# Time Series Forecasting and Point Anomaly Detection of Sensor Signals Using LSTM Neural Network Architectures


**Antham Bhoomika**
Department of Computer Science and Engineering
Maturi Venkata Subba Rao (MVSR) Engineering College
bhoomika.antham@gmail.com

**Sri Naga Snigdha Chitta**
Department of Computer Science and Engineering
Maturi Venkata Subba Rao (MVSR) Engineering College
srinagasnigdha9@gmail.com

**Karthik Laxmisetti**
Department of Computer Science and Engineering
Maturi Venkata Subba Rao (MVSR) Engineering College
karthiklaxmisetti@gmail.com

**B Sirisha**
Department of Computer Science and Engineering
Maturi Venkata Subba Rao (MVSR) Engineering College
sirishavamsi@gmail.com



***Abstract*—The sensor data points that exhibit unexpected behaviour that considerably deviates from the norm are considered anomalies in the time-series sensor data. We can model univariate time series data using a number of traditional methods, including ARIMA, GARCH, SARIMA and VAR. Modern deep learning algorithms have lately been used to study time series analysis and prediction. This study contrasts three autoregressive models, which project future events based on past observations. Three different sets of temperature, vibration, and pressure sensor data are used to compare the performance of the Bidi- rectional Long Short-Term Memory (Bi-LSTM), Convolutional Long Short-Term Memory (C-LSTM), and Stacked Long Short- Term Memory (S-LSTM) architectures. The models performance and training time are reported for healthy, unhealthy and noisy time-series data. Experimental results shows that, Model trained and build using Bi-LSTM can consistently detect point anomaly for healthy, unhealthy and noisy time-series data with minimum error rate across three sensor datasets.**



***Index Terms*—Time Series, Sensor signals, RNN, LSTM, Forecasting, Anomaly Detection**


## I. INTRODUCTION

With the rapid development of sensor data acquisition and data storage and processing technology, the vast amount of data is growing exponentially. Across manufacturing, the private and public sectors are wasting time and money, whether due to wasted energy, airflow, temperature control, or very poor utilization of equipment and capacity. Here the Internet of Things prediction initiative will play an important role, helping the smart industry to plan and operate efficiently. Internet of Things based on temperature prediction has already been used very successfully in the smart industry to manage energy consumption efficiently and signal prediction from pressure, temperature and vibration sensors has been used in the avionics industry for aircraft engine control and it is already very successfully used to manage. Figure[1] shows the signal pattern of Time series data acquired from five Pressure Sensors (P1 to P5). This article specifically deal with time series analysis (forecasting and anomaly detection) for the Internet of Things. Time series data is typical of physical sensor data from smart electricity meters, temperature sensors, pressure sensors, volumetric flow sensors and vibration sensors in the smart industry to predict or monitor system conditions. Collecting, processing and storing time-series data from Internet of Things (IoT) sensors can be very challenging due to the sheer volume of it. Today's research focuses on how to analyze time series data after collection to make predictions and detect anomalies. Time-series sensor data analysis, forecasting are rapidly expanding and gaining popularity as a result of their ability to achieve high accuracy in anomaly detection of sensor data signals in IoT domain[1]. Time-series data is a set of values or readings taken sequentially through the time. The mathematical representation of Time series data is shown below

$$T_e(f); [e = 1, 2, .....g; f = 1, 2, .....t]; \qquad (1)$$

e= Index value of various readings at each time unit-f t= No. of observed Variables

g= No. of observations

In time-series analysis, the prime variable is time. We can examine this data with the aim to extract worthy statistics and other characteristics. The prime goal of time-series data analysis is to forecast the upcoming data points or values based on previous values or data points. If the time series data has single variable, it is termed as uni-variate(i.e. g=1). It is principally the eloquent examination of a single variable used to define characteristics of a given sample. If the time series data has single variable, it is termed as uni-variate(i.e. g=1). If the data has more than variable i.e. $g > 1$, it is termed as multi-variate.

