Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 10 Issue 01
2025
Research Article
https://doi.org/10.56979/1001/2025/1127

Multivariate Air Pollution Anomaly Detection via LSTM Autoencoders on Beijing Multi-Site Sensor Data

Muhammad Talha Gulzar[1], Khushbu Khalid Butt[2*], Maria Tariq[1], Sundus Muir[3], and Omer Irshad[4]

[1]Department of Computer Sciences, Lahore Garrison University, Lahore, Pakistan.
[2]Department of Information Technology, Lahore Garrison University, Lahore, Pakistan.
[3]Department of Criminology, Lahore Garrison University, Lahore, Pakistan.
[4]Department of Software Engineering, Lahore Garrison University, Lahore, Pakistan.
[*]Corresponding Author: Khushbu Khalid Butt. Email: drkhushbukhalid@lgu.edu.pk

Received: September 02, 2025 **Accepted:** October 28, 2025

Abstract: The current paper investigates the application of a Long Short-Term Memory Autoencoder (LSTM-AE) to identifying anomalies in the multi-variate time-series of air-pollution measurements. The algorithm was used to the sensor data of $PM_{2.5}$, CO, O_3, NO_2, TEMP, and WSPM of the Beijing Multi-Site Air-Quality Data Set described on Kaggle. The test set included fifty synthetic anomalies that were used to assess performance. The anomalies were identified using the reconstruction error calculated using Mean Squared Error (MSE) as a dynamic threshold value of 0.002957. The presented model produced the Precision, Recall, F1-Score, and ROC-AUC of 77.00%, 100.00%, 87.01%, and 99.86%, respectively, proving its effectiveness in detecting minor and drastic changes in the pattern of air quality.

Keywords: Air Pollution; Anomaly Detection; Time-Series Analysis; LSTM Autoencoder; Deep Learning; Environ- Mental Monitoring; Multivariate Sensor Data; Reconstruction Error; Unsupervi sed Learning, Beijing Air Quality Data

1. Introduction

Air pollution is considered to be one of the most severe environmental and health problems on the universal scale [1]. It is particularly acute in the industrial cities with a high population density, which could result in developing respiratory illnesses, cardiovascular issues and environmental deterioration due to the presence of massive amountsof pollutants, such as PM2.5, CO, NO2, and O3 [2]. Real-time monitoring of the air quality permits the authorities to take immediate measures that would safeguard the health of people [3].

Contemporary sensor-based networks have the capacity to take recordings continuously of numerous pollu- tants and other meteorological variables in other positions [4]. But usually noise, missing values, and abrupt changes are problems in the collected data [5]. Identification of such anomalies is key to defining both detection of pollution outbreaks as well as verification of sensor accuracy and interpretation of rare event needs attributable to either weather or respective industrial activity [6].

Conventional anomaly detectors operate on constant thresholds or pre-structured rules by the subject matter experts [7]. Although these methods are helpful, they tend to be inflexible and might break rules to exhibit compli- cated dynamics and/or respond to the factors of fluctuating conditions [8]. In addition, most of the environmental data are unlabeled, and therefore, supervised techniques are impractical [9].

Deep learning techniques are a potential alternative to handle such a challenge [10]. Specifically, anomaly detection in time-series data is effectively performed using Long Short-Term Memory Autoencoders (LSTM-AEs) [11]. LSTM networks have the ability to learn temporal information about sequences and when auto encoders are used, they can be combined and used to recreate normal behavior patterns [12]. Any major variation in the reconstructive process can be held as an anomaly [13].

The paper suggests an unsupervised reconstruction-based anomaly detector based on an LSTM Autoencoder (LSTM-AE). The model is only trained using normal multivariate sequences of air-pollution and meteorological variables whereby the model learns normal temporal trends. Testing identifies sequences whose reconstruction error is abnormally high, which are considered anomalous. Anomalous samples during training do not have any labels; the synthetic anomalies are only injected during the test set, in order to assess performance. This renders the approach a conventional unsupervised anomaly detection method as opposed to semi-supervised method [14].

To assess our model, we consider the dataset of the Beijing Multi-Site Air Quality with several years of measurements of pollutants and weather markers in multiple monitoring sites [15]. The challenges presented by this dataset are very realistic being that it adds dependencies over time and seasonality along with missing values [16]. To evaluate the model, this proposed model is tested on the synthetic anomalies to check its performance, and the results turn out to be good precision, recall, and ROC-AUC scores, depicting its sound aspects in detecting abnormal patterns in air quality [17].

This study would result in a deep learning replica of a Long Short Term memory Autoencoder (LSTM- AE) to identify anomalies in multivariate air pollution data [10]. The model is trained using normal sequences and identifies the anomalies on the basis of reconstruction error [13]. Anomalies based on synthesis are also inserted into test set to test the effectiveness of the model. The assessment of performance is carried out through the same standardized measures as Precision, Recall, F1-Score and ROC-AUC. It seeks to demonstrate that LSTM-AE can be successfully extracting abnormal patterns in real-life air quality data [5].

