International Journal Software Engineering and Computer Science (IJSECS)
6 (2), 2026, 627-635
Published Online August 2026 in IJSECS (http://www.journal.lembagakita.com/index.php/ijsecs)
P-ISSN: 2776-4869, E-ISSN: 2776-3242. DOI: https://doi.org/10.35870/ijsecs.v6i2.7593.

RESEARCH ARTICLE Open Access

Analysis of Early Detection of Automatic Weather Station Sensor Failures: A Comparative Study of Supervised Deep Neural Networks and Unsupervised Autoencoders

Hasbullah Zuhri Hasibuan [1*], **Sudarno Wiharjo** [2], **Yan Mitha Djaksana** [3]

[1*,2,3] Master of Informatics Engineering Program, Universitas Pamulang, South Tangerang City, Banten Province, Indonesia.

**Corresponding author*: zuhri05@gmail.com.

Received: April XX, 2026; Accepted: April XX, 2026; Published: August 1, 2026.

Abstract: Automatic Weather Stations (AWS) continuously collect and record meteorological data in real time and support weather information services. The reliability of AWS observations depends on sensor performance, as sensor failures, equipment degradation, and communication disturbances may produce anomalous data and reduce data quality. This study aims to develop an early-detection approach for AWS sensor anomalies using a Deep Neural Network (DNN) and an Autoencoder, compare their performance, and identify the sensors most frequently associated with anomalies. The dataset consists of 385,237 observations collected from the AWS Ancol station in North Jakarta from January to September 2025, covering nine meteorological and oceanographic parameters. The study involved data preprocessing, Min-Max normalization, model training, and evaluation using accuracy, precision, recall, F1-score, confusion matrix, and ROC-AUC. The DNN achieved 99.6% accuracy, 0.922 precision, 0.996 recall, 0.958 F1-score, and 1.000 AUC. The Autoencoder achieved 95.8% accuracy, 0.578 precision, 0.023 recall, 0.043 F1-score, and 0.584 AUC. The AUC of 1.000 obtained by the DNN should be interpreted within the characteristics of the labeled dataset used in this study and should not be considered evidence of perfect generalization. Sensor analysis identified wind direction as the parameter most frequently associated with anomalies, contributing 46.67% of the detected anomalies in the DNN and 66.02% in the Autoencoder. These findings indicate that the supervised DNN performed better than the Autoencoder for anomaly detection on the labeled AWS dataset used in this study.

Keywords: Automatic Weather Station; Deep Neural Network; Autoencoder; Anomaly Detection; Predictive Maintenance.

1. Introduction

An Automatic Weather Station (AWS) is an automated observation system that continuously measures and records various meteorological parameters in real time. AWS supports weather information services, early warning systems, and decision-making in sectors such as transportation, maritime activities, agriculture, and disaster mitigation (Ariffudin & Musa, 2022; WMO, 2021). Its main advantage is the ability to collect high-frequency observations with limited need for continuous operator intervention. The continuous observations generated by AWS provide an important source of data for monitoring weather and environmental conditions.

However, the quality of the information generated by AWS depends heavily on the condition of the sensors used. Sensor failure, device degradation due to age, environmental disturbances, or data communication errors can lead to abnormal data or anomalies (WMO, 2021; Akbar *et al.*, 2024). If these conditions are not detected promptly, inaccurate observations may be included in the analysis and processing of meteorological information. The reliability of meteorological observations therefore depends not only on continuous data collection but also on the ability to identify measurement problems and maintain observation quality (WMO, 2021). The identification of AWS sensor problems may involve on-site inspection and evaluation of historical observations by operators. Such processes can require considerable time and human resources, particularly when monitoring systems generate large volumes of observations at short intervals. Manual evaluation may also make it difficult to identify abnormal patterns consistently across multiple sensor parameters. An automated approach for detecting anomalous observations can therefore assist in identifying potential sensor problems at an earlier stage and reduce the risk of prolonged degradation in data quality.

