ISSN 2355-3286

Predictive Maintenance: Automatic Weather Station Sensors Error Detection using Long Short-Term Memory

Bayu Santoso[1,2], Muhammad Ryan[3], Haryas Subyantara Wicaksana[1], Naufal Ananda[1], Irvan Budiawan[4], Faqihza Mukhlish[4], Deddy Kurniadi[1]
[1] Instrumentation and Control Study Program, ITB, Bandung, Indonesia
[2] Department of Instrumentation and Calibration, BMKG, Medan, Indonesia
[3] Center of Aviation Meteorology, BMKG, Jakarta, Indonesia
[4]Engineering Physics Study Program, ITB, Bandung, Indonesia
[1]bayu.santoso@bmkg.go.id, [2]muhammad.ryan@bmkg.go.id, [3]haryas.wicaksana@bmkg.go.id,
[4]naufal.ananda@bmkg.go.id, [5]budiawan.irvan@gmail.com, [6]faqihza.m@itb.ac.id, [7]kurniadi@itb.ac.id

Accepted on 13 November 2023
Approved on 04 December 2023

***Abstract*— Weather information plays a crucial role in various sectors due to Indonesia's wide range of weather and extreme climate phenomena. Automatic Weather Stations (AWS) are automated equipment designed to measure and collect meteorological parameters such as atmospheric pressure, rainfall, relative humidity, atmospheric temperature, wind speed, and wind direction. Occasionally, AWS sensors may produce erroneous values without the technicians' awareness. This study aims to develop sensors error detection system for predictive maintenance on AWS using the Long Short-Term Memory (LSTM) model. The AWS dataset from Jatiwangi, West Java, covering the period from 2017 to 2021, will be utilized in the study. The study revolves around developing and testing four distinct LSTM models dedicated to each sensor: RR, TT, RH, and PP. The research methodology involves a phased approach, encompassing model training on 70% of the available dataset, subsequent validation using 25% of the data, and finally, testing on 5% of the dataset alongside the calibration dataset. Research outcomes demonstrate notably high accuracy, exceeding 90% for the RR, TT, and PP models, while the RH model achieves above 85%. Moreover, the research highlights that Probability of Detection (POD) values generally trend high, surpassing 0.8, with a low False Alarm Rate (FAR), typically below 0.1, except for the RH model. Sensor condition requirements will adhere to the rules set by World Meteorological Organization (WMO) and adhere to the permitted tolerance limits for each weather parameter.**

***Index Terms*— automatic weather station; long short-term memory; predictive maintenance; sensor error detection**

I. Introduction

Indonesia exhibits a wide array of weather and extreme climate phenomena [1]. Weather information plays a pivotal role across various sectors, serving as the cornerstone for policymaking by the central government, local authorities, and other stakeholders in infrastructure development, transportation, agriculture, tourism, energy, industry, and more. In the year 2021, the National Disaster Management Agency (BNPB) recorded that a striking 99.5% of the disasters occurring in Indonesia were of hydrometeorological nature. The top three prevailing events were floods, extreme weather, and landslides [2].

Automatic Weather Stations (AWS) are automated equipment utilized for observing meteorological parameters, including atmospheric pressure, rainfall, relative humidity, atmospheric temperature, wind speed, and wind direction. The Meteorology, Climatology, and Geophysics Agency (BMKG) currently operates 368 AWS units distributed across Indonesia, both within and outside the vicinity of Meteorological Station Management Units (UPT) (BMKG, 2023). BMKG currently conducts maintenance activities, which encompass both corrective and predictive maintenance [4]. Corrective maintenance involves actions taken when AWS sensors are damaged, necessitating replacement or repair. Preventive maintenance, on the other hand, is a routine maintenance activity performed at scheduled intervals. But maintenance is not limited to corrective and preventive maintenance alone but also includes predictive maintenance [5].

Predictive maintenance can be classified into three primary approaches: knowledge-based, physics-based, and data-based methods [6]. The knowledge-based method leverages the expertise and experience of specialists to diagnose equipment failures. The physics-based method employs mathematical or physical comprehension of the system to assess the remaining useful life of the machinery. The data-based method makes use of historical data collected by sensors on the equipment to make failure predictions. Of the three approaches mentioned, the most suitable predictive maintenance model for AWS sensors is the data-based approach. This is because AWS generates data for each interconnected meteorological parameter, and this data is critical for detecting the sensor's condition.

The study aims to develop a system that can monitor sensor output values and detect AWS sensor errors based on historical sensor data as a pivotal step in predictive maintenance. The sensor parameters used in this study include pressure atmospheric pressure, rainfall, relative humidity, and atmospheric temperature. The machine learning algorithm chosen for this study is Long Short-Term Memory (LSTM). Sensor condition requirements will align with the World Meteorological Organization (WMO) standards concerning measurement tolerance for meteorological parameters, which specify maximum allowable deviations as follows: 5% for rainfall, 0.2°C for atmospheric temperature, 3% for relative humidity, 0.15 hPa for atmospheric pressure [7].

Within the realm of predictive maintenance, the LSTM (Long Short-Term Memory) algorithm outperforms other machine learning algorithms. The LSTM algorithm, classified within the neural network model category, possesses exceptional capabilities in understanding long-term relationships within sequential data. Additionally, it adeptly captures temporal relationships within data, exhibiting a high level of accuracy and facilitating the generation of exceptionally precise predictions regarding the future state of equipment. Oh and Kim obtained results indicating that predictions developed the LSTM model mirror the trends present in actual values. This model is applied to predictive maintenance for real-time equipment status diagnosis. Nevertheless, the LSTM model's accuracy is significantly compromised owing to an insufficiency of training data [8].

Jiang, et.al conclude that the proposed A^2-LSTM method outperforms other existing techniques in the prediction of remaining useful life (RUL). The comparative results illustrate that the A^2-LSTM method can proficiently identify critical attributes and create temporal dependencies within the manufacturing system, offering valuable assistance to maintenance personnel in their duties [9]. Dey and Jana have obtained results indicating that the proposed LSTM model surpasses the KernelRidge regression model in RMSE values, making it a viable choice for effectively conducting predictive maintenance on rotating machinery [10]. Ruhiyat, et.al conclude that the LSTM algorithm can be utilized for predictive maintenance of a ventilator system. The most significant result indicates a 98.4% probability of failure within 50 cycles with an 82% accuracy [11].

