case of the proposed algorithm.

Fig. 9 shows the real time humidity data. The anomalies due to disturbance are accurately identified through LSTM approach and avoided. The anomaly detection is achieved through accurate predictions of live data.

Fig. 10 provides the effective anomaly detection in temperature data.

Fig. 11 showcase the training accuracy and efficiency in anomaly detection of the proposed algorithm when compared with the KNN algorithm. The proposed methodology provides 96.03% effective anomalies rejections with 92.48% high training accuracy. The methodology showcase improved with 6.54% of effective anomaly detection and 5.13% of training accuracy when compared with KNN algorithm.

In order to evaluate the LSTM's prediction accuracy, we choose the rootmean-square logarithmic error (RMSLE) as our loss function to minimize the difference between the true value yT and the projected value y'T. The definition of the loss function is given

Table 2
Comparison between KNN and LSTM-Markov.

Algorithm	Training accuracy	Efficiency
KNN	87.35%	89.49%
LSTM-MARKOV	92.48%	96.03%

Table 3
Accuracy comparison of KNN and LSTM-MARKOV.

Algorithm	RMSLE	MAE	R^2
KNN	5.32%	8.34%	6.12%
LSTM-MARKOV	3.12%	4.01%	1.96%

RMSLE.

$$\text{RMSLE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\log(y_i' + 1) - \log(y_i + 1))}$$

where the real values y stands in for the projected values y'. The data's order of magnitude may be quite huge, which provides the justification. The introduction of a log function may reduce the effect of these values on the total errors when there is a significant difference between a small number of projected values and actual values in the data. Fig. 12 shows that the RMSLE algorithm provides improved results and reduced RMSLE values when compared with the KNN algorithms.

The mean absolute error (MAE) of the algorithm is given in Fig. 13. The proposed algorithm shows improved reduced results and high accuracy when compared with the KNN algorithm. The determination coefficient MAEcalculation for the time series data is given by

$$\text{MAE} = \frac{1}{n} - \sum_{i=1}^{n}|y_i' - y_i|$$

Fig. 13 showcase the MAE of the proposed algorithm with the KNN approach. The algorithm proposed provided reduced MAE when compared with the KNN approach.

Fig. 14 shows the comparison of determination coefficient of the proposed algorithm with the KNN approach. The algorithm provides reduced coefficient and higher values of accuracy when compared with the KNN approach. The determination coefficient is calculated based on proportional squares of residuals to the total values.

Table 2 provides detailed accuracy comparison of the proposed with the KNN algorithm, in all cases the proposed LSTM-MARKOV approach provides better results when compared with the KNN approach. The accuracy comparison of KNN and LSTM-MARKOV shows a Table 3.

The proposed approach is tested with the light weight data set such as temperature, humidity sensors. The data validation with high data sensors are to be experimented the proposed algorithm is light weight protocols.

5. Conclusion

Efficient anomaly detection could save more computations and effective decision making inside the smart environment. The proposed LSTM markov based anomaly detection provides better detection and high accuracy when compared with the KNN based anomaly detection. The model proposed also filters anomaly data due to fault sensor patterns. The methodology is tested with real