2. Related Work

Also, the topic of anomaly detection of time-series data has been deeply investigated especially in the field of environmental monitoring [5]. Malhotra et al. (2015) were the first to apply Long Short-Term Memory (LSTM) networks to the problem of anomaly detection in time series, and this task proved to be an effective learning problem, where the LSTM approach achieved success. This method created a basis to implement LSTM based models in different fields, such as air quality monitoring [10].

Recent experiments performed by Zhou et al. (2013) [14] and Tabassum et al.(2021) [21], demonstrated that semi-supervised deep learning can be effective with relatively small amounts of labeled data and this result is especially applicable to real-world environmental data where anomalies are poorly labeled [14]. Continuing this thought, some literature uses autoencoder-based models namely LSTM-Autoencoders in identifying abnormal trends in multivariate time-series data. Such models have been able to perform well in industrial use-case and sensor fault diagnosis since they are able to capture complex temporal patterns and indicate anomalies of normal behaviour [10]. Researchers using unsupervised methods have been applied to the scenario of air pollution where Zhang and

Audibert have used unsupervised methods to identify an abnormal trend in multivariate sensor records [5]. The models are effective even in the absence of annotations of anomalies and can be adjusted to different conditions in the environment. Nevertheless, a few studies have implemented LSTM based methods to multi site air quality information with synthetic anomalies. The study fills that gap by applying LSTM-Autoencoder to identify anomalies in pollution data at Beijing and testing it against synthetic anomalies.

Table 1. Samples of Related Work on Anomaly Detection in Time-series

Paper	Year	Method	Key Contribution
Malhotra et al.	2015	LSTM	Proposed LSTM as an anomaly detector on time-series data and pointed out the possibility to learn temporal relationships [8].
Zhou et al.	2013	Semi-	Demonstrated that it is possible to learn LSTM

		supervised LSTM	using little la- beled data [14].
Audibert et al.	2020	Autoencoder	Put forward a non- supervised scheme to identify anomalies in multivariate time se- ries [22].
Zhang et al.	2019	Unsupervised Learning	Implemented unsupervised approaches in detecting anomalies within air quality data [5].

Table 2. Evaluation Metrics Used in the Anomaly Detection Model

Matric	Description
Precision	Calculates the percentage of the number of correct identified anomalies of the number of de- tected anomalies. A high preci- sion implies that there is less false positive rate [18].
Recall	Calculates the amount of properly detected anomalous instances as a percentage of all actual anomalous instances. high recall is synonymous with a low false negative rate [18].
F1-Score	Precision and recall harmonic mean. It optimizes the tradeoff between false positive and false negative trade-off [19].
ROC-AUC	The model capability of distin- guishing between classes. It is desirable to have a high Area Under the Curve (AUC) and that shows greater separability [17].
Confusion Matrix	A table representing performance of a classification model, of val- ues of coutputtp, coutputtn, cout- putfp, coutputfn in the form of a table, displaying values of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) [20].

3. Materials and Methods

In this study, the researcher will utilize the Beijing Multi-Site Air-Quality Data Set found in the Kaggle site [15], since it includes hourly quality air data of the various monitoring stations in Beijing over the period of 2013 to 2017. The records have six essential characteristics of PM2.5, CO, O3, NO2, temperature (TEMP), and wind speed (WSPM) that provide an overview of the atmospheric conditions in the city.

In order to maintain consistency between the monitoring stations, data was standardized with MinMax Scaler (handling IoT data in the best practice [23]) using a training set to prevent data leakage [24]. Multivariate sequences (24x6) were extracted using a 24-hour sliding window with a 6-hour stride, effectively capturing the difference between days in order to be used as inputs into an LSTM-based model [25]. This method divides the unstructured time series data, e.g. PM2.5, into structured inputs as shown in Figure 1, so that the model can learn temporal patterns localized in time to detect anomalies correctly [8].

Table 3. First Five Rows of Raw Air Quality Data from Wanliu Station (March 2013)

No	Year	Month	Day	Hour	PM2.5	Station	CO	O3	NO2	TEMP	WSPM
1	2013	3	1	0	8.0	Wanliu	400.0	52.0	28.0	-0.7	4.4
2	2013	3	1	1	9.0	Wanliu	400.0	50.0	28.0	-1.1	4.7
3	2013	3	1	2	3.0	Wanliu	400.0	55.0	19.0	-1.1	5.6
4	2013	3	1	3	11.0	Wanliu	NaN	NaN	14.0	-1.4	3.1
5	2013	3	1	4	3.0	Wanliu	300.0	54.0	NaN	-2.0	2.0