II. Basic Concepts

A. Predictive Maintenance

Predictive maintenance involves the utilization of condition monitoring technology to observe the deterioration of components, predict their future status, and consistently revise maintenance plans in accordance with the predictive outcomes [5]. The primary objectives of predictive maintenance are to diagnose the current condition (diagnostic) and forecast future conditions (prognostic). Predictive maintenance can be classified into three approaches: knowledge-based, physics-based, and data-based methods [6], as illustrated in Figure 1.

The knowledge-based method is employed for diagnosing and prognosticating failures, primarily relying on expert knowledge and experience. This approach utilizes historical failure data as a primary tool for prediction. Within the knowledge-based method, three model categories can be identified: rule-based models, case-based models, and fuzzy logic-based models. The physics-based approach leverages the physical understanding of the system to assess the remaining useful life of machinery. This method is divided into several models, including mathematical models, Hidden Markov models, and filtering models such as the Kalman Filter, Extended Kalman Filter, and others.

The data-based approach utilizes data collected from sensors on equipment, components, and machinery to predict failures. This data is extracted to process, analyze, and derive degradation information from it. Choosing the suitable machine learning or deep learning algorithms should align with the pertinent parameters of the equipment. This method provides the benefit of not depending on the precision of mathematical and physical models or intricate expert rule formulation.

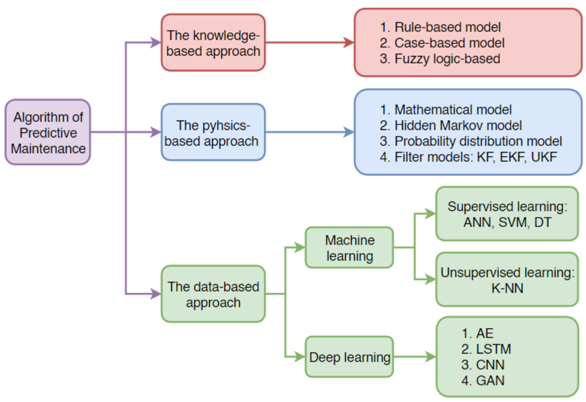


Fig. 1. Algorithm of predictive maintenance

The data-based approach utilizes data collected from sensors on equipment, components, and machinery to predict failures. This data is extracted to process, analyze, and derive degradation information from it. Choosing the suitable machine learning or deep learning algorithms should align with the pertinent parameters of the equipment. This method provides the benefit of not depending on the precision of mathematical and physical models or intricate expert rule formulation.

B. *Automatic Weather Station*

Automatic Weather Station (AWS) is a meteorological station responsible for conducting observations and transmitting data automatically [12]. The primary measurements performed by an AWS include essential weather parameters, such as pressure atmospheric pressure, rainfall, relative humidity, atmospheric temperature, wind speed, and wind direction. In maritime environments, additional parameters such as evaporation, water temperature, and water level are incorporated. Automatic Agroclimate Weather System (AAWS) is implemented in agroclimatology, which encompasses sensors for solar radiation, soil temperature, and soil moisture [13].

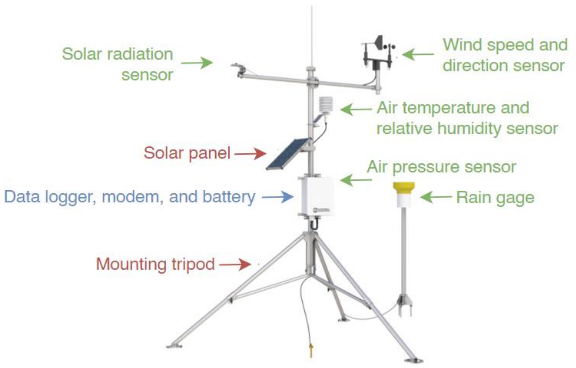


Fig. 2. Components of AWS

AWS is categorized into two groups based on data presentation: AWS real-time and offline. AWS real-time refers to meteorological stations that offer real-time data to users, featuring communication systems and alert mechanisms for extreme weather conditions like storms, heavy rain, high temperatures, and more. On the other hand, AWS offline refers to weather stations that focus on data recording, storing it in storage media, and displaying the current data. The stored data can be downloaded as required at any time. Sensor condition requirements will adhere to the rules set by World Meteorological Organization (WMO). Measurement tolerances that meet the requirements are presented in Table I.

TABLE I. WMO REGULATION

Parameter	Range	Achievable measurement tolerance
Rainfall	0 ~ 500 mm	higher 5% or 0.1 mm
Atmospheric temperature	-80 ~ 60 °C	0.2 °C
Relative humidity	0 ~ 100%	3%
Atmospheric pressure	500 ~ 1080 hPa	0.15 hPa

C. *Long Short-Term Memory*

Long Short-Term Memory (LSTM) is an improvement on the Recurrent Neural Network (RNN). Its main objective is to create models with the ability to retain long-term memory, while also having the capacity to filter out irrelevant information in the training data.

LSTM utilizes a combination of two activation functions: the hyperbolic tangent (tanh) and the sigmoid functions [14]. In the tanh function, the output values are bounded within the (-1,1) range, facilitating the regulation of data flow through the network and preventing the vanishing gradient issue. Additionally, the sigmoid activation function is also incorporated into LSTM. It confines the output values to the (0,1) range, enabling the neural network to filter out unrelated data. When the output value approaches zero, it essentially becomes zero. The tanh and sigmoid activation function are defined as follows:

$$\tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \quad (1)$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2)$$

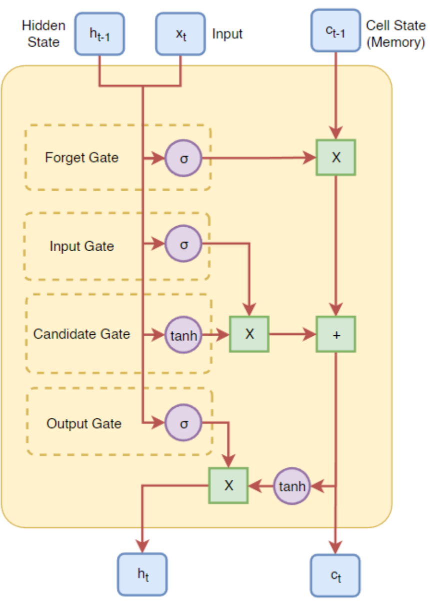


Fig. 3. LSTM architecture

LSTM introduces a cell state, which is a crucial component of the algorithm. The hidden state functions as short-term working memory, while the cell state is employed as long-term memory to retain important information from previous data. LSTM possesses the ability to modify cell state values using a mechanism called Gates. LSTM incorporates four gates, as illustrated in Figure 3:

- The Forget Gate decides which values from the preceding cell state to discard and which ones to preserve.
- The Input Gate picks values from the prior hidden state and the present input for updating by subjecting them to a sigmoid function. The output of this function is subsequently multiplied by the previous cell state.
- The Cell State Candidate Gate initially governs the flow of information within the network by using a tanh function on the prior hidden state and the present input. The resultant of the tanh function is then multiplied by the output of the Input gate to compute the candidate for the current cell state. This candidate is then added to the previous cell state.
- The Output Gate calculates the current hidden state by employing a sigmoid function to decide which new information is crucial to consider. This is accomplished by applying the sigmoid function to the previous hidden state and the current input. The current cell state value is then processed by a tanh function. Finally, the results of these two functions are multiplied together.

III. Methods

The study relies on recorded sensor data for rainfall, relative humidity, atmospheric temperature, and atmospheric pressure. This data was obtained from the AWS Jatiwangi site, which is situated at the Class III Meteorological Station in Kertajati, Majalengka Regency, West Java Province. Data records were obtained from the BMKG Central Office in Jakarta, accessible through the website *https://awscenter.bmkg.go.id/*.

Fig. 4. AWS Jatiwangi site

The study covers a time frame starting on January 1, 2017 and concluding on December 31, 2021 with data recorded at 10-minute intervals. The entire study flowchart is depicted in Figure 5.

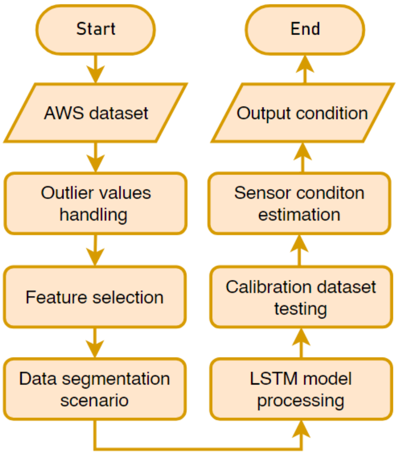


Fig. 5. Study flowchart

A. *Outlier Values Handling*

Handling outlier values aims to ensure that they do not exert an excessive influence on the outcomes and interpretations of the forthcoming LSTM model. The BMKG Central Database sets forth criteria concerning the quality of AWS data. These guidelines, presented in Table II, represent general requirements applied by BMKG to enforce quality control for AWS data throughout Indonesia.

TABLE II. Quality control AWS data

Sensor	Minimum Threshold	Maximum Threshold	Stepcheck Threshold
Rainfall (mm)	0	300	30
Atmospheric temperature (ºC)	5	45	3
Relative humidity (%)	5	100	15
Atmospheric pressure (hPa)	800	1050	2

B. *Feature Selection*

Feature selection is conducted to recognize the most significant and informative parameters among the features available in the dataset. This process involves the elimination of features that do not make a substantial contribution to the model under development. Feature selection encompasses several methods, including filtering methods, wrapper methods, and embedded methods [15]. In this study, feature selection is conducted using the embedded method, specifically the Random Forest Importance Feature technique.

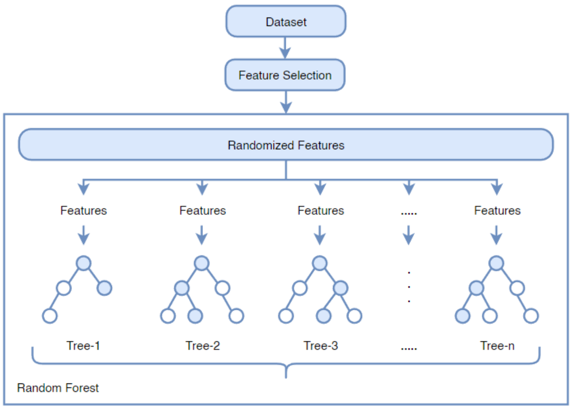


Fig. 6. Random forest architecture

Random Forest is a machine learning algorithm that amalgamates numerous decision trees to generate more precise and robust predictions [16]. Features with a greater impact on the model's predictions will exhibit higher degrees of importance in the feature importance metric. This technique includes permuting feature values to evaluate their influence on the model's performance. Features that have a significant impact on the model will result in a notable decrease in accuracy when their values are shuffled. According to [17], their study results showed that the Random Forest algorithm attained an accuracy exceeding 90% for feature selection within a dataset.

C. Data Segmentation

The data segmentation scenario is split into two conditions: the normal dataset and the synthetic dataset [18]. The normal dataset contains the original parameter values, signifying sensors operating under standard conditions. Meanwhile, the synthetic dataset is created by adjusting parameter values beyond the normal sensor tolerance limits in compliance with the 2021 WMO No. 8 standard, indicating sensors in an erroneous state.

The AWS dataset is segmented into three components: dataset fo the training process, validation process, and testing process. The training dataset is utilized to train the LSTM algorithm and generate training models for each sensor parameter. The validation dataset is employed to assess the performance of the model on data that it has not encountered during the training process. Finally, the testing dataset is prepared to assess the LSTM models that have been built, comparing their results with the calibration dataset from AWS Jatiwangi.

The AWS dataset will be distributed as follows: 70% will be used for the LSTM training process, 25% will be reserved for validation the LSTM model, and 5% will be allocated for testing against the AWS Jatiwangi calibration dataset. Within each process, 50% will be derived from the normal dataset, while 50% will be synthesized. Synthetic data generated from data transformation scenarios will be randomly inserted into the dataset. This random scenario is intended to make the error patterns of sensor readings resemble real-world occurrences during operations.