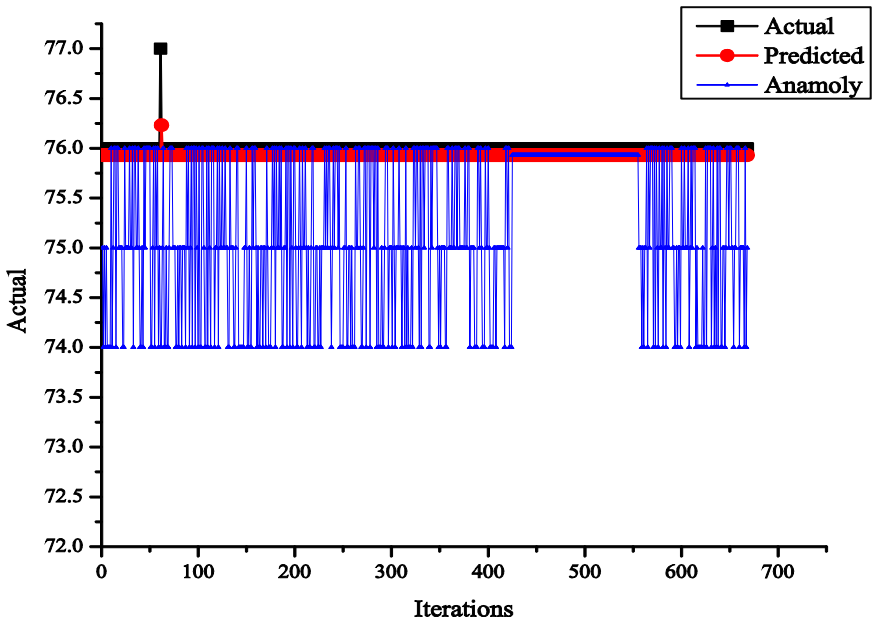

Fig. 9. Anomaly detection in Humidity data from the sensors.

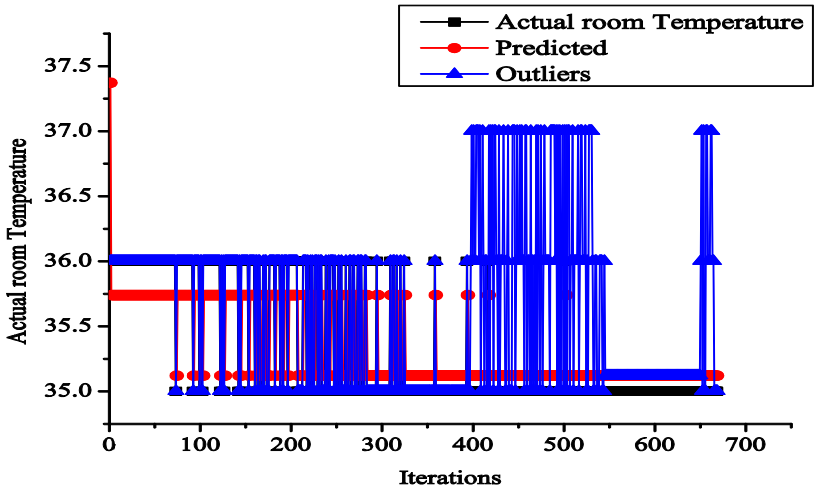

Fig. 10. Room temperature anomaly detection.

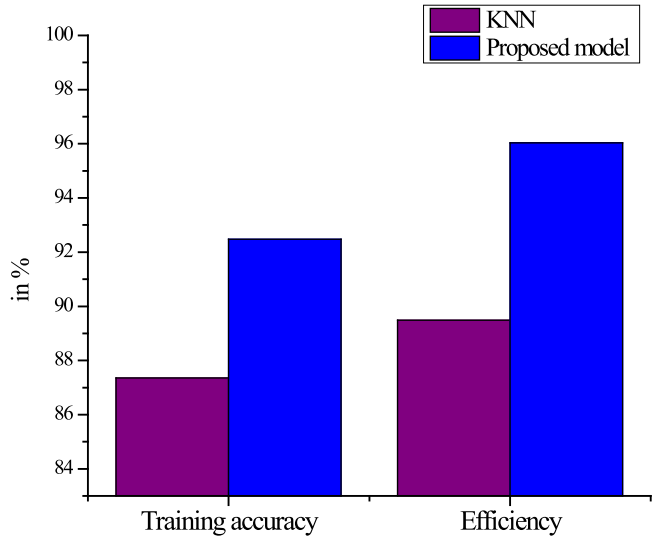

Fig. 11. Training accuracy and efficiency comparison.

time IoT setup to acquire real-time data. The proposed methodology provides 96.03% effective anomalies detection with 92.48% high training accuracy. The methodology showcase improved with 6.54% of effective anomaly detection and 5.13% of training accuracy and when compared with KNN algorithm. The approach also provides reduced RMSLE, MAE and R^2 parameters when compared with the KNN algorithm. The algorithm provides 3.12, 4.01 and 1.96% of RMSLE, MAE and R^2, where as in case of KNN it is 5.32, 8.34 and 6.12%. The proposed approach can be used to identify anomalies in all-timeseries-based sensordata in the smart city environment. The algorithm can be tested with high data sensors and fast response sensors. The algorithm can be tested for time critical sensor events as a future perspective.The algorithm implementation for high scale smart city environment and its consistency to detect anomalies in huge environment can be verified. The sensitivity analysis of the algorithm for high data set such as audio, video and sensor data such as lidar, radar can be verified for ADAS system.