Recent developments in machine learning and deep learning provide opportunities for automated analysis of environmental sensor data. Deep learning models can learn complex nonlinear relationships from high-dimensional data and have been applied to classification, prediction, and time-series analysis (Goodfellow, 2016; LeCun *et al.*, 2015). Studies involving AWS and meteorological sensor data have also applied machine learning and deep learning techniques to address problems such as missing-data treatment and sensor-related data processing (Akbar *et al.*, 2024; Pahlepi *et al.*, 2023; Wicaksana *et al.*, 2023). These studies indicate that data-driven methods can be applied to AWS observations with different characteristics and objectives. Deep learning has also been applied to anomaly detection in multivariate sensor and weather-related data. Malhotra *et al.* (2016) investigated an LSTM-based encoder-decoder approach for multivariate sensor anomaly detection, while Sakurada and Yairi (2014) examined Autoencoders for anomaly detection using nonlinear dimensionality reduction. Kaya *et al.* (2023) investigated anomaly detection in weather-related data at IoT edges, while Duraj *et al.* (2025) examined an LSTM-CNN model for anomaly detection in data streams. Yoon *et al.* (2022) also investigated adaptive model pooling for online deep anomaly detection in complex and evolving data streams. These studies demonstrate the application of deep learning to anomaly detection across different sensor and data-stream environments. A further consideration is the distinction between supervised and unsupervised anomaly detection. A supervised DNN can classify normal and anomalous observations when labeled data are available, whereas an Autoencoder can learn patterns in the input data and identify anomalies based on reconstruction error (Goodfellow, 2016; Malhotra *et al.*, 2016; Sakurada & Yairi, 2014). In this study, the two approaches are compared using the same multivariate AWS dataset to examine their performance under the same data conditions. This comparison also considers the different data requirements and anomaly-detection mechanisms of the two approaches.

Based on these considerations, this study proposes an approach for early detection of AWS sensor anomalies using Deep Neural Networks (DNNs) and Autoencoders. The DNN is used as a supervised learning method to classify normal and anomalous observations based on labeled data, while the Autoencoder is used as an unsupervised approach that identifies anomalies based on reconstruction error (Goodfellow, 2016; Malhotra *et al.*, 2016; Sakurada & Yairi, 2014). The study uses data from the AWS Ancol station in North Jakarta collected from January to September 2025, comprising 385,237 observation records and nine meteorological and oceanographic parameters. The main contributions of this study are: (1) developing an anomaly detection approach for AWS sensor observations using supervised and unsupervised deep learning methods; (2) comparatively evaluating the performance of DNN and Autoencoder models on a multivariate AWS dataset; and (3) analyzing the ranking of sensor parameters associated with anomalies to provide information that can support sensor maintenance decisions. The results are expected to contribute to improving meteorological data quality and provide additional information for more effective AWS sensor maintenance.

2. Related Work

Previous research has shown the increasing use of machine learning and deep learning for anomaly detection in complex and time-series data. Deep learning methods are capable of learning nonlinear representations from high-dimensional data and have been applied to a wide range of classification, prediction, and anomaly detection problems (Goodfellow, 2016; LeCun *et al.*, 2015). These capabilities make deep learning suitable for sensor data in which normal and anomalous patterns may involve relationships among multiple variables. In environmental and meteorological monitoring, several studies have investigated deep learning-based approaches for detecting anomalies and processing sensor data. Kaya *et al.* (2023) developed an anomaly detection approach for weather forecasting at IoT edges, demonstrating the application of intelligent anomaly detection to weather-related data. For evolving data streams, Yoon *et al.* (2022) proposed an adaptive deep anomaly detection approach that considers changes in data distribution over time. Duraj *et* *al.* (2025) investigated an LSTM-CNN model for anomaly detection in data streams and demonstrated the use of a hybrid deep learning architecture for multivariate anomaly detection. These studies indicate that deep learning can be applied to complex sensor and data-stream environments, although their datasets and application settings differ from the AWS dataset examined in this study.