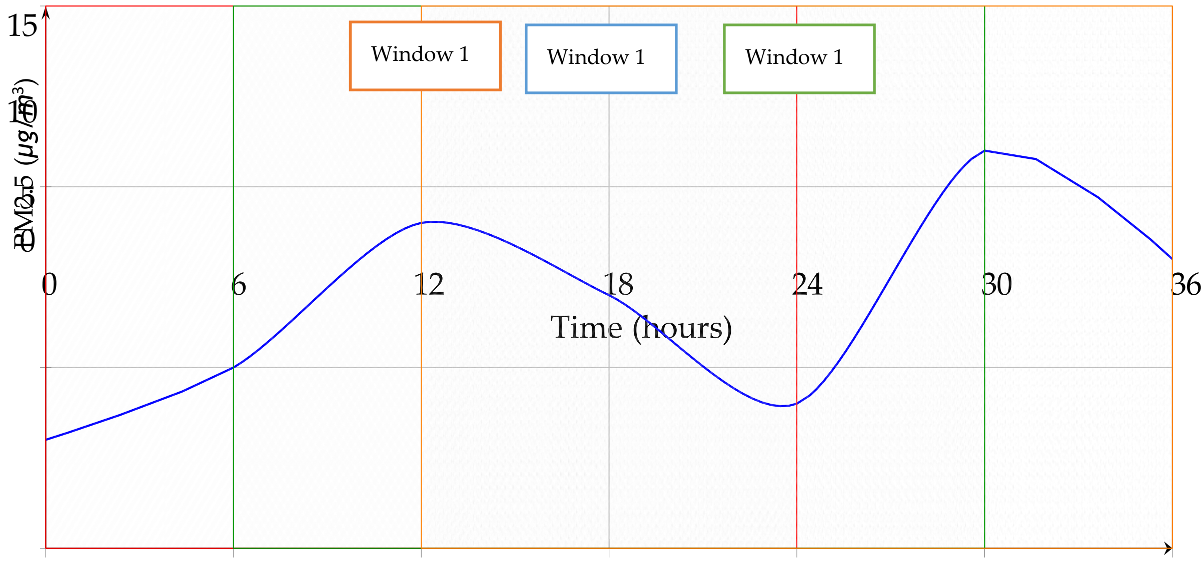


Figure 1. Sliding Window (24h Window, 6h Stride)

Fig. 1: Multivariate sliding window extraction of air quality. In every 24 hours (colored areas) there are 6 characteristics (PM2.5, CO, O3, etc.). The 6-hour stride results in the overlap of windows by 18 hours (75 % overlap). Stride length and window size are expressed clearly by using the lower arrows.

In order to eliminate temporal leakage, the dataset was chronologically divided into 70% training, 10% validation and 20% testing. The data of all the monitoring stations was merged into one multivariate data. Linear interpolation of the pollutant variables (NaNs) were applied at the individual station and then the boundary cases were filled by forward/backward filling. Every row that was not completed was eliminated after interpolation. The scaling consisting of MinMaxScaler was carried out on the training set and the trained scaler was used on the validation and test sets.

To test the model, a 10% alteration of the test windows was performed to generate synthetic anomalies to determine the performance of detection. Such anomalies were created either by adding sudden spikes, contaminating pollutant values by random values between 2 and 5, or by introducing short-term irregular variations. These mutations simulate real-world pollution occurrences and, at the same time, are designed in such a way that polluted sequences do not escape the test split.

An unsupervised LSTM-AutoEncoder that was trained on normal air pollution data to learn to re-create typical patterns [10]. In the process of inference, more errors tend to be experienced in the reconstruction procedure when anomalous sequences are encountered; such measures are expressed as MSE [13]. The anomalies are classified by a threshold based on training errors. The method successfully captures time patterns and is useful in detecting anomalies in real world data (both large and small) [5].

Sensitivity test was done by calculation of threshold with the help of the following statistical rule:

Threshold = μtrain + 3σtrain

where μtrain and σtrain are mean and the standard deviation of the training reconstruction errors, respectively [7]. Those samples above this threshold were treated as anomalous.

In the training of a model, the Adam optimizer was used with the learning rate set to 0.001 [26], and a loss function of Mean Squared Error (MSE), given by the eq 2 [27]. The training was performed on 20 epochs, size of the batch was 64 [28].

The reconstruction error ET which is the mean squared error between the input xt and the reconstruction x^t (Equation 2) is the metric of how well the model is approximating normal patterns [13]. The greater the errors, the greater the possibility of anomalies and this is used as the foundation of the detection strategy [7].

The LSTM-Autoencoder design of this study is expressed in Table IV. The encoder is used to compress multivariate time-series input, and decoder reconstructs the same. The design is capable of capturing temporal and pollutant interactions effectively and hence, it is suitable to detect the anomalies [11].

The architecture has been useful in capturing temporal dynamics and interaction of pollutants [29], and it allows the reconstruction of normal sequences effectively and magnification of errors occur once anomalies take place [13].