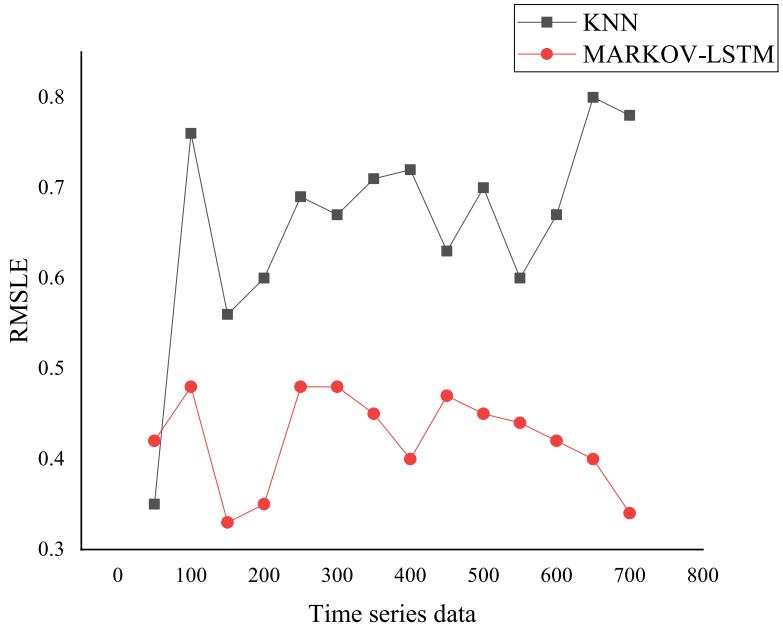

Fig. 12. RMSLE comparison of the KNN and Markov-LSTM.

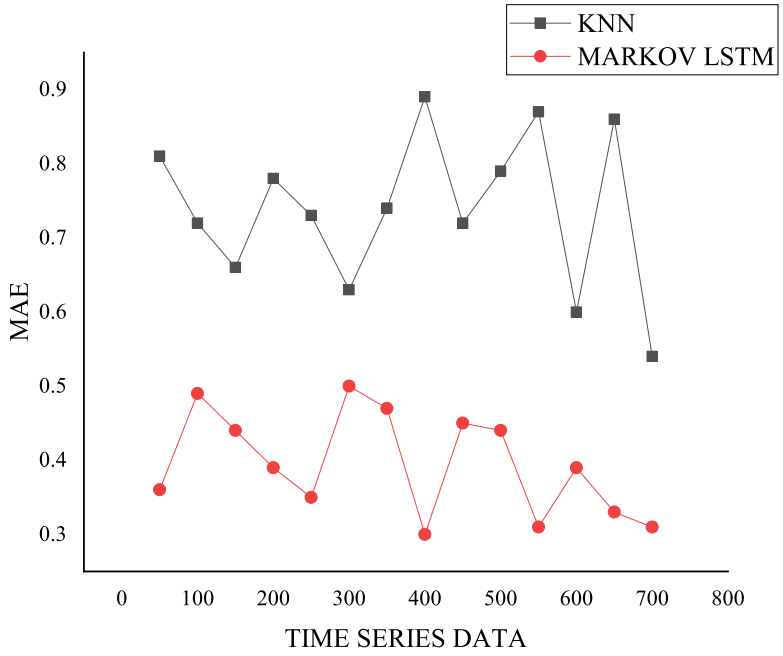

Fig. 13. MAE comparison of algorithm.