## II. TIME SERIES ANOMALIES

Anomaly, in time-series data are the data points at time step(s) that show unexpected behavior that differs significantly from the norm which is also referred to as an outlier. Detection of anomaly mainly aims to identify the observations that do not follow expected behavior. These observations are termed as anomalies, novelty, outliers, exceptions and surprises in varied application domains. The most common term in literature are outliers and anomalies. There are three categories of anomalies [See Figure[2]]:

- Point Anomalies: Is also called as global outliers, these anomalies exist very far i,e, outward to the entireness of a data set.
- Contextual or conditional Anomalies: The data points or observations that deviates considerably from the other points in the identical context.
- Collective Anomalies: A subset of observations among a set is abnormal to the entire observations or dataset, those points are named collective outliers.

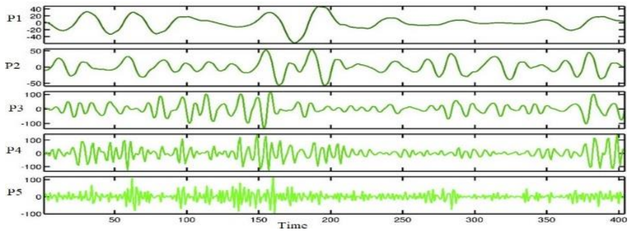


Fig. 1. Time series data Acquired from varied Pressure Sensors-P1 to P5

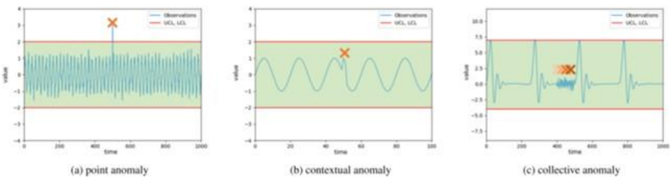


Fig. 2. Different Types of Time Series Anomalies in Sensor Data

*METHODS OF ANOMALY DETECTION:* Model Normal time series data: The data which is available for normal operating conditions of system is used and a model is trained. When there is a difference in the observation from the expected be- havior (as predicted by the model), it is labelled as an anomaly based on the severity of the discrepancy. The advantage we obtain is that the samples of data belonging to fault conditions is not necessary, Unsupervised learning. However, it is difficult to differentiate between noise and fault, leading to high false positives or low detection recall often in these approaches. The methods can be classified as supervised, Unsupervised and semi-supervised approach,

The architecture of deep neural networks with long-short-term memory makes a significant contribution to solving real-world applications in multiple domains. Though Feed forward models perform very well for applications related to prediction, they have several setbacks. In FFNs, data is fed in a forward direction, so interdependent data between past and future values cannot be captured [2][3]. Recurrent neural networks (RNNs) are used to extract interdependent features in time series data. These network models though widely employed in time series data, they fail to extract the interdependencies between data points of long-range time series signal [4]. Long Short-Term Memory (LSTM) overcomes this setback, by extracting interdependency features from static as well dynamic long range data. The competence to attain numerous levels of data visualization that complement the pigeon-holed features of the tailor-made interactive architecture is the prime characteristics of deep learning (DL). The main feature of this model is to learn the temporal interdependencies of hetero and homogeneous data and extract interdependent features from short as well as long range time series signals. In this paper, we implement a stacked bi-LSTM model. Stacked bidirectional LSTM is proposed for the purpose of predicting and detecting anomalies in sensor time series data. The architecture consists of a stack of different Bidirectional LSTM layers, each containing a large number of neural networks NN. The model is tested with help of 3 sensor dataset. The paper focuses in

- Identify the optimal number of neurons and epochs required to build a robust model.
- Identify appropriate LSTM model for point anomaly detection in univariate time series signal data.

The rest of the paper is catalogued as follows, Section 2, presents Time series Anomalies and related work of time series anomaly detection. In Section 3 we introduce the proposed time series forecasting and point anomaly detection model architectures In Section 4 presents the experimental results and discussion and Section 5 concludes the work.