Unsupervised approaches, particularly Autoencoders, have also been applied to anomaly detection. Sakurada and Yairi (2014) investigated Autoencoders for anomaly detection using nonlinear dimensionality reduction, showing that reconstruction-based approaches can be used when labeled anomaly data are not available. Malhotra *et al.* (2016) proposed an LSTM-based encoder-decoder approach for multivariate sensor anomaly detection, in which temporal patterns are learned from sensor observations and deviations from the learned patterns are used to identify anomalies. These studies provide a basis for applying Autoencoder-based methods to multivariate and time-series sensor data, particularly when labeled anomaly observations are limited. Research on AWS has also addressed the processing and quality of meteorological sensor data. Akbar *et al.* (2024) investigated multivariate imputation for solar radiation data from an Automatic Weather Station, while Pahlepi *et al.* (2023) examined LSTM-based imputation of solar radiation measurements. Wicaksana *et al.* (2023) investigated the treatment of missing wind-speed measurements in AWS using Multivariate Imputation by Chained Equations. These studies demonstrate the application of data-driven methods to AWS sensor observations, particularly for addressing missing or incomplete measurements. The quality and reliability of meteorological observations are also addressed in the observation guidelines provided by the World Meteorological Organization (WMO, 2021).

The existing studies show that deep learning has been applied to anomaly detection in sensor data and that machine learning methods have been used to process AWS observations. However, the studies differ in their datasets, objectives, and learning approaches. In particular, supervised classification and unsupervised reconstruction-based detection require different types of training information and use different mechanisms to distinguish normal from anomalous observations. This study therefore compares a supervised Deep Neural Network (DNN) and an unsupervised Autoencoder using the same multivariate AWS dataset. The study also analyzes the sensor parameters associated with detected anomalies to determine which parameters contribute most frequently to anomalous observations. The comparison provides a basis for evaluating the suitability of the two approaches for AWS anomaly detection and for providing information that can support sensor maintenance decisions.

3. Methodology

This study employs a quantitative experimental approach to detect anomalies in AWS sensor data using deep learning methods. The primary objective is to develop and compare the performance of a Deep Neural Network (DNN) and an Autoencoder in detecting anomalous AWS observations. The research workflow consists of literature review, data preparation, anomaly labeling, preprocessing, model development and training, model evaluation, and sensor anomaly analysis.

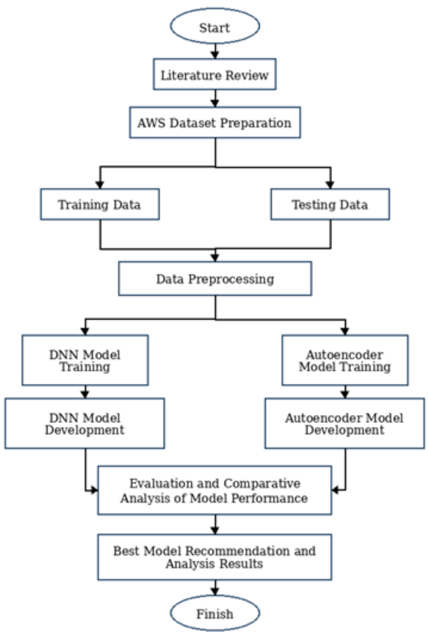

Figure 1. Research Design Flowchart

The research process began with a literature review and identification of anomaly detection problems in AWS sensor observations. The subsequent stages included historical data collection and validation, data preprocessing, anomaly labeling, development and training of DNN and Autoencoder models, evaluation and comparison of model performance, and analysis of the sensor parameters associated with detected anomalies. The dataset contained 385,237 observations, consisting of 3,287 anomalous records and 381,950 normal records. The resulting model comparison and sensor analysis were used to determine the more suitable approach for AWS anomaly detection and to provide information that can support sensor maintenance decisions.

3.1 Automatic Weather Station

An Automatic Weather Station (AWS) is an automated observation system used to measure and record meteorological parameters continuously. Typical AWS parameters include air temperature, humidity, air pressure, wind speed and direction, rainfall, and solar radiation (WMO, 2021). AWS provides observations at high temporal resolution and can support weather monitoring and early warning activities. However, AWS observations may be affected by sensor malfunction, equipment degradation, environmental disturbances, and communication problems. Therefore, anomaly detection is required to identify abnormal sensor observations and support the maintenance of observation quality (WMO, 2021).