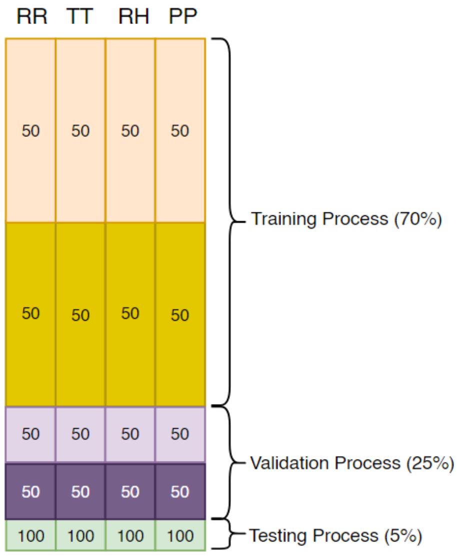


Fig. 7. Segmentation dataset process

D. LSTM Model

The dataset processed in the LSTM algorithm consists of AWS Jatiwangi data from January 2017 to December 2021. The LSTM model is implemented in Python using the PyTorch library. The LSTM design used in this study is presented in Table III.

TABLE III. DESIGN OF LSTM MODEL

Parameterization	**RR**	**TT**	**RH**	**PP**
Unit	mm	ºC	%	hPa
Input Layer (I)	3	3	3	3
Hidden Layer (H)	2	2	2	2
Output Layer (O)	RR	TT	RH	PP
Epoch	100			
Hidden Size	64			
Optimizer	Adam			

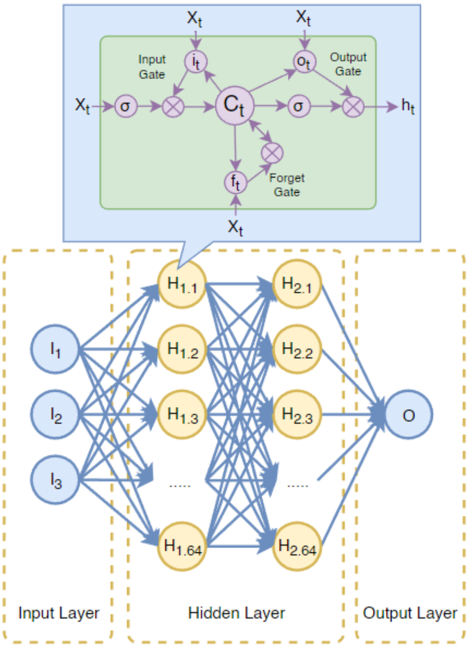


Fig. 8. Architecture LSTM sensor model

The LSTM model design for rain gauge sensors uses three input parameters: rainfall, atmospheric temperature, and relative humidity. The structure comprises a single input layer, two hidden layers, and single output layer. For synthetic data, the scenario includes randomly introducing damage values ranging from +5% to +400% and 0.1 to 100 for 0 mm from the normal values.

The LSTM model design for atmospheric temperature sensors uses three input parameters: atmospheric temperature, relative humidity, and rainfall. The structure comprises a single input layer, two hidden layers, and single output layer. For synthetic data, the scenario involves randomly introducing damage values ranging from ±2% to ±50% from the normal values.

The LSTM model design for relative humidity sensors uses three input parameters: relative humidity value, atmospheric temperature, and rainfall. The structure comprises a single input layer, two hidden layers, and single output layer. For synthetic data, the scenario involves randomly introducing damage values ranging from -9% to -60% and +9% to +70% from the normal values.

The LSTM model design for atmospheric pressure sensors uses two input parameters: atmospheric pressure value and atmospheric temperature. The structure comprises a single input layer, two hidden layers, and single output layer. For synthetic data, the scenario includes randomly introducing damage values ranging from ±0.04% to 10% from the normal values.

LSTM sensor model architecture is depicted in Figure 8.

E. *Performance Evaluation*

The Confusion Matrix is employed as an evaluation tool for assessing the classification model's performance in the LSTM algorithm [19]. This evaluation method measures how accurately the classification model predicts the class or label of the data. The confusion matrix comprises four main cells: *True Positives (TP)* are the data points correctly identified as positive by the model, while *True Negatives (TN)* are the data points correctly identified as negative. *False Positives (FP)*, or Type I Errors, are the data points inaccurately identified as positive, and *False Negatives (FN)*, or Type II Errors, are the data points inaccurately identified as negative.

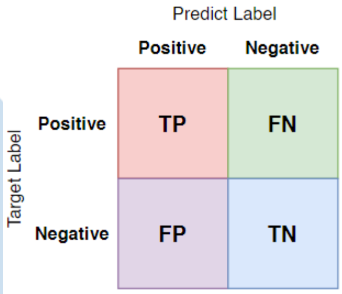


Fig. 9. Confusion matrix

The confusion matrix additionally enables the computation of various performance evaluation metrics, detailed as follows:

- Accuracy: Measures how well correct classifications are made compared to the total predictions. In pattern recognition, accuracy assesses how well a system can correctly identify patterns.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3)$$

- Probability of Detection (POD): Measures how well the model can detect actual positive data. POD is a vital component of the confusion matrix as it specifically highlights the system's success in detecting actual occurrences.

$$POD = \frac{TP}{TP + FN} \quad (4)$$

- False Alarm Rate (FAR): Measures how often the model generates false positive signals, indicating instances where the system incorrectly identifies negatives as positives. FAR provides insights into the system's reliability in identifying negative events, serving as a crucial measure of its trustworthiness.

ISSN 2355-3286

$$FAR = \frac{FP}{FP + TN} \tag{5}$$

IV. Result and Discussion

The dataset comprises 5 columns: the "tanggal" column indicates the date and time, the "rr" column represents the rainfall sensor values in mm, the "tt_air_avg" column denotes the atmospheric temperature sensor values in oC, the "rh_avg" column signifies the relative humidity sensor values in %, and the "pp_air" column displays the atmospheric pressure sensor values in hPa.

	tanggal	rr	tt_air_avg	rh_avg	pp_air
0	2017-01-01 00:00:00+00	0.0	20.4	100.0	1003.7
1	2017-01-01 00:10:00+00	0.0	20.8	100.0	1004.3
2	2017-01-01 00:20:00+00	0.0	21.6	99.9	1005.1
3	2017-01-01 00:30:00+00	0.0	23.3	96.0	1006.2
4	2017-01-01 00:40:00+00	0.0	25.0	90.5	1006.4
...	...	...	...	...	...
186737	2021-12-31 23:10:00+00	0.5	25.0	96.6	1005.0
186738	2021-12-31 23:20:00+00	0.5	25.0	96.4	1005.1
186739	2021-12-31 23:30:00+00	0.5	25.2	95.8	1005.1
186740	2021-12-31 23:40:00+00	0.5	25.4	95.0	1005.1
186741	2021-12-31 23:50:00+00	0.5	25.6	94.1	1005.3