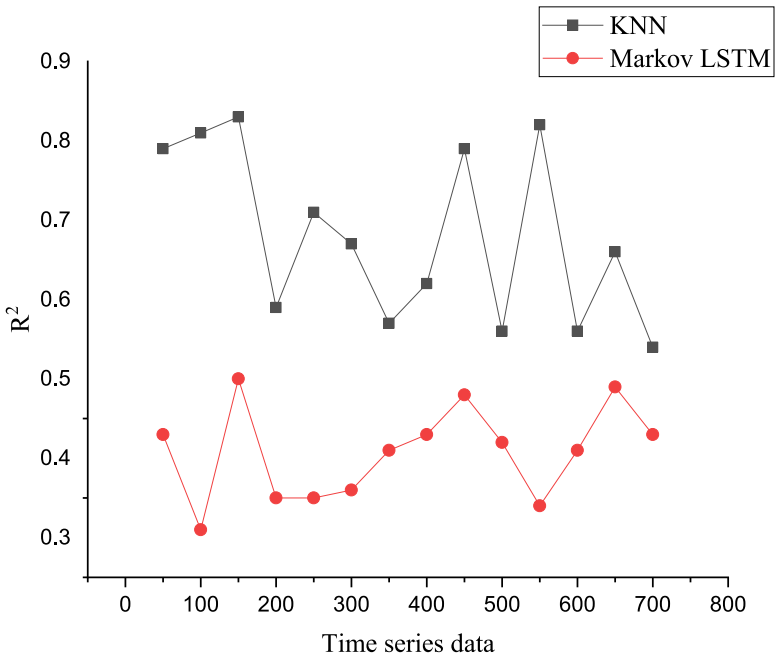

Fig. 14. R^2 comparison of the algorithms.