3.2 Deep Neural Network (DNN)

A Deep Neural Network (DNN) is a deep learning architecture consisting of an input layer, one or more hidden layers, and an output layer. Each neuron performs a weighted computation followed by an activation function to produce an output (Goodfellow, 2016; LeCun *et al.*, 2015). DNNs can learn nonlinear relationships from input features and have been applied to classification and other data analysis tasks (Goodfellow, 2016). In this study, the DNN is implemented as a supervised classification model that learns the relationship between AWS sensor observations and their corresponding normal or anomalous labels.

3.3 Autoencoder

An Autoencoder is a neural network architecture commonly used for unsupervised representation learning and anomaly detection. It consists of an encoder that transforms the input into a latent representation and a decoder that reconstructs the input from that representation. The model is trained by minimizing the reconstruction error between the original and reconstructed data (Sakurada & Yairi, 2014). For anomaly detection, observations that produce substantially higher reconstruction errors than normal observations can be classified as anomalous. Malhotra *et al.* (2016) demonstrated the use of an LSTM-based encoder-decoder approach for detecting anomalies in multivariate sensor time-series data. In this study, the Autoencoder is used as an unsupervised model to identify anomalous AWS observations based on reconstruction error.

3.4 Research Dataset

This study uses data from the Automatic Weather Station (AWS) at Ancol, North Jakarta. The observations were collected at one-minute intervals from January to September 2025. The dataset consists of 385,237 observation records covering nine meteorological and oceanographic parameters: wind speed, wind direction, air temperature, relative humidity, air pressure, rainfall, solar radiation, sea water temperature, and water level.

3.5 Data Preprocessing

The preprocessing stage was performed to prepare the AWS observations for model development. Records containing missing values were removed from the dataset. The remaining data were normalized using Min-Max Scaling so that the input parameters were represented on a common scale. The dataset was then divided into training and testing sets using an 80:20 ratio. Anomaly labeling was performed using a rule-based approach that combined sensor operational limits, historical AWS data evaluation, maintenance documentation, field inspection results, and verification by BMKG meteorological observers. The operational rules included conditions such as wind speed exceeding the specified physical limits, air temperature outside the defined regional range of 15–40°C, relative humidity exceeding 100%, and sudden unrealistic changes within a one-minute observation interval. These rules were combined with historical evaluation and observer verification to determine the final anomaly labels. The resulting dataset contained 3,287 anomalous records and 381,950 normal records, indicating a substantial class imbalance. This imbalance was considered in the evaluation of the classification models by using precision, recall, F1-score, and AUC in addition to accuracy.

3.6 Evaluation and Validation Scheme

The DNN was implemented as a supervised learning model for classifying AWS observations into normal and anomalous classes. The architecture consists of one input layer, two hidden layers using ReLU activation, and one output layer using sigmoid activation. The model was optimized using Adam with binary cross-entropy as the loss function.

Table 1. DNN Model Configuration

Parameter	Value
Hidden Layer 1	64 neurons
Hidden Layer 2	32 neurons
Activation Function	ReLU
Output Activation	Sigmoid
Optimizer	Adam
Loss Function	Binary Cross-Entropy
Epochs	100
Batch Size	64

The DNN architecture is consistent with the use of neural networks for nonlinear pattern learning and supervised classification (Goodfellow, 2016; LeCun *et al.*, 2015). The Autoencoder was implemented as an unsupervised model to learn representations of AWS sensor observations through encoding and decoding processes. Anomalies were identified based on reconstruction error. A threshold was applied to the reconstruction error to distinguish normal observations from anomalous observations. In this study, the anomaly threshold was determined using the mean reconstruction error plus three standard deviations (3-sigma). This threshold was used to classify observations with unusually large reconstruction errors as anomalies. All model development, training, and evaluation processes were implemented using Python 3.10. TensorFlow and Keras were used to construct and train the deep learning models, while scikit-learn was used for data preprocessing and calculation of evaluation metrics.