## III. Related Work

Review of the state-of-the-art existing techniques for detecting time series point anomalies in sensors and other identical fields. Sebastian Schmidl et al. [4] classified Anomaly detection into 5 categories based on algorithms in research areas such as 1) Deep Learning,2) Statistics,3) Classic Machine Learning,4) Outlier Detection and 5) Data Mining. Univariate Time series data is considered for evaluation. Hence the input dimensionality is univariate. Based on the learning type of model author grouped them under 3 categories as 1) Unsupervised Learning, 2) Supervised Learning and 3) Semi-supervised Learning. On top of this he also categorized anomaly detection algorithms into 6 diverse areas as 1) forecasting, 2) reconstruction, 3) distance, 4) encoding, 5) distribution and 6) trees method. Based on learning types the author sub- divided research areas along with algorithms and family

:

- Unsupervised Learning

1) Classic ML: NoveltySVR [distance], PS-SVM[distance]
2) Data Mining: Ensemble, GI[encoding], GrammerViz [encoding] ,HOTSAX[distance], TSBitmap [encoding], NormA- SJ[distance], SAND[distance], Series2Graph[encoding], STAMP[distance], STOMP[distance], VALMOD[distance],Left STAMPi [distance], SSA[distance], PST[trees]
3) Deep Learning:NumentaHTM[forecasting]
4) Outlier Detection:Sub-LOF[distance],Sub-IF[trees]
5) Statistics:S-H-ESD [distribution], DSPOT [distribution], ARIMA [forecasting], MedianMethod[forecasting], SARIMA[forecasting]ES[forecasting],PCI[reconstruction]

- Semi-supervised Learning

1) Classic ML: RForest[forecasting], XGBoosting[forecasting]
2) Data Mining: TARZAN[encoding]
3) Deep Learning: HealthESN [forecasting], OceanWNN [forecasting], Bagel[reconstruction],Donut[reconstruction],IE- CAE[reconstruction],SR-CNN[reconstruction], LSTM[forecasting]
4) Statistics:Sub-Fast-MCD [distribution]

Sebastian Schmidl et al. used several anomaly detection algo- rithms on different time series datasets and gave an overview of strength and weakness of algorithm selection in different families.

Zahra Zamanzadeh Darban et al. [5] gave a clear idea on deep anomaly detection methods such as 1) Time series Anomaly Detection 2)Forecasting-based models:RNN(Recurrent Neural Network),CNN(Convolution Neural Network),GNN(Graph Neural Net- work),HTM(Hierarchical Temporal Memory),Transformers 3)Reconstruction-based models: Autoencoders(AE),Variational Autoencoder(VAE),Generative adversarial net- work(GAN),Transformers 4)Hybrid models He explained application areas of deep anomaly detection in time series. Benjamin Lindemann et al. [6] demonstrated hybrid approaches of two neural networks. He explained different LSTM based approaches for Anomaly Detection. He gave clear idea about graph-based approaches and transfer learning approaches in learning based Anomaly Detection. Nouar AIDahoul et al. [7] demonstrated on Network Anomaly Detection(NAD).He explained about some NAD methods and Correlational Paraconsistent Machine(CPM),Support Vector Machine(SVM) and some algorithms and they are used to detect NAD which is internal part of Network Behaviour Analysis(NBA).He used two deep neural networks for model fusion.He used binary model and multi-class model which includes feature pre-processing and DNN(Deep Neural Network) here DNNs are used to classify or categorize the traffic data. H. Izakian et al. [8] used a reconstruction convention to reconstruct the optimal cluster based on Fuzzy C-means and partition matrix based on a multivariate data. To detect anomalies in multivariate data they used the reconstruction error to improve robustness of Particle Swarm Optimization (PSO) algorithm but due issue in algorithm trap in local optimum it is unable to let out the high dimensional multivariate time series.

A. Mondal et al. [9] proposed an inexpensive structure named Indo Air Sense for estimation and forecasting of in- door air quality in classrooms of a university. They initially used Multi-Layer Perceptron (MLP) and extreme Gradient Boosting Regression (XGBR) for the estimation of air quality. Then they used LSTM (Long Short Term Memory) without a forget gate to decrease the complication to forecast indoor air pollution. Since they did not use the forget gate to keep long term memory, their model with time series data is unable to detect anomalies. S. M. H. Zaidi et al. [10] proposed a model to predict the concentration of mixed air pollutants which scrutinize the comprehensive quality of an indoor ambience based on LSTM. In their evaluation they gathered the concentration of NH3,CO etc data using IoT sensors. Their model has the ability to detect the anomalies in air quality and it can send the caution message. T. Kang et al. [11] sensed the temperature, humidity and brightness of an indoor space using 3 IoT sensors and used LSTM to detect the abnormal deviation of indoor conditions from a threshold while predicting the indoor space facility management. M. A. U. Shariff et al.