Fig. 10. AWS dataset

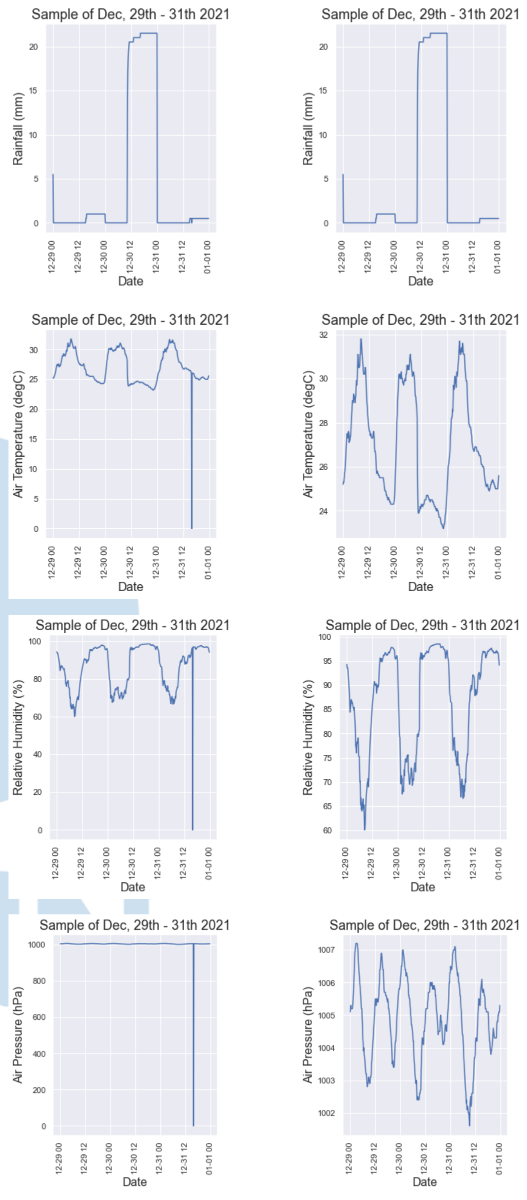
A. *Handling outlier values*

According to the BMKG Central Database requirements, outliers are removed by excluding them from the dataset. TABLE IV displays the dataset attributes after going through data preprocessing. Each column shows a reduction in data count due to the removal of outliers. The minimum and maximum values for each sensor attribute have been adjusted to align with BMKG's AWS data quality control standards.

TABLE IV. After handling outlier values

Attribute	RR	TT	RH	PP
Total data	175,787	175,787	175,787	175,787
Minimum	0.0	17.5	15.9	994.5
Maximum	276.0	38.6	100.0	1011.1
Mean	4.2	27.5	75.7	1004.6
Std	15.2	3.4	18.2	1.9

Figure 11 displays a sample dataset from December 29th to December 31st, 2021, where all parameters have values of 0. A sensor recording a value of 0 is interpreted as an indication of a malfunction in other components of the AWS, such as power supply issues or data transmission failures to the Central Database. This inference is drawn from the consistent occurrence of 0 values for each parameter within the same minute. It is also predicated on the absence of direct checks on the datalogger at the AWS Jatiwangi site, as the dataset was solely obtained from the AWS Center BMKG website. Consequently, the 0 values can be disregarded, as the failures are attributed to other components, rendering the sensors inactive or switched off.



(a) (b)

Fig. 11. Imputation testing plot with 60 minutes of missing data

B. *Feature Selection*

The feature importance for the relative humidity sensor is 0.43, for the atmospheric pressure sensor is 0.29, and for the atmospheric temperature sensor is 0.27 in relation to the rainfall sensor. Regarding the atmospheric temperature sensor, the relative humidity sensor holds a feature importance of 0.86, while the atmospheric pressure sensor has 0.12, and the rainfall sensor has 0.02. As for the relative humidity sensor, the atmospheric temperature sensor holds a feature importance of 0.82, the rainfall sensor has 0.12, and the atmospheric pressure sensor has 0.05. In the context of the atmospheric pressure sensor, the relative humidity sensor has a feature importance of 0.48, the atmospheric temperature sensor has 0.39, and the rainfall sensor has 0.12.

The LSTM model employs the two most prominent feature importance values as inputs from the other sensors. Therefore, it was found that the RR model utilizes inputs from RR, RH, and PP, the TT model uses inputs from TT, RH, and PP, the RH model employs inputs from RH, TT, and RR, and the PP model takes inputs from PP, RH, and TT.

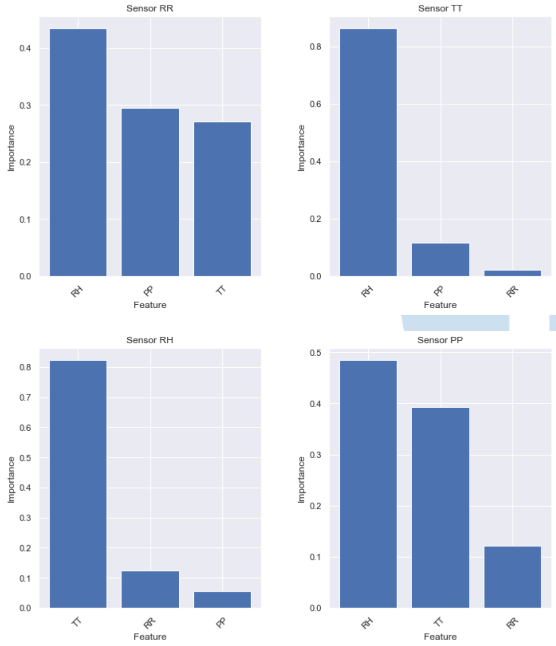


Fig. 12. Sensor feature selection

C. Data Segmentation

The dataset composition used for the LSTM model training process is 70% of the data, which amounts to 123,051 data points. The dataset used for the LSTM model validation process is 25%, which equals 43,947 data points. Additionally, the dataset used for the testing process with the AWS Jatiwangi calibration dataset is 5%, totaling 8,789 data points. Visualization of randomize synthetic data shown in Figure 13. Label 1 indicates synthetic values within the dataset, while label 0 represents normal values within the dataset.