Table 2. Autoencoder Model Configuration

Parameter	Value
Encoder Layer	32 neurons
Bottleneck Layer	16 neurons
Decoder Layer	32 neurons
Activation Function	ReLU
Output Activation	Linear
Optimizer	Adam
Loss Function	Mean Squared Error (MSE)
Batch Size	64

The use of Autoencoders for reconstruction-based anomaly detection is supported by Sakurada and Yairi (2014), while encoder-decoder approaches have also been applied to multivariate sensor time-series anomaly detection (Malhotra *et al.*, 2016).

3.7 Model Evaluation

Model performance was evaluated using the confusion matrix, accuracy, precision, recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (ROC-AUC). These metrics were used to compare the ability of the DNN and Autoencoder to distinguish normal and anomalous observations. In addition to model-level evaluation, the study analyzed the sensor parameters associated with detected anomalies. The sensor ranking was determined from the frequency of anomaly occurrences associated with each parameter. The resulting ranking was used to identify parameters that require greater attention in sensor monitoring and maintenance.The analysis provides information that can support sensor maintenance prioritization and AWS data-quality improvement.

4. Result and Discussion

4.1 Results

This study evaluated the performance of the DNN and Autoencoder models in detecting anomalies in AWS Ancol North Jakarta data. The evaluation was performed using the test dataset and included accuracy, precision, recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (ROC-AUC).

Table 3. Performance Comparison of Anomaly Detection Models

Metric	DNN	Autoencoder
Accuracy	99.6%	95.8%
Precision	0.922	0.578
Recall	0.996	0.023
F1-Score	0.958	0.043
ROC-AUC	1.000	0.584

Based on Table 3, the DNN achieved higher performance than the Autoencoder across all reported evaluation metrics. The DNN achieved an accuracy of 99.6%, precision of 0.922, recall of 0.996, F1-score of 0.958, and ROC-AUC of 1.000. The Autoencoder achieved an accuracy of 95.8%, precision of 0.578, recall of 0.023, F1-score of 0.043, and ROC-AUC of 0.584. The test dataset consisted of 77,048 observations, including 3,287 anomalous records and 73,761 normal records. The dataset therefore had a substantial class imbalance, with anomalous observations representing a relatively small proportion of the test data.

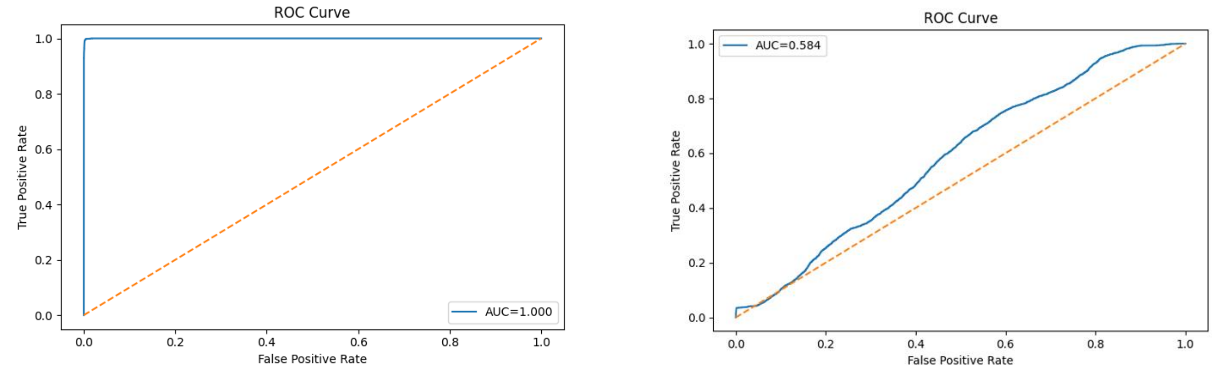

Figure 4. ROC Curve for DNN and Autoencoder

The confusion matrix results provide further information about the classification performance of the two models.