[12] proposed a model with the combination of two kinds of RNN model i.e., GRU and LSTM respectively) to predict and forecast the day-to-day Air Quality Index (AQI). In their model they used GRU as the first dense layer and LSTM as the second hidden layer in the model followed by 2 dense layers. Their results depicted that instead of a single model of GRU or LSTM their combined model improved the overall performance of AQI.

J. Jang-Jaccard et al. [13] constructed a hybrid model which detects anomalies in indoor space. They initially used Mean and Standard Deviation (MSD) to remove noisy data by which the noise formed by clustering. Then they applied the K- means algorithm to get better optimal clustering. P. Kumar et al. [14] presented Auto Regressive Integrated Moving Average (ARIMA) and k-Nearest Neighbour (KNN) techniques to detect contextual and point anomalies in an air quality dataset. They used Euclidean distance to find the similarity between points and find the anomaly to each individual point. They used ARIMA to detect contextual anomalies. They found both point and contextual anomalies and further classified as anomaly and normal with the use of K-means clustering algorithm. H. Chen et al. [15] proposed a model with an add on error correction model (ECM) along with LSTM such that to improve the anomalous temperature in a building. They trained the model with predicted and calculated data and improved the overall performance in the prediction of testing data.

Kyle Hundman et al. [16] demonstrated about 3 categories of anomaly detection as point, contextual and collective. He gave ideas on Dynamic Error Thresholds after Prediction with LSTMs. They found the point and contextual anomalies for the Soil Moisture Active Passive (SMAP) satellite and the Mars Science Laboratory (MSL) rover data to overcome the issues using expert-labeled telemetry anomaly. Pankaj Malhotra et al. [17] in his paper gave a clear idea on LSTM-based on Encoder-Decoder for Anomaly Detection. He explained the problem with Encoder-decoder model. He trained the model with normal instances and it learnt to reconstructed them but when he give an anomalous sequence, it lead to higher reconstruction errors since it is not able to reconstruct the anomalous sequence. This is especially useful in cases when anomalous data is not available. He used LSTM Encoder-Decoder as a reconstruction model with a single hidden layer and found the anomaly for multi-sensor. Pavel Filonov et al. [18] in his research paper generated a mathematical model for anomalous data and generated faults in it. He used sequence-to- sequence architecture of LSTM network for forecasting model and considered mean square error as loss function to detect faults in data. He also used precision and recall and F1 scores as metrics to detect faults.

In our research work we employed three LSTM(Long Short-Term Memory) model architectures such as Bidi rectional Long Short-Term Memory(BI-LSTM), Convolution Long Short-Term Memory(C-LSTM) ,Stacked Long Short-

Term Memory(S-LSTM) to forecast and detect point anomaly in multiple sensor data.

## IV. Bi LSTM-AD: Stacked Bidirectional LSTM-based Point Anomaly Detection

In this section firstly we introduce Standard LSTM network architecture followed by LSTM based Anomaly detection and Stacked Bidirectional LSTM forecasting model

### A. *Standard LSTM Network Architecture*

A generic unit of LSTM comprised of a cell unit with three gates, i.e., an input, an output, and a forget gate. The cell unit memorizes data values over a random interval of time, and the three gates control the flow of information (data) into and out of the cell unit. The input data consists of real-time information for the current time step. Cell state refer to the network's long-term memory, which stores a list of previous data. The hidden state deals with previous outputs, hence called short-term memory.