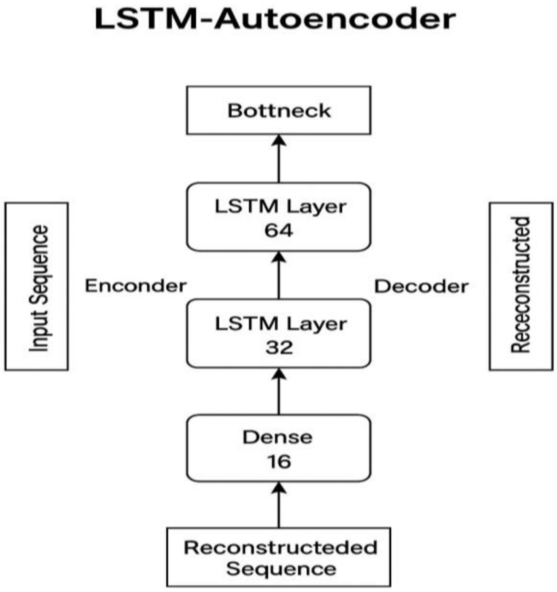


Figure 2. Architecture of the LSTM-Autoencoder

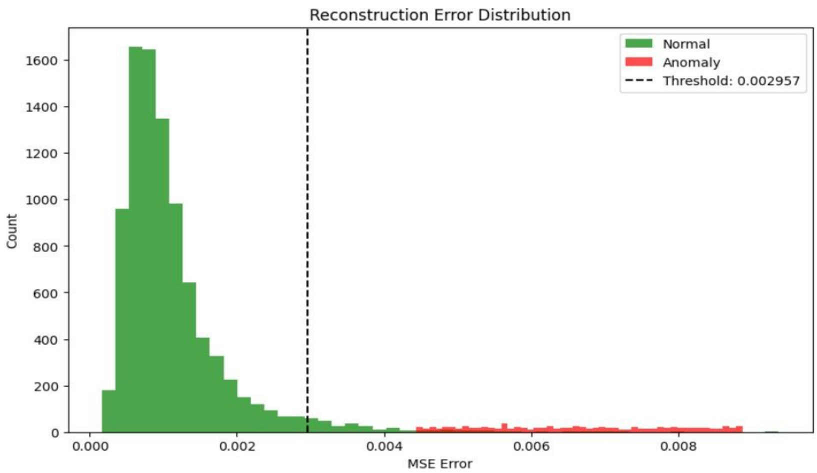


Figure 3. Histogram showing reconstruction error distributions for normal vs. anomalous samples, including the threshold line

The experiments were performed using Google Colab with Tesla T4 GPU and python 3.10. The model utilized several libraries amongst which were PyTorch (model training) [28], NumPy and Pandas (data processing), Scikit-learn (normalization, metrics), and Matplotlib (plotting) [24].

Table 4. Overview of LSTM Autoencoder Architecture

Component	Description
Encoder	The temporal LSTM features are extracted through two LSTM layers that take the input of a dimension of 24 × 6 and the output is a 16- dimensional vector through using a dense layer.
Decoder	The bottleneck is broadened through dense layer and reshaped which undergoes two LSTM layers in order to recover the initial input sequence.

4. Results

The model has done well(similar to multi-cloud detection applications [30]) Precision of 77.00%, Recall of 100.00%, F1 Score of 87.01%, and ROC-AUC of 99.86%, which means that it identifies them accurately and they are balanced. As an imbalance measure, the F1 Score is a harmonic mean of Precision and Recall that provides a balanced analysis where imbalance exists as well [20].

F1 Score =2* (Precision* Recall)/ (Precision*Recall) (3)

Precision is the number of accurately found anomalies divided by the number of all positives [19]. The greater the precision, the less false positives and more accurate detection there is.

Precision = (True Positives) / (True Positives + False Negatives) (4)

Recall or sensitivity is a quotient of well-identified real positives. It is an indication of how accurate the model might be in finding out all the anomalies as far as the data is concerned [18]. The perfect level of recall does not reveal the presence of false negative level.

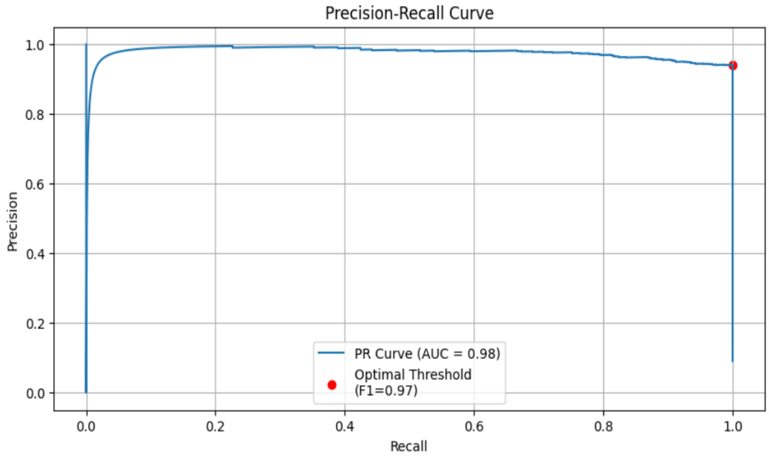
Recall= (True Positives) / (True Positives + False Negatives) (5)

The Precision-Recall shows the existence of a trade-off between Precision and Recall at various values of threshold which is useful to understand the anomaly detection performance of the model (Figure 4) [17].