Table 4. Detailed Evaluation Metrics and Sensor Anomaly Detection Results

Parameter	DNN	Autoencoder
Accuracy	99.6%	95.8%
Precision, Class 1	0.922	0.578
Recall, Class 1	0.996	0.023
F1-Score, Class 1	0.958	0.043
ROC-AUC	1.000	0.584
True Positive (TP)	3,275	74
False Negative (FN)	12	3,213
Sensor Parameters with Detected Anomalies	9	8
Total Sensor-Level Anomaly Occurrences	3,261*	103*
Dominant Anomaly Parameter	Wind Direction	Wind Direction
Learning Method	Supervised	Unsupervised
Label Data	Yes	No

The definition and calculation of total sensor-level anomaly occurrences should be clearly specified in the methodology. The total dataset after preprocessing consisted of 385,237 observations. The data were divided into 308,189 training observations and 77,048 testing observations. The evaluation was conducted using the 77,048 observations in the test dataset. For the DNN, the model was trained using observations labeled as normal (0) and anomalous (1). The model generated a probability for each test observation, which was converted into a class using a threshold of 0.5. The DNN correctly identified 3,275 anomalous observations and produced 12 false-negative observations. For the Autoencoder, anomaly detection was performed using reconstruction error. The model reconstructed each test observation, and the Mean Squared Error (MSE) between the original and reconstructed observations was calculated. Observations with higher reconstruction errors were considered to have greater deviations from the learned data patterns. In this study, a threshold based on the mean reconstruction error plus three standard deviations was used to classify observations with high reconstruction errors as anomalies. Based on the reported confusion matrix, the Autoencoder detected 74 of the 3,287 labeled anomalous observations, while 3,213 anomalous observations were classified as false negatives. In addition to model performance, the study analyzed the sensor parameters associated with the detected anomalies. The analysis was based on the frequency of anomaly occurrences associated with each sensor parameter.

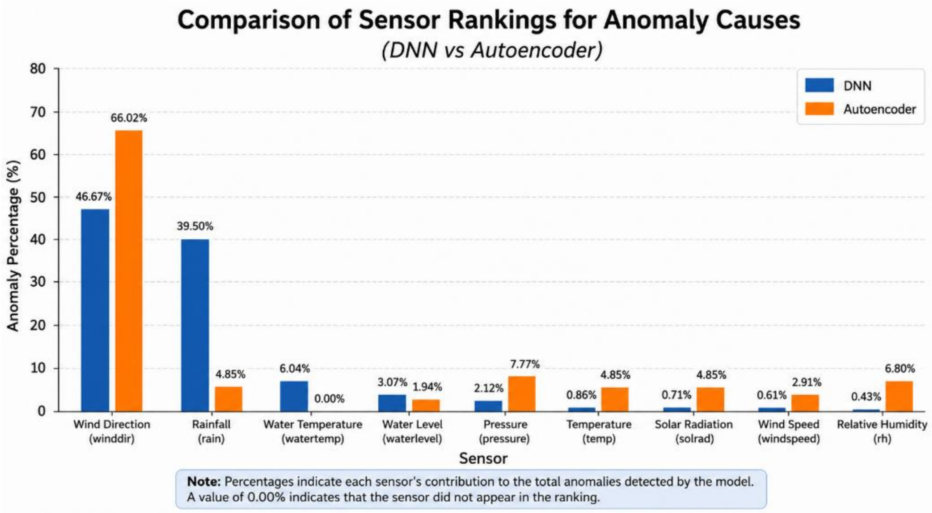

Figure 5. Sensor Anomaly Ranking: DNN and Autoencoder

The results show that wind direction was the most frequently associated parameter with detected anomalies for both models. For the DNN, wind direction accounted for 46.67% of the reported sensor-level anomaly occurrences, followed by rainfall at 39.50%. For the Autoencoder, wind direction accounted for 66.02% of the reported sensor-level anomaly occurrences. The DNN detected anomalies across all nine sensor parameters, whereas the Autoencoder detected anomalies across eight parameters. Wind direction was the dominant anomaly-associated parameter for both models.