*Step1 : Forget Gate*

Given both the previous hidden state and the current cell state, the forget gate's primary function is to determine which bits should be ignored. To do this, the previously hidden state input and current input data are fed to the cells network, which uses a sigmoidal activation function to generate a vector with elements in each vector on the interval [0,1]. The forget gate portion of the cell is trained to have an output near to 0 when some components of the input are irrelevant and an output near to 1 otherwise. These signal outputs are then transmitted and then multiplied by the previous cell unit state point by point. Mathematically, the forget gate result ($fg_t$) can be expressed as

$$fg_t = \sigma(w_{fg}[H_{t-1}, X_t] + b_{fg}) \tag{2}$$

where $fg_t$ - forget gate, $\sigma$ - activation function,$b_{fg}$ *and* $w_{fg}$ are the bias and weights of the forget gate. $H_{t-1}, X_t$ are the fusion of current input and hidden state respectively.

*Stage2 :LSTM's input gate (IG ) :*

IG first checks if it should keep the previously hidden data points and the new input data points. Next, decide what new data or information to add to the cell state. To do this, the input gate first combines the previous hidden state with the new input data to create a new memory update vector denoted as $Ci'_t$. The tanh enable function is used to obtain the elements of the memory update vector to include values in the range [-1, 1] where the negative values exist. The values are used to mitigate the effects of cell state components. This vector shows how much each cell state component should change in response to fresh information. Equation 2 describes this procedure as follows:

$$Ci'_t = tanh(w_{ic}[H_{t-1}, X_t] + b_{ic}) \tag{3}$$

where tanh is the activation function, $b_{ic}$ *and* $w_{ic}$ are the bias and weights of the input gate. $H_{t-1}, X_t$ are the fusion of current input and hidden state respectively. The IG's next task is to determine, in light of the circumstances and traits of the prior concealed state, which portions of the new input are important enough to remember. The IG is taught to output a vector of values in the [0,1] range using a sigmoidal activation function, just as the forget gate. Outputs near 0 are not updated with the cell state. This is expressed in the below Equation :

$$ig^t_t = \sigma(w_{ig}[H_{t-1}, X_t] + b_{ig}) \tag{4}$$

where $b_{ig}$ *and* $w_{ig}$ are the bias and weights of the input gate. the two functions are multiplied pointwise. This will lead to generate new information whose magnitude is set to the value 0, if it is needed. The vector is combined to the cell state, which results in yielding networks long term memory, shown in equation below

$$C_{i_t} = f_{g_t} \odot C_{i_{t-1}+i_t} \odot C_{i'_t} \tag{5}$$

*Stage-3: Output Gate (OG):*

Once the long-term memory update is complete, next it work on the output gate. This gate determine new hidden states. The OG uses three different types of information for this purpose: most recent input data, previously hidden HS status, and recently updated CS-cell state. Apply the most recent input data and the previous HS- hidden state through a sigmoidal network first to obtain the filter vector. $Og_t$, (See Equation 5).

$$Og^t_t = \sigma(w_{Og}[H_{t-1}, X_t] + b_{Og}) \tag{6}$$

$b_{Og}$ and $w_{Og}$ are the bias and weights of the input gate. The cell state (CS) is delivered through the tanh enable function, where forces the data values to the time interval [-1, 1], producing a squashed cell state. This state is now applied to FV(filter vector) . The new HS $h_t$ and CS $Cs_t$ is generated. $H_t = O_t \odot tanh(C_t)$ The latest cell and hidden state $Cs_t$ and $H_t$ generated will be fed as previous cell and hidden state $Cs_{t-1}$ and $H_{t-1}$)to proceeding LSTM. Let T be the time series data represented by equation-1

$$T = (t^{(1)}, t^{(2)}, t^{(3)}, t^{(4)} ...... t^{(n-1)}, t^{(n)}) \tag{7}$$

where every single data point $T^t \varepsilon R^m$ in the time series. This is a m-vector dimension vector shown in equation-2, whose parameters corresponds to the given input variables.

$$T = \{T^t, T^t, T^t ....... T^t\} \tag{8}$$

### B. *LSTM-based Anomaly Detection*

A forecasting model learns to forecast the consecutive l datapoints for d input variables. The original data sequences are partitioned into four sets of values.

- Healthy Trainset : $S_H$
- Healthy validation-1: $v_{H1}$
- Healthy validation-2: $v_{H2}$
- Healthy Test: $t_H$

The data sequence which is anomalous are partitioned into two sets of values. The data sequence which is noisy in nature (healthy and anamalous) are partitioned into two sets of values.