Figure 5 shows the ROC-AUC score which measures the performance of the model in regard to its capability to differentiate between normal and anomalous sequences. AUC being 99.86 % is quite good classification [17] [20].

Confusion matrix (Figure 6) is a summary of the performance of the model in classification and it will display True Positives, False Positives, True Negatives [20], and False Negatives. It points out that the model is quite efficient in the detection of anomalies and minimization of errors [31].

As shown in Figure 7, the evolution of the anomaly detection by the model with time indicates when anomalous patterns are identified and gives us an idea how well the model performed in this respect over time.



. **Figure 4.** Precision-Recall Curve

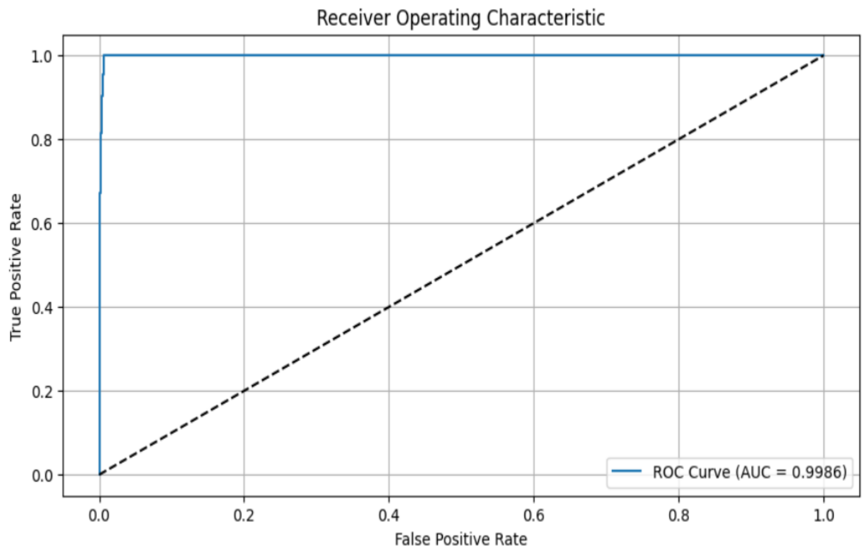


Figure 5. Receiver Operating Characteristic (ROC) Curve

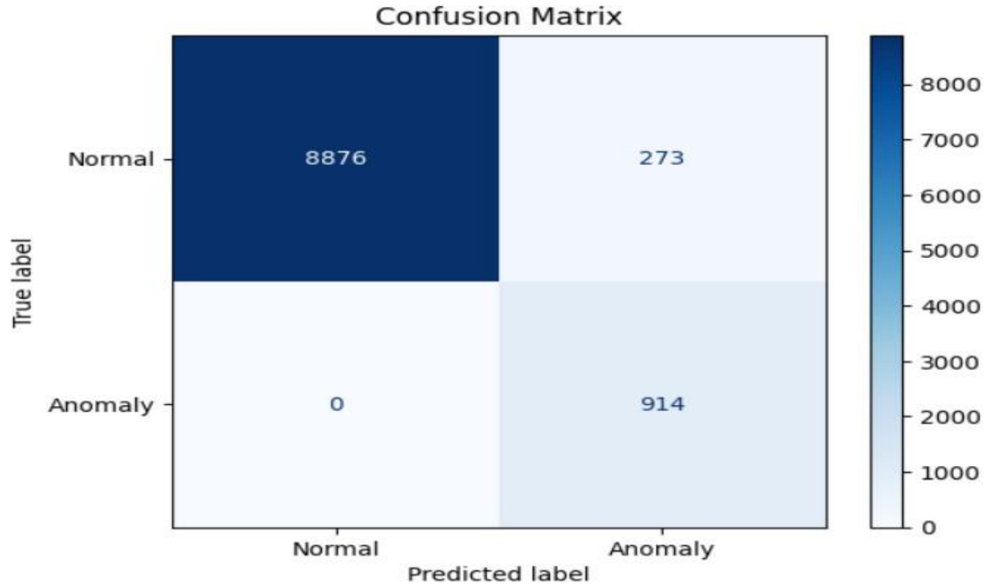


Figure 6. Confusion Matrix

The feature-wise reconstruction of the first test sample as seen in Figure 8 demonstrates that the model was able to learn and train each feature present in the time-series data with great accuracy and is therefore capable of recreating them in a precise manner [13].