4.2 Discussion

The results show a substantial difference between the DNN and Autoencoder in detecting anomalies in the AWS dataset. The DNN achieved higher accuracy, precision, recall, F1-score, and ROC-AUC, with a recall of 0.996 and an F1-score of 0.958. This higher performance is related to its supervised learning mechanism, which uses labeled observations to learn patterns associated with normal and anomalous conditions. In contrast, the Autoencoder detects anomalies based on reconstruction error without directly using anomaly labels during training (Malhotra *et al.*, 2016; Sakurada & Yairi, 2014). Under the configuration used in this study, the Autoencoder produced a recall of 0.023, indicating that most of the labeled anomalous observations were not detected. The relatively low recall may be related to the reconstruction-based detection mechanism and the use of the mean reconstruction error plus three standard deviations as the anomaly threshold. This threshold requires relatively large deviations before an observation is classified as anomalous. The difference between the two models should be interpreted according to the characteristics of the dataset and experimental configuration. The DNN achieved an ROC-AUC of 1.000, while the Autoencoder achieved 0.584. The perfect AUC of the DNN reflects its performance on the labeled dataset and test procedure used in this study and should not be interpreted as evidence of perfect performance under all AWS operating conditions. The results therefore indicate that the DNN was more effective for detecting the labeled anomalies in the AWS Ancol dataset. However, further evaluation using independent AWS datasets is required to determine whether this performance can be maintained under different operating conditions.

The sensor-level analysis identified wind direction as the parameter most frequently associated with anomalies in both models. Wind direction accounted for 46.67% of the reported sensor-level anomaly occurrences in the DNN, while rainfall accounted for 39.50%. In the Autoencoder results, wind direction accounted for 66.02% of the reported occurrences. The consistency of wind direction as the dominant parameter indicates that this sensor requires particular attention in the monitoring of AWS Ancol data. However, this finding does not establish that the wind direction sensor is the direct physical cause of the detected anomalies. Field inspections and maintenance records would be required to determine whether the anomalies resulted from sensor malfunction, environmental conditions, or other factors.

The findings also have implications for AWS monitoring and maintenance. The DNN is more suitable for the current experimental setting because labeled anomaly data were available and the model achieved substantially higher detection performance. The Autoencoder, although less effective under the tested configuration, may still serve as a complementary approach for identifying observations with large deviations, particularly when labeled anomaly data are limited. The sensor ranking can also support maintenance prioritization by directing attention to parameters that repeatedly appear in anomaly detection results. Overall, the findings are specific to the AWS Ancol dataset, labeling procedure, model configurations, and detection threshold applied in this study and should not be generalized to other AWS stations without further validation.

5. Conclusion and Recommendations

This study developed an early-detection approach for anomalies in Automatic Weather Station (AWS) sensor data using DNN and Autoencoder models. Using 385,237 observations from the AWS Ancol, North Jakarta, collected from January to September 2025, the DNN achieved the best performance, with an accuracy of 99.6%, precision of 0.922, recall of 0.996, F1-score of 0.958, and ROC-AUC of 1.000. The Autoencoder achieved an accuracy of 95.8%, precision of 0.578, recall of 0.023, F1-score of 0.043, and ROC-AUC of 0.584. These results indicate that the supervised DNN was more effective than the Autoencoder in detecting the labeled anomalies in the dataset used in this study. The sensor-level analysis identified wind direction as the parameter most frequently associated with detected anomalies in both models, accounting for 46.67% of the reported sensor-level anomaly occurrences for the DNN and 66.02% for the Autoencoder. This consistency indicates that wind direction should receive particular attention in AWS data monitoring and subsequent sensor inspection. However, the result does not establish that the wind direction sensor was the direct physical cause of the anomalies.

The findings suggest that supervised DNNs are more suitable for the present AWS dataset because labeled anomaly data were available. The Autoencoder may still be used as a complementary approach, particularly when labeled anomaly data are limited. Further research should evaluate the models using data from multiple AWS stations and temporal validation to assess generalization. Future studies may also compare the proposed models with other approaches, such as LSTM, Isolation Forest, XGBoost, and Random Forest, and investigate hybrid methods that combine supervised and unsupervised anomaly detection.