- Anamalous(A) validation : $v_A$

- Anamalous(A)test : $t_A$
- Noisy validation : $\upsilon_N$
- Noisy test : $t_N$

### C. Stacked Bidirectional LSTM-based Point Anomaly Detec- tion

We have considered three LSTM-based network architec- tures 1. Stacked LSTM 2. Stacked Bidirectional LSTM 3. Convolutional LSTM architecture. Our model forecasts one-time period data values. Hence single unit is taken in the input layer, and the dimension of the data is m, the dimension of the output layer is $dxl$i.e. one unit is employed for every single l future forecast of d dimension. Recurrent neural network connections are employed in the hidden layers of the LSTMs. We pile several LSTM layers i.e. every single unit in a below LSTM HL- hidden layer is connected fully to every single unit of LSTM HL above, through a feed-forward connection. The forecasting model is learned with help of s-sequences in $s_n$. $\upsilon_{n1}$ is employed for stopping while the network weights are learning. The anomaly is detected using the forecasting error distribution. With a forecasting length represented as $l$ each of the identified dimensions $d$ of $x^{(t)} \in X$ for $l < t \leq n - l$ is forcasted l times. We have computed an data point error vector $e^{(t)}$ for data point $x^{(t)}{}_t$ as $t$

$$e^t{}_i = [e^{(t)}{}_{11}, ...., e^{(t)}{}_{1l}, ....., e^{(t)}{}_{d1}, ....., e^{(t)}{}_{dl}] \quad (9)$$

where $e^{(t)}{}_{ji}$ is the data point difference among $x_i{}^t$ and its data value forecasted at time $t - j$. The forecasted model built on $s_n$ will compute the error for every single data point in test and validation data sequence.

## V. Experiments

We built our Stacked Bidirectional LSTM model for sensor data from the Hydraulic sensor dataset, Helwig, Pignanelli, and Schtze introduced this dataset in [19].The data collection focuses on the condition evaluation of a hydraulic test rig using data from multiple sensors. The volume of historic data for the pressure sensor is 4032 , temperature sensor is 7267 and vibration sensor is 22695. We have partitioned the data into two parts one is training set with 75% of data and validation set 25% of data for forecasting the predicted error. For the purpose of anomaly detection, we build three LSTM architecture models: the Bidirectional LSTM (BI-LSTM), the convolutional LSTM (CONVLSTM), and the Stacked LSTM (S-LSTM). To increase the reliability of the results, we trained all the models using three separate real-world datasets that were further divided into good, bad, and noisy categories based on labels of healthy and unhealthy in the dataset.

We categorized the data as healthy data with labels indicat- ing healthy, unhealthy data with labels indicating unhealthy, and noisy as a combination of data with labels indicating healthy and unhealthy. Using min max scalar, we were able to scale the values between 0 and 1 for input data . As we know Mean Square Error is more sensitive to outliers. For projected data, actual data, or validation data, we used Root Mean Square Error to identify any point anomalies. The model's loss function is Mean Squared Error (MSE).We use Root Mean Square Error as a metric for our examination of each dataset and present the point anomaly in the data. We used 2/3rds or almost 66% of the healthy data to train our models. After the model had been trained, we provided the final 1/3rd of the data, predicted the values, and calculated the RMSE using real, reliable data. Using all the models that were trained with healthy data, we determined the RMSE for unhealthy and noisy data for all three datasets. The key observation from Table-1,2 and 3 is stacked bidirectional LSTM can forecast and detect the point anomaly better than convolutional LSTM and Stacked LSTM for temperature, pressure and sensor data.