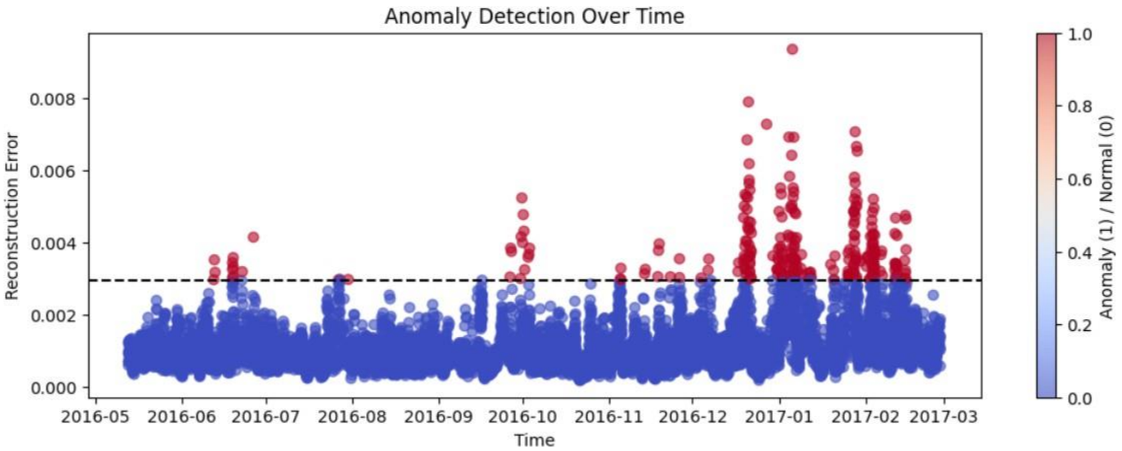


Figure 7. Anomaly Detection Over Time

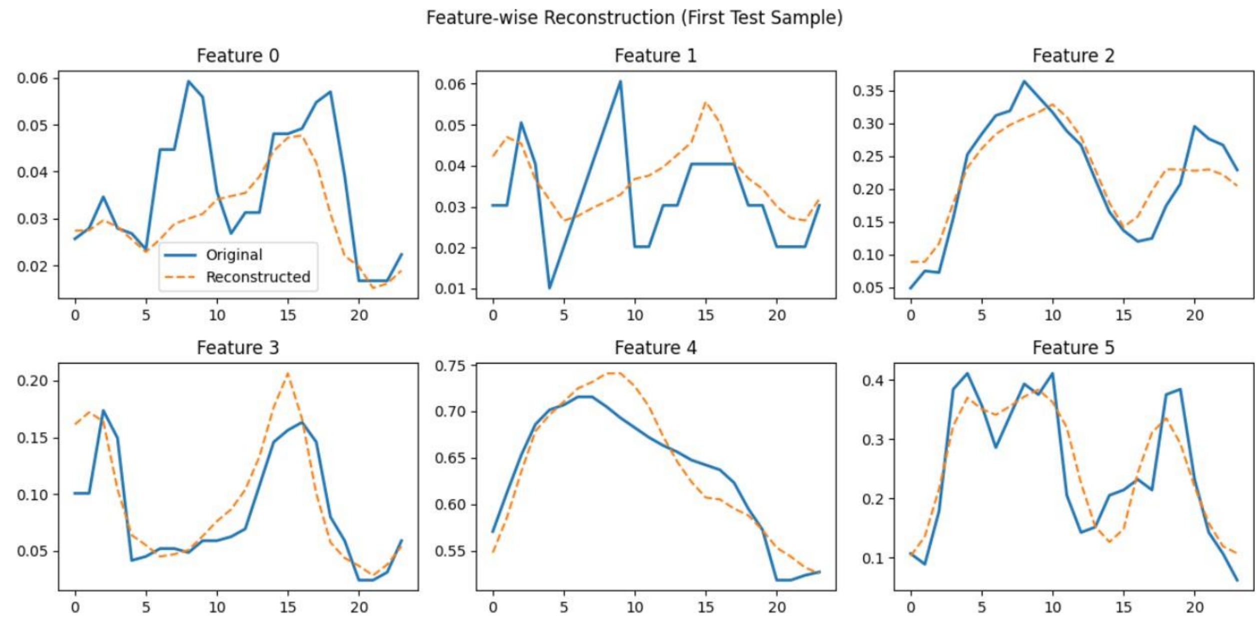


Figure 8. Feature-wise Reconstruction (First Test Sample)

The training loss curve given in Figure 9 indicates the convergence of the model under training. The constant and even loss reduction proves that the model was successfully reducing the amount of reconstruction error [24].

Table V gives the loss values of the LSTM- Autoencoder after the 20 training epochs [24]. In the model, the Precision was 77.00 percent, Recall was 100.00 percent, F1-Score was 87.01 percent, and ROC-AUC was 99.86 percent, which is a good indicator of similar results across anomaly detection performance. Threshold to detect the anomaly was calculated based on the reconstruction error at 0.002957. The confusion matrix displayed the number of true positive structural anomalies, false positive structural anomalies, true negative structural anomalies, and false negative structural anomalies of 914, 273, 8,876, and zero respectively. The model had a 100% recall and high ROC-AUC value of 99.86 which translates to the model being able to detect all injected anomalies and the model is capable of differentiating between normal and abnormal sequences. Nonetheless, the accuracy of 77% (273 false positives) shows that the rate of false-alarm is high. This trade-off is typical of detectors based on reconstruction, which are unsupervised detectors, and there is a tendency to detect as many anomalies as possible, at the expense of classifying as anomalies some ordinary windows. As such despite having a perfect recall, the precision of the model demonstrates that better threshold tuning or the hybrid approach should be used to minimize false positives.