References

Akbar, G., Prajitno, P., Ariffudin, & Ananda, N. (2024). Multivariate imputation chained equation on solar radiation in automatic weather station. *Jurnal Penelitian Pendidikan IPA, 10*(7), 3633–3639. https://doi.org/10.29303/jppipa.v10i7.7679

Ariffudin, A., & Musa, P. (2022). Analisa sistem komunikasi data berbasis Internet of Things (IoT) menggunakan metode PIECES pada sistem pengamatan cuaca otomatis di Badan Meteorologi Klimatologi dan Geofisika (BMKG). *Jurnal Meteorologi dan Geofisika, 23*(2), 81. https://doi.org/10.31172/jmg.v23i2.831

Duraj, A., Szczepaniak, P. S., & Sadok, A. (2025). Detection of anomalies in data streams using the LSTM-CNN model. *Sensors, 25*(5), 1610. https://doi.org/10.3390/s25051610

Goodfellow, I. (2016). *Deep learning*. MIT Press.

Heaton, J. (2018). Ian Goodfellow, Yoshua Bengio, and Aaron Courville: *Deep learning: The MIT Press, 2016, 800 pp, ISBN: 0262035618*. *Genetic Programming and Evolvable Machines, 19*(1), 305–307.

Kaya, Ş. M., İşler, B., Abu-Mahfouz, A. M., Rasheed, J., & AlShammari, A. (2023). An intelligent anomaly detection approach for accurate and reliable weather forecasting at IoT edges: A case study. *Sensors, 23*(5), 2426. https://doi.org/10.3390/s23052426

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.

Malhotra, P., Ramakrishnan, A., Anand, G., Vig, L., Agarwal, P., & Shroff, G. (2016). LSTM-based encoder-decoder for multi-sensor anomaly detection. *arXiv*. https://arxiv.org/abs/1607.00148

Pahlepi, R., Soekirno, S., & Wicaksana, H. S. (2023). Solar radiation intensity imputation in pyranometer of automatic weather station based on long short term memory. *Ultima Computing: Jurnal Sistem Komputer*, 35–40. https://doi.org/10.31937/sk.v15i2.3348

Rhamadania, K., Makhsun, M., Basir, C., & Ananda, N. (2026). Imputation of missing weather data in automatic weather station using the GRU algorithm. *Journal of Applied Informatics and Computing, 10*(2), 1165–1171. https://doi.org/10.30871/jaic.v10i2.12301

Sakurada, M., & Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the MLSDA 2014 2nd Workshop on Machine Learning for Sensory Data Analysis* (pp. 4–11). Association for Computing Machinery. https://doi.org/10.1145/2689746.2689747

Sevruk, B., Ondrás, M., & Chvíla, B. (2009). The WMO precipitation measurement intercomparisons. *Atmospheric Research, 92*(3), 376–380. https://doi.org/10.1016/j.atmosres.2009.01.016

Utku, A., & Kaya, S. K. (2025). LSTM-based deep learning model for air temperature prediction. *IDUNAS Natural & Applied Sciences, 26*(32), 26–32. https://doi.org/10.38061/idunas.1596669

Wicaksana, H. S., Winarko, S., & Kurniadi, D. (2023). Treatment on missing data of wind speed measurement in automatic weather station using multivariate imputation chained equation. In *2023 8th International Conference on Instrumentation, Control, and Automation (ICA)* (pp. 81–85). IEEE. https://doi.org/10.1109/ICA58538.2023.10273106

World Meteorological Organization. (2021). *Guide to instruments and methods of observation (WMO-No. 8)*.

Yoon, S., Lee, Y., Lee, J.-G., & Lee, B. S. (2022). Adaptive model pooling for online deep anomaly detection from a complex evolving data stream. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (pp. 2347–2357). Association for Computing Machinery. https://doi.org/10.1145/3534678.3539348