TABLE I. ROOT MEAN SQUARE ERROR FOR TEMPERATURE SENSOR DATA USING THREE POPULAR LSTM MODELS.

| Temperature Sensor Data | Healthy | Unhealthy | Noisy |
|---|---|---|---|
| Stacked Bidirectional -LSTM | 0.0213 | 0.813 | 0.721 |
| Convolutional LSTM | 0.071 | 0.127 | 0.365 |
| StackedLSTM | 0.048 | 0.258 | 0.315 |

TABLE II. ROOT MEAN SQUARE ERROR FOR PRESSURE SENSOR DATA USING THREE POPULAR LSTM MODELS

| Temperature Sensor Data | Healthy | Unhealthy | Noisy |
|---|---|---|---|
| Stacked Bidirectional -LSTM | 0.0337 | 0.8272 | 0.7424 |
| Convolutional LSTM | 0.0834 | 0.1412 | 0.3864 |
| StackedLSTM | 0.0604 | 0.2722 | 0.3364 |

It is observed from Table-1,2 and 3,compared to Con- volutional and stacked LSTM models, stacked bidirectional LSTM model has the ability to discriminate healthy sensor signal, un-healthy sensor signal as well as the noisy sensor signal(corrupted due to transmission noise).Hence we can infer from the above experiments that stacked, bidirectional LSTM model is highly suitable for detecting point anomalies of univariate signal data. The prime goal of this research work is to identify the optimal number of stacked bidirectional layers need to be added to the base LSTM architecture to construct the model that is efficient. We have experimented by increasing the stacked layers. we experimented stacking of 3, 5 and 7 layers depth. In addition, we analyzed the number of neurons per bidirectional LSTM layer. Initially we have employed 30 neurons and later increased to 60 and 90. The number of epochs has been experimented with to help the model learn more effectively. Our initial epochs were 30 and 60, then we increased them to 90.

Based on empirical evidence, the combination of thirty neurons produces the lowest error rate. After the identification of layers and neurons, we experimented with the number of epochs which helped the model learn better. At 30 epochs, the model learns better.

TABLE III. ROOT MEAN SQUARE ERROR FOR VIBRATION SENSOR DATA USING THREE POPULAR LSTM MODELS.

| Number of Hidden Layers | Number of Neurons | Total Number of Epochs | Root Mean Square Error-RMSE |
|---|---|---|---|
| 3 | 30 | 100 | 0.101 |
| 3 | 60 | 100 | 0.245 |
| 3 | 90 | 100 | 0.288 |
| 3 | 30 | 50 | 0.089 |
| 3 | 60 | 50 | 0.233 |
| 3 | 90 | 50 | 0.276 |
| 3 | 30 | 30 | 0.048 |
| 3 | 60 | 30 | 0.192 |
| 3 | 90 | 30 | 0.235 |
| 5 | 30 | 100 | 0.113 |
| 5 | 60 | 100 | 0.257 |

| | | | |
|---|---|---|---|
| 5 | 90 | 100 | 0.3 |
| 5 | 30 | 50 | 0.101 |
| 5 | 60 | 50 | 0.245 |
| 5 | 90 | 50 | 0.288 |
| **5** | **30** | **30** | **0.006** |
| 5 | 60 | 30 | 0.204 |
| 5 | 90 | 30 | 0.247 |
| 7 | 30 | 100 | 0.197 |
| 7 | 60 | 100 | 0.341 |
| 7 | 90 | 100 | 0.384 |
| 7 | 30 | 50 | 0.185 |
| 7 | 60 | 50 | 0.329 |
| 7 | 90 | 50 | 0.372 |
| 7 | 30 | 30 | 0.144 |
| 7 | 60 | 30 | 0.288 |
| 7 | 90 | 30 | 0.288 |

## VI. Conclusion

A time series forecasting model and a time series anomaly detection model are developed in this paper. In sensor time series data problems, the forecasting model can be applied to a number of univariate anomaly detection applications. Experiments have demonstrated that stacking multiple layers of bidirectional LSTM improves the performance of shallow recurrent neural networks (RNN). Across the three datasets, the proposed forecasting model reduces average RMS error by 2% and 5% compared to stacked LSTM and convolutional LSTM. The model proved to be robust in detecting point anomalies in one-dimensional long-range dependencies, heterogeneous and complex time series data.In addition, the model effectively describes the nonlinearity of the inputs and outputs.

## References


[1] Kadous, M.W.et.al: Classification of Multivariate Time Series and Struc- tured Data Using Constructive Induction. Mach Learn 58,(2005),PP: 179216 .