Table 5. LSTM-Autoencoder after 20 training epoch's assessment.

Metric	Value
Threshold	0.002957
Precision	77.00%
Recall	100.00%
F1-Score	87.01%
ROC-AUC	99.86%
True Positives	914
False Positives	273
True Negatives	8876
False Negatives	0

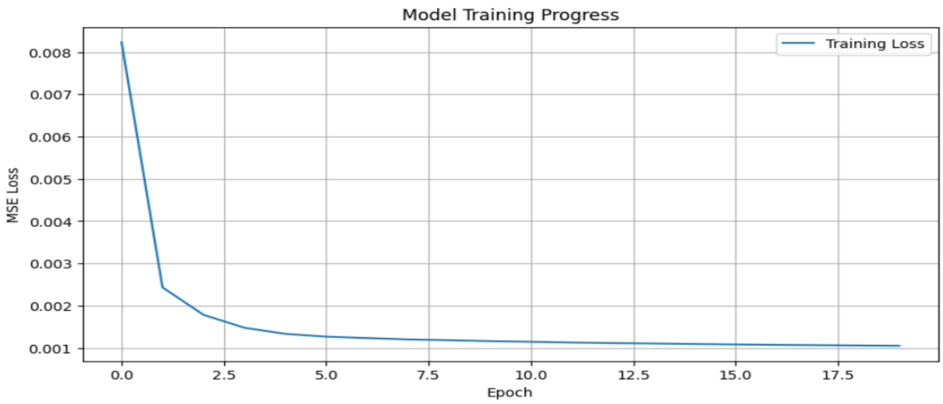


Figure 9. Training Loss Curve

5. Conclusion and Future Work

This study introduced an unsupervised, reconstruction-oriented anomaly detection model utilizing an LSTM Autoencoder (LSTM-AE) for multivariate air pollution data derived from the Beijing Multi-Site Air Quality dataset. The model was trained solely on normal sequences obtained via a 24-hour sliding window and assessed using synthetic anomalies introduced exclusively in the test split. The LSTM-AE learns the normal temporal patterns of pollutants and weather features very well, as shown by a high ROC-AUC of 99.86% and a perfect Recall of 100%, which means that all injected anomalies were found.

The model also had a Precision of 77%, which means it had 273 false positives. This means that the false-alarm rate was fairly high. This trade-off between perfect recall and moderate precision is typical in reconstruction-based unsupervised methods, which are meant to be very sensitive but might mistake normal changes for anomalies. So, even though the model works well to find rare or sudden pollution events, the fact that it sometimes gives false positives means that threshold selection and post-processing strategies need to be improved.

In future endeavors, the incidence of false positives may be diminished by investigating alternative thresholds, station-specific modeling, probabilistic reconstruction metrics, or hybrid models that integrate reconstruction and classification. Testing the model on real labeled anomalies and adding spatial features or transformer-based architectures might also make it more stable.