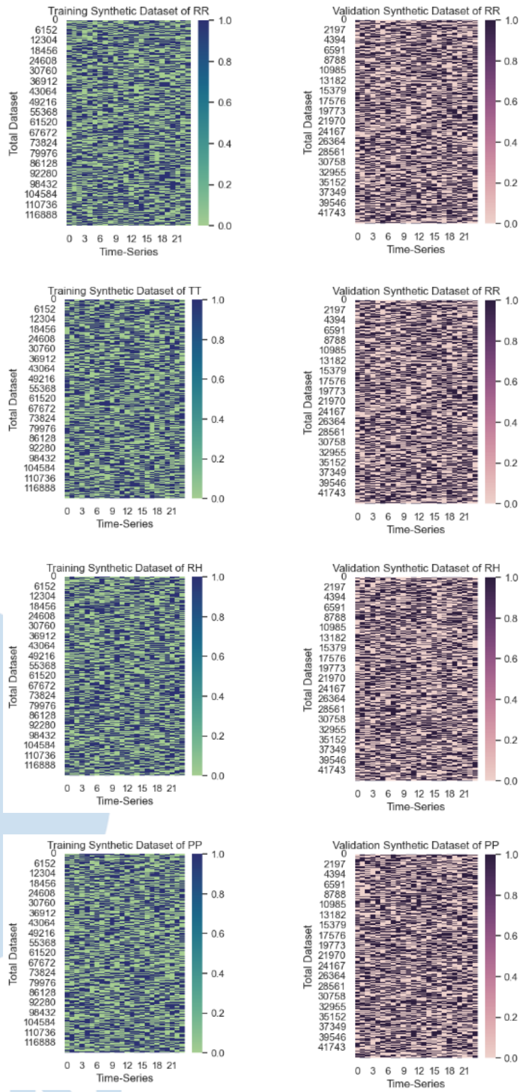


Fig. 13. Imputation testing plot with 60 minutes of missing data

In handling time series data, converting the data into matrices for Python processing using vectorization can greatly improve the speed and efficiency of the process compared to employing *for* or *while* loops that operate on individual elements separately. Vectorization, utilizing NumPy's broadcasting operations, enables mass array operations, allowing for parallel processing and optimization by the library. This leads to a significant performance enhancement, especially when dealing with extensive datasets.

D. Evaluation of LSTM Model

Training is conducted to enable the model to discern patterns within both the input and output data from the training dataset. In this process, four LSTM architecture models are constructed, namely the RR model, TT model, RH model, and PP model. Each of these models is designed to handle specific aspects or parameters of the data. The information provided describes the architecture and training progress for four different LSTM models, each with 2 hidden layers consisting of 64 neurons:

- Model RR for estimating the condition of rainfall sensors: Loss at epoch 1: 0.2384 and Loss at epoch 100: 0.0269.
- Model TT for estimating the condition of atmospheric temperature sensors: Loss at epoch 1: 0.2904 and Loss at epoch 100: 0.0322.
- Model RH for estimating the condition of relative humidity sensors: Loss at epoch 1: 0.3636 and Loss at epoch 100: 0.0981.
- Model PP for estimating the condition of atmospheric pressure sensors: Loss at epoch 1: 0.1766 and Loss at epoch 100: 0.0098.

These models are used to estimate the condition of sensors for different environmental parameters based on the given training data. The loss values at different epochs indicate how well the models are performing during the training process, with lower loss values typically indicating better model performance. Among these four LSTM models, the PP model has the lowest loss, suggesting that it is the most predictable pattern when there are reading errors by the PP sensor. In other words, the PP model exhibits a higher degree of accuracy in estimating the condition of the atmospheric pressure sensor when compared to the other sensor models.

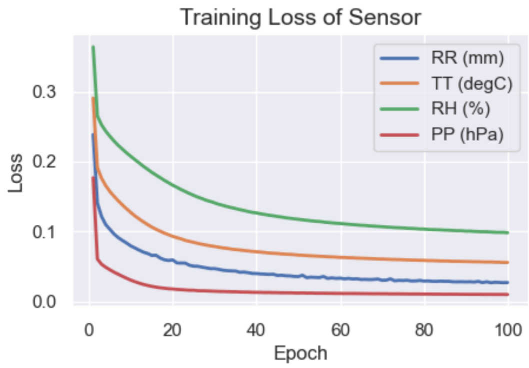


Fig. 14. Sensor loss

During the validation of the sensor dataset, the LSTM model demonstrates an accuracy above 85%, with POD exceeding 0.85 and FAR below 0.15. These metrics signify the model's proficiency in accurately recognizing the status of AWS sensors. Enclosed is the distribution table illustrating evaluation metrics during the 4-hour time-series validation in Table V.