CRediT authorship contribution statement

Shanmuganathan V.: Conceptualization, Methodology, Writing – original draft. **Suresh A.:** Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] Y. Himeur, K. Ghanem, A. Alsalemi, F. Bensaali, A. Amira, Artificial intelligence based anomaly detection of energy consumption in buildings: A review, current trends and new perspectives, Appl. Energy 287 (2021) 116601.
[2] N.M. Shagari, M.Y.I. Idris, R.B. Salleh, I. Ahmedy, G. Murtaza, H.A. Shehadeh, Heterogeneous energy and traffic aware sleep-awake cluster-based routing protocol for wireless sensor network, IEEE Access 8 (2020) 12232–12252.
[3] Z. Cao, G. Lu, S-AODV: Sink routing table over AODV routing protocol for 6LoWPAN, in: 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing. Vol. 2, IEEE, 2010, pp. 340–343.
[4] D. Sharma, A.P. Bhondekar, Traffic and energy aware routing for heterogeneous wireless sensor networks, IEEE Commun. Lett. 22 (8) (2018) 1608–1611.
[5] G. Ahmed, J. Zou, M.M.S. Fareed, M. Zeeshan, Sleep-awake energy efficient distributed clustering algorithm for wireless sensor networks, Comput. Electr. Eng. 56 (2016) 385–398.
[6] M. Thangamani, P. Thangaraj, Fuzzy ontology for distributed document clustering based on genetic algorithm, Appl. Math. Inf. Sci. 7 (4) (2013) 1563–1574.
[7] V. Mahima, A. Chitra, Battery recovery based lifetime enhancement (BRLE) algorithm for wireless sensor network, Wirel. Pers. Commun. 97 (4) (2017) 6541–6557.
[8] P. Sethi, S.R. Sarangi, Internet of things: architectures, protocols, and applications, J. Electr. Comput. Eng. 2017 (2017).
[9] H. Xue, D.Q. Huynh, M. Reynolds, SS-LSTM: A hierarchical LSTM model for pedestrian trajectory prediction, in: 2018 IEEE Winter Conference on Applications of Computer Vision, WACV, IEEE, 2018, pp. 1186–1194.
[10] Q. Zhou, X. Shi, L. Ge, Predicting mental disorder from noisy questionnaires: an anomaly detection approach based on keyword extraction and machine learning techniques, J. Intell. Fuzzy Syst. (2021) 1–13, (Preprint).
[11] M. Zamini, S.M.H. Hasheminejad, A comprehensive survey of anomaly detection in banking, wireless sensor networks, social networks, and healthcare, Intell. Decis. Technol. 13 (2) (2019) 229–270.
[12] G. Wu, Z. Zhao, G. Fu, H. Wang, Y. Wang, Z. Wang, J. Hou, L. Huang, A fast kNN-based approach for time sensitive anomaly detection over data streams, in: International Conference on Computational Science, Springer, Cham, 2019, pp. 59–74.
[13] H. Peng, L. Liu, J. Liu, J.R. Lewis, Network traffic anomaly detection algorithm using mahout classifier, J. Intell. Fuzzy Systems 37 (1) (2019) 137–144.
[14] S. Siami-Namini, N. Tavakoli, A.S. Namin, The performance of LSTM and BiLSTM in forecasting time series, in: 2019 IEEE International Conference on Big Data, Big Data, IEEE, 2019, pp. 3285–3292.
[15] C. Zhou, R.C. Paffenroth, Anomaly detection with robust deep autoencoders, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp. 665–674.
[16] A. Arronte Alvarez, F. Gómez, Motivic pattern classification of music audio signals combining residual and LSTM networks, 2021.
[17] R. Xu, Y. Cheng, Z. Liu, Y. Xie, Y. Yang, Improved long short-term memory based anomaly detection with concept drift adaptive method for supporting IoT services, Future Gener. Comput. Syst. 112 (2020) 228–242.
[18] J. Contreras-Castillo, S. Zeadally, J.A. Guerrero Ibáñez, A seven-layered model architecture for internet of vehicles, J. Inform. Telecommun. 1 (1) (2017) 4–22.
[19] Z. Chen, C.K. Yeo, B.S. Lee, C.T. Lau, Autoencoder-based network anomaly detection, in: 2018 Wireless telecommunications symposium, WTS, IEEE, 2018, pp. 1–5.
[20] M. Deore, U. Kulkarni, Mdfrcnn: Malware detection using faster region proposals convolution neural network, 2022.
[21] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A.Y. Zomaya, R. Ranjan, A hybrid deep learning-based model for anomaly detection in cloud datacenter networks, IEEE Trans. Netw. Serv. Manag. 16 (3) (2019) 924–935.
[22] N. Moustafa, J. Hu, J. Slay, A holistic review of network anomaly detection systems: A comprehensive survey, J. Netw. Comput. Appl. 128 (2019) 33–55.
[23] S. Garg, K. Kaur, S. Batra, G. Kaddoum, N. Kumar, A. Boukerche, A multi-stage anomaly detection scheme for augmenting the security in IoT-enabled applications, Future Gener. Comput. Syst. 104 (2020) 105–118.
[24] B. Wang, Y. Chen, D. Liu, X. Peng, An embedded intelligent system for online anomaly detection of unmanned aerial vehicle, J. Intell. Fuzzy Systems 34 (6) (2018) 3535–3545.

[25] F. Cauteruccio, L. Cinelli, E. Corradini, G. Terracina, D. Ursino, L. Virgili, C. Savaglio, A. Liotta, G. Fortino, A framework for anomaly detection and classification in multiple IoT scenarios, Future Gener. Comput. Syst. 114 (2021) 322–335.
[26] E. Roberts, B.A. Bassett, M. Lochner, Bayesian anomaly detection and classification for noisy data, Int. J. Hybrid Intell. Syst. 16 (4) (2020) 207–222.
[27] W. Eberle, L. Holder, Anomaly detection in data represented as graphs, Intell. Data Anal. 11 (6) (2007) 663–689.
[28] A. Aljuhani, Machine learning approaches for combating distributed denial of service attacks in modern networking environments, IEEE Access 9 (2021) 42236–42264.
[29] S. Bansod, A. Nandedkar, Transfer learning for video anomaly detection, J. Intell. Fuzzy Systems 36 (3) (2019) 1967–1975.
[30] T. Pourhabibi, K.L. Ong, B.H. Kam, Y.L. Boo, Fraud detection: A systematic literature review of graph-based anomaly detection approaches, Decis. Support Syst. 133 (2020) 113303.