References

1. H. M. Tran, F.-J. Tsai, Y.-L. Lee, J.-H. Chang, L.-T. Chang, T.-Y. Chang, K. F. Chung, H.-P. Kuo, K.-Y. Lee, K.-J. Chuang, et al., "The impact of air pollution on respiratory diseases in an era of climate change: A review of the current evidence," Science of the Total Environment, vol. 898, p. 166340, 2023.
2. H. N. H. Nawaz, M. U. M. Umar, F. P. F. Parvaiz, M. T. S. M. T. Shehzad, and I. Afraz, "Impacts of climatic changes and air pollution on public health and environment," Journal of Health and Climate Change, vol. 3, no. 1, 2024.
3. K. Zheng, S. Zhao, Z. Yang, X. Xiong, and W. Xiang, "Design and implementation of lpwa-based air quality monitoring system," IEEE Access, vol. 4, pp. 3238–3245, 2016.
4. N. Laptev, S. Amizadeh, and I. Flint, "Generic and scalable framework for automated time-series anomaly detection," in ACM SIGKDD, pp. 1935–1944, 2017.
5. J. Zhang, Y. Chen, and X. Li, "Unsupervised anomaly detection in air quality monitoring networks," Environmental Science & Technology, vol. 53, pp. 10845–10854, 2019.
6. Y. Su, Y. Zhao, C. Niu, R. Liu, W. Sun, and D. Pei, "Robust anomaly detection for multivariate time series through stochastic recurrent neural network," in Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, pp. 2828–2837, 2019.
7. K. Hundman, V. Constantinou, and C. e. a. Laporte, "Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding," ACM SIGKDD, pp. 387–395, 2018.
8. P. Malhotra, L. Vig, G. Shroff, P. Agarwal, et al., "Long short term memory networks for anomaly detection in time series," in Proceedings, vol. 89, p. 94, 2015.
9. Y. Ouali, C. Hudelot, and M. Tami, "An overview of deep semi-supervised learning," arXiv preprint arXiv:2006.05278, 2020.
10. X. Ding, J. Wang, Y. Liu, and U. Jung, "Multivariate time series anomaly detection using working memory connections in bi-directional long short-term memory autoencoder network," Applied Sciences, vol. 15, no. 5, p. 2861, 2025.
11. Y. Wei, J. Jang-Jaccard, F. Sabrina, W. Xu, S. Camtepe, and A. Dunmore, "Reconstruction-based lstm-autoencoder for anomaly-based ddos attack detection over multivariate time-series data," arXiv preprint arXiv:2305.09475, 2023.
12. C. Kaymakci, S. Wenninger, and A. Sauer, "Energy anomaly detection in industrial applications with long short-term memory-based autoencoders," Procedia CIRP, vol. 104, pp. 182–187, 2021.
13. H. Torabi, S. L. Mirtaheri, and S. Greco, "Practical autoencoder based anomaly detection by using vector reconstruction error," Cybersecurity, vol. 6, no. 1, p. 1, 2023.
14. S. Zhou, Q. Chen, and X. Wang, "Active deep learning method for semi-supervised sentiment classification," Neurocomputing, vol. 120, pp. 536–546, 2013.
15. S. Saxena, "Beijing multi-site airquality data set." Kaggle Dataset, 2019.
16. Y. Liang, S. Ke, and J. Zhang, "Temporal-spatial interpolation for air pollution data," Environmental Modelling Software, vol. 124, p. 104587, 2020.
17. A. P. Bradley, "The use of the area under the roc curve in the evaluation of machine learning algorithms," Pattern Recognition, vol. 30, no. 7, pp. 1145–1159, 1997.
18. D. M. Powers, "Evaluation: From precision, recall and f-measure to roc, informedness, markedness and correlation," Journal of Machine Learning Technologies, vol. 2, no. 1, pp. 37–63, 2020.
19. C. J. Van Rijsbergen, Information Retrieval. Butterworth-Heinemann, 1979. Defines F1 as harmonic mean of precision/recall.
20. G. Naidu, T. Zuva, and E. M. Sibanda, "A review of evaluation metrics in machine learning algorithms," in Computer science on-line conference, pp. 15–25, Springer, 2023.
21. N. Tabassum et al., "Prediction of cloud ranking in a hyperconverged cloud ecosystem using machine learning," Computational Materials and Continua, vol. 67, no. 3, pp. 3129–3141, 2021.
22. J. Audibert, P. Michiardi, F. Guyard, S. Marti, and M. A. Zuluaga, "Usad: Unsupervised anomaly detection on multivariate time series," in
23. Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining, pp. 3395–3404, 2020.
24. M. I. Tariq, N. A. Mian, A. Sohail, T. Alyas, and R. Ahmad, "Evaluation of the challenges in the internet of medical things with multicriteria decision making (ahp and topsis) to overcome its obstruction under fuzzy environment," Mobile Information Systems, vol. 2020, 2020. Article ID not provided.
25. D. Nguyen, T. Nguyen, and S. Nahavandi, "Impact of feature scaling on deep learning for time-series forecasting,"

IEEE Access, vol. 8, pp. 12322–12333, 2020.

26. M. Vafaeipour, O. Rahbari, M. A. Rosen, F. Fazelpour, and P. Ansarirad, “Application of sliding window technique for prediction of wind velocity time series,” International Journal of Energy and Environmental Engineering, vol. 5, pp. 1–7, 2014.
27. Z. Zhang, “Improved adam optimizer for deep neural networks,” in 2018 IEEE/ACM 26th international symposium on quality of service (IWQoS), pp. 1–2, Ieee, 2018.
28. M. S. Error, “Mean squared error,” MA: Springer US, vol. 5, 2010.
29. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., “Scikit-learn: Machine learning in python,” the Journal of machine Learning research, vol. 12, pp. 2825–2830, 2011.
30. R. Carmona-Cabezas, J. Go´mez-Go´mez, E. Gutie´rrez de Rave´, and F. J. Jime´nez-Hornero, “A sliding window-based algorithm for faster transformation of time series into complex networks,” Chaos: An Interdisciplinary Journal of Nonlinear Science, vol. 29, no. 10, 2019.
31. H. H. Naqvi, T. Alyas, N. Tabassum, U. Farooq, A. Namoun, and S. A. M. Naqvi, “Comparative analysis: Intrusion detection in multi-cloud environment to identify way forward,” International Journal of Advanced Trends in Computer Science and Engineering, vol. 10, no. 3,pp. 2533–2539, 2021.
32. M. Munir, S. A. Siddiqui, A. Dengel, and S. Ahmed, “Fusead: Unsupervised anomaly detection in streaming sensors,” IEEE Access, vol. 7,pp. 22932–22949, 2019.