TABLE V. PERFORMANCE EVALUATION OF SENSORS IN VALIDATION PROCESS

Time-Series	Accuracy (%)				POD				FAR			
	RR	TT	RH	PP	RR	TT	RH	PP	RR	TT	RH	PP
t+0	85.1	86.4	77.1	97.3	0.812	0.963	0.811	0.996	0.099	0.200	0.260	0.047
t+1	91.2	89.4	82.3	97.9	0.906	0.927	0.835	0.992	0.082	0.133	0.189	0.033
t+2	93.3	91.0	86.3	98.2	0.942	0.927	0.878	0.990	0.076	0.106	0.151	0.025
t+3	94.0	92.0	87.5	98.3	0.944	0.930	0.887	0.988	0.064	0.090	0.136	0.021
t+4	94.5	92.6	88.1	98.5	0.948	0.932	0.887	0.989	0.058	0.080	0.126	0.019
t+5	94.8	92.9	88.6	98.6	0.950	0.934	0.890	0.991	0.054	0.077	0.119	0.019
t+6	95.0	93.3	88.7	98.8	0.951	0.939	0.890	0.993	0.050	0.073	0.117	0.018
t+7	95.2	93.5	88.8	98.8	0.954	0.942	0.891	0.993	0.049	0.071	0.114	0.017
t+8	95.3	93.7	89.1	98.8	0.954	0.943	0.893	0.994	0.048	0.069	0.111	0.017
t+9	95.4	93.8	89.2	98.9	0.956	0.944	0.894	0.995	0.047	0.069	0.110	0.017
t+10	95.5	93.8	89.3	98.9	0.957	0.945	0.895	0.995	0.046	0.068	0.110	0.017
t+11	95.6	93.9	89.2	98.9	0.959	0.946	0.895	0.995	0.047	0.068	0.111	0.017
t+12	95.7	93.9	89.3	98.9	0.960	0.946	0.896	0.995	0.046	0.068	0.109	0.017
t+13	95.7	93.9	89.4	98.9	0.960	0.946	0.896	0.994	0.046	0.068	0.109	0.017
t+14	95.7	93.9	89.5	98.9	0.960	0.947	0.897	0.995	0.046	0.068	0.108	0.017
t+15	95.7	93.9	89.4	98.9	0.960	0.946	0.896	0.995	0.046	0.069	0.108	0.017
t+16	95.7	93.9	89.5	98.9	0.960	0.947	0.897	0.995	0.046	0.068	0.107	0.017
t+17	95.8	93.9	89.4	98.9	0.961	0.946	0.897	0.995	0.045	0.068	0.108	0.017
t+18	95.8	93.9	89.4	98.9	0.961	0.946	0.896	0.995	0.045	0.067	0.108	0.017
t+19	95.8	94.0	89.4	98.9	0.962	0.947	0.897	0.995	0.045	0.067	0.109	0.017
t+20	95.8	94.0	89.5	98.9	0.962	0.947	0.898	0.995	0.045	0.067	0.108	0.017
t+21	95.8	94.0	89.5	98.9	0.962	0.947	0.898	0.995	0.045	0.067	0.109	0.017
t+22	95.8	94.0	89.4	98.9	0.962	0.948	0.898	0.995	0.045	0.067	0.109	0.017
t+23	95.8	94.0	89.5	98.9	0.962	0.947	0.897	0.995	0.045	0.068	0.108	0.017

TABLE VI. CALIBRATION DATASET TESTING

Data point-	TT		RH		PP		Status	
	STD	UUT	STD	UUT	STD	UUT	STD	UUT
1	28,41	28,19	80,94	79,00	1007,68	1007,92	normal	normal
2	28,46	28,09	80,33	81,00	1007,67	1007,92	normal	normal
3	28,48	28,04	80,86	80,30	1007,65	1007,92	normal	normal
4	28,50	28,04	78,42	80,80	1007,63	1007,89	normal	normal
5	28,55	28,07	79,87	78,84	1007,62	1007,88	normal	TT Error
6	28,56	28,16	79,05	79,91	1007,60	1007,86	normal	normal
7	28,61	28,17	80,81	80,10	1007,58	1007,83	normal	normal
8	28,66	28,15	80,29	81,60	1007,54	1007,81	normal	normal
9	28,64	28,11	78,17	80,10	1007,53	1007,79	normal	normal
10	28,67	28,04	78,85	79,36	1007,50	1007,77	normal	normal
11	28,72	28,06	76,82	79,77	1007,48	1007,74	normal	normal
12	28,79	28,09	77,30	78,75	1007,47	1007,73	normal	normal
13	28,91	28,14	77,19	78,26	1007,45	1007,72	normal	normal
14	29,05	28,26	79,75	79,39	1007,44	1007,70	normal	normal
15	29,12	28,42	76,16	79,98	1007,42	1007,69	normal	normal
16	29,23	28,53	76,14	77,90	1007,39	1007,66	normal	normal
17	29,34	28,68	76,54	77,07	1007,38	1007,64	normal	normal
18	29,48	28,79	76,49	77,58	1007,38	1007,64	normal	normal
19	29,58	28,93	75,64	77,22	1007,38	1007,64	normal	normal
20	29,55	29,01	74,01	75,73	1007,39	1007,64	normal	normal
21	29,46	28,99	73,74	73,01	1007,38	1007,64	normal	normal
22	29,47	29,01	72,13	74,49	1007,39	1007,65	normal	normal
23	29,46	29,00	71,68	72,94	1007,39	1007,65	normal	normal
24	29,45	29,03	72,40	71,80	1007,41	1007,67	normal	normal
25	29,47	29,08	73,24	74,14	1007,42	1007,68	normal	normal
26	29,53	29,10	74,55	73,34	1007,44	1007,70	normal	normal

The increase in accuracy, such as the lower accuracy at t+0 compared to t+1, stems from several factors:

- Short-term detections (t+1) are simpler due to the availability of information from t+0, which can be used for t+1 detections. This added information can enhance accuracy by providing the model access to more current data.
- The known data at t+0 enables clearer pattern recognition, aiding the model in identifying patterns for t+1 detections.
- The potential lack of relevant features or necessary data for t+0 detections might lead to reduced accuracy in precisely detection at t+0.

Testing was conducted on the calibration dataset from AWS Jatiwangi. This dataset was acquired from the calibration activities performed by BBMKG Wilayah II Tangerang Selatan on August 22, 2022. The calibration dataset comprises parameters TT, RH, and PP with normal values labeled as 0. During the testing using the calibration dataset, the models tested were restricted to TT and PP sensors as input data for sensor RR was not available. Hence, the RH and RR models could not be tested due to the absence of RR sensor data as input for the model. This limitation in testing the calibration dataset stems from the lack of RR sensor input, as the calibration method for RR sensors by BBMKG Wilayah 2 Tangerang Selatan differs from the calibration method used for TT, RH, and PP sensors. Result of testing is shown in Table VI.

The STD column contains values from the standard calibration sensor for each parameter, while the UUT column displays values from the AWS Jatiwangi sensor. The testing results reveal the model's proficiency in identifying patterns within the standard calibration sensor values, confirming the normal state of this sensor. However, within the AWS Jatiwangi sensor, the model detected an abnormal pattern in a single data point, particularly in the 5th entry, where the temperature (TT) measured 28.05°C.

Although an anomaly was detected in one data point, overall assessments still categorize the TT and PP sensors from AWS Jatiwangi as within the normal range. The model's analysis indicates that this anomaly was only observed in one data point, resulting in the general conclusion that both sensors remain considered stable and normal.