[2] Xuexia Liu et al:Study on Knowledge-based Intelligent Fault Diagnosis of Hydraulic System”, TELKOMNIKA, Dec 2012, vol-10, pp: 2041-2046.

[3] Alex Nanopoulos et.al: Feature-based Classification of Time-series Data

[4] ,International Journal of Computer Research-2001,vol-10, pp:49-61.

[5] F. Morchen et.al:Time series feature extraction for data mining using DWT and DFT, 2003.

[6] Zahra Zamanzadeh Darban, Geoffrey I. Webb, Shirui Pan, Charu C. Aggarwal, Mahsa Salehi, Deep Learning for Time Series Anomaly Detection:A Survey, arXiv, 2022.

[7] Benjamin Lindemann, Benjamin Maschler, Nada Sahlab, Michael Weyrich, A survey on anomaly detection for technical systems using LSTM networks, Computers in Industry, vol. 131, p. 103498, 2021.

[8] Nouar AIDahoul, Hezerul Abdul Karim and Abdulaziz Saleh Ba Wazir, Model fusion of deep neural networks for anomaly detection, Journal of Big Data, vol. 8, pp.1 - 18, 2021.

[9] J. Li, H. Izakian, W. Pedrycz, and I. Jamal, Clustering-based anomaly detection in multivariate time series data, Applied Soft Computing, vol. 100, p. 106919, 2021.

[10] P. K. Sharma, A. Mondal, S. Jaiswal, M. Saha, S. Nandi, T. De, and S. Saha, IndoAirsense: A framework for indoor air quality estimation and forecasting, Atmospheric Pollution Research, vol. 12, no. 1, pp. 1022, 2021.

[11] R. Mumtaz, S. M. H. Zaidi, M. Z. Shakir, U. Shafi, M. M. Malik, A. Haque, S. Mumtaz, and S. A. R. Zaidi, Internet of things (iot) based indoor air quality sensing and predictive analytica covid-19 perspective, Electronics, vol. 10, no. 2, p. 184, 2021.

[12] Y. Jung, T. Kang, and C. Chun, Anomaly analysis on indoor office spaces for facility management using deep learning methods, Journal of Building Engineering, vol. 43, p. 103139, 2021.

[13] E. Hossain, M. A. U. Shariff, M. S. Hossain, and K. Andersson, A novel deep learning approach to predict air quality index, in Proceedings of International Conference on Trends in Computational and Cognitive Engineering. Springer, 2021, pp. 367381.

[14] Y. Wei, J. Jang-Jaccard, F. Sabrina, and H. Alavizadeh, Large-scale outlier detection for low-cost pm10 sensors, IEEE Access, vol. 8, pp. 229 033229 042, 2020.

[15] T.B. Ottosen and P. Kumar, Outlier detection and gap filling methodologies for low-cost air quality measurements, Environmental Science: Processes and Impacts, vol. 21, no. 4, pp. 701713, 2019.

[16] C. Xu, H. Chen, J. Wang, Y. Guo, and Y. Yuan, Improving prediction performance for indoor temperature in public buildings based on a novel deep learning method, Building and Environment, vol. 148, pp. 128 135, 2019.

[17] Kyle Hundman, Valentino Constantinou, Christopher Laporte, Ian Colwell, Tom Soderstrom, Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding, Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.387 - 395, 2018.

[18] Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, Gautam Shroff, LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection, CoRR, vol. abs/1607.00148, 2016.

[19] Pavel Filonov, Andrey Lavrentyev and Artem Vorontsov, Multivariate Industrial Time Series with Cyber-Attack Simulation: Fault Detection Using an LSTM-based Predictive Data Model, CoRR, vol. abs/1612.06676, 2016.

[20] Nikolai Helwig, Eliseo Pignanelli, Andreas Schtze, ”Condition Monitoring of a Complex Hydraulic System Using Multivariate Statistics”, in Proc. I2MTC-2015 - 2015 IEEE International Instrumentation and Measurement Technology Conference, paper PPS1-39, Pisa, Italy, May 11-14, 2015, doi: 10.1109/I2MTC.2015.7151267.