V. CONCLUSION

The paper presents a technique for detecting sensor errors in AWS using LSTM algorithms. The method produces four sensor models: RR, TT, RH, and PP models. These models predict by identifying patterns of reading errors within synthetic data scenarios in the training dataset. There's a notable decrease in loss values as the number of epochs increases. Individual sensor performance evaluations show that the models can detect sensor reading errors with high average accuracy (>90%, except for RH), high POD, and low FAR. The proposed sensor error detection can serve as a prospective method for predictive maintenance, offering potential implementation for future AWS maintenance procedures. The predictive maintenance framework comprises critical stages related to sensor anomalies in both primary and secondary processes. In this study, the commencement of future work on developing algorithms for estimating remaining useful life should be grounded in the detection of errors.

ACKNOWLEDGMENT

The authors extend their gratitude to the Indonesia Lembaga Pengelola Dana Pendidikan (LPDP) Grant 202109110107819 for offering partial funding support for this research and also express their appreciation to Badan Meteorologi, Klimatologi, dan Geofisika for granting access to the AWS Jatiwangi dataset and calibration dataset, which significantly contributed to this study.

REFERENCES

[1] BMKG, "Peraturan Badan Meteorologi, Klimatologi, Dan Geofisika Republik Indonesia Nomor 4 Tahun 2020 Tentang Rencana Strategis Badan Meteorologi, Klimatologi, Dan Geofisika Tahun 2020-2024," 2020.

[2] BNPB, "BNPB Verifikasi 5.402 Kejadian Bencana Sepanjang Tahun 2021," https://bnpb.go.id/berita/bnpb-verifikasi-5-402-kejadian-bencana-sepanjang-tahun-2021 Accessed: Feb. 13, 2023. [Online]. Available: https://bnpb.go.id/berita/bnpb-verifikasi-5-402-kejadian-bencana-sepanjang-tahun-2021

[3] BMKG, "AWS Center BMKG," https://awscenter.bmkg.go.id/. Accessed: Feb. 20, 2023. [Online]. Available: https://awscenter.bmkg.go.id/

[4] BMKG, "Peraturan Kepala Badan Meteorologi, Klimatologi, dan Geofisika Nomor 7 Tahun 2014 tentang Standar Teknis dan Operasional Pemeliharaan Peralatan Pengamatan Meteorologi, Klimatologi, dan Geofisika," 2014.

[5] J. Zhao, C. Gao, and T. Tang, "A Review of Sustainable Maintenance Strategies for Single Component and Multicomponent Equipment," Sustainability (Switzerland), vol. 14, no. 5. MDPI, Mar. 01, 2022. doi: 10.3390/su14052992.

[6] N. Es-sakali, M. Cherkaoui, M. O. Mghazli, and Z. Naimi, "Review of predictive maintenance algorithms applied to HVAC systems," Energy Reports, vol. 8, pp. 1003–1012, Nov. 2022, doi: 10.1016/j.egyr.2022.07.130.

[7] WMO, "Guide to Instruments and Methods of Observation Volume I-Measurement of Meteorological Variables," 2021.

[8] S. H. Oh and J. G. Kim, "LSTM-based PdM Platform for Automobile SCU Inspection Equipment," in International Conference on Information Networking, IEEE Computer Society, 2023, pp. 363–365. doi: 10.1109/ICOIN56518.2023.10048924.

[9] Y. Jiang, P. Dai, P. Fang, R. Y. Zhong, X. Zhao, and X. Cao, "A2-LSTM for predictive maintenance of industrial equipment based on machine learning," Comput Ind Eng, vol. 172, Oct. 2022, doi: 10.1016/j.cie.2022.108560.

[10] D. Dey and R. Jana, "Bearing Fault Predictive Maintenance using LSTM," in ICAN 2022 - 3rd International Conference on Computing, Analytics and Networks - Proceedings, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ICAN56228.2022.10007139.

[11] Y. H. Ruhiyat, S. Sumaryo, and E. Susanto, "Predictive Maintenance for a Ventilator Using LSTM Algorithm," in APWiMob 2022 - Proceedings: 2022 IEEE Asia Pacific Conference on Wireless and Mobile, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/APWiMob56856.2022.10014252.

[12] WMO, "Guide to Instruments and Methods of Observation Volume III-Observing Systems," 2021.

[13] B. Santoso, M. Ryan, F. Mukhlish, and D. Kurniadi, "Generating Synthetic Values on Automatic Weather Station Dataset using GAIN," in 2023 8th International Conference on Instrumentation, Control, and Automation (ICA), IEEE, Aug. 2023, pp. 218–223. doi: 10.1109/ICA58538.2023.10273082.

[14] K. Zarzycki and M. Ławryńczuk, "LSTM and GRU neural networks as models of dynamical processes used in predictive control: A comparison of models developed for two chemical reactors," Sensors, vol. 21, no. 16, Aug. 2021, doi: 10.3390/s21165625.

[15] N. Pudjihartono, T. Fadason, A. W. Kempa-Liehr, and J. M. O'Sullivan, "A Review of Feature Selection Methods for Machine Learning-Based Disease Risk Prediction," Frontiers in Bioinformatics, vol. 2, Jun. 2022, doi: 10.3389/fbinf.2022.927312.

[16] M. I. Prasetiyowati, N. U. Maulidevi, and K. Surendro, "Feature selection to increase the random forest method performance on high dimensional data," International Journal of Advances in Intelligent Informatics, vol. 6, no. 3, pp. 303–312, 2020, doi: 10.26555/ijain.v6i3.471.

[17] R. C. Chen, C. Dewi, S. W. Huang, and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," J Big Data, vol. 7, no. 1, Dec. 2020, doi: 10.1186/s40537-020-00327-4.

[18] P. Wellyantama and S. Soekirno, "Temperature, pressure, relative humidity and rainfall sensors early error detection system for automatic weather station (AWS) with artificial neural network (ANN) backpropagation," in The 10th International Conference on Theoretical and Applied Physics, IOP Publishing Ltd, Mar. 2020. doi: 10.1088/1742-6596/1816/1/012056.

[19] M. Kubat, An Introduction to Machine Learning. Cham: Springer International Publishing, 2017. https://doi.org/10.1007/978-3-319-63913-0