Anomaly Detection of Streamflow Time Series using LSTM Autoencoder

Arathy Nair G R*, Adarsh S

Department of Civil Engineering, TKM College of Engineering, APJ Kerala Technological University, India
*Corresponding author
doi: https://doi.org/10.21467/proceedings.156.16

ABSTRACT

Streamflow data obtained from the stream-gauge stations usually comprises of an ample volume of outliers. Anomaly detection is a requisite step in streamflow monitoring and analysis, especially in the context of water resources management, planning and flood risk studies. This study suggests a hybrid deep-learning anomaly detection method that combines an autoencoder and a long-short-term memory (LSTM) network. Multiple LSTM cells that collaborate with one another to understand the long-term dependencies of the data in a time series sequence make up the LSTM network. Based on the reconstruction error of the autoencoder's decoding phase, anomaly identification is accomplished. The applicability of the proposed method is demonstrated by considering the streamflow data (from 1985 to 2015) of Thumpamon streamgauge station of Greater Pamba River basin, Kerala. The hybrid framework exhibits promising results after computing the accuracy, precision, recall and the F1-Scores values as 99.51%, 100%, 89.89% and 94.73% respectively.

Keywords: Anomaly Detection, Long-Short Memory Network, Autoencoder

1 Introduction

Streamflow data is an essential prerequisite for water resources planning and management studies. The streamflow data collected from the gauge stations typically involves a large number of abnormal values. It may be due to the improper instrument exposure or installment, errors occurred during the recording and errors that occurred in the processing stage. These outliers should be removed in order to ensure high quality streamflow data for any further analysis.

An Anomaly Detection is a step in data-mining that pinpoint observations that depart from a dataset's regular behavior. This is considered to be an essential process since anomalous data can indicate changes in the typical behavior of the data points [1]. The breadth of outlier detection on time series data has been the subject of a significant amount of earlier work.

Detecting anomalies based on automated techniques are targeted on automatically recognizing the unusual patterns that do not hold on to the anticipated behavior of the systems under consideration [2]. There are various studies based on statistical and machine learning algorithms in this regard [3]. Numerous studies rely on supervised learning technique where the labelled data is used for training the algorithm. Unsupervised machine learning algorithms contrastingly, have proven to be successful in various anomaly detection applications [4], [5].

One such unsupervised technique deals with the usage of autoencoders for outlier detection. Autoencoders are a form of deep neural networks that employ non-linear dimensionality reduction to learn representations of the data [6]. It is more efficacious to train several layers with an autoencoder and is convenient when the data problems are complex and non-linear in nature. Autoencoders basically comprises of two different phases: encoding phase used to lessen the dimensionality of the input data and the decoding phase aims at reconstructing the data back by reducing the reconstruction error, which measures the difference between the original data and its reconstruction [7].

In this article, we propose a hybrid deep learning model for detecting anomalous observations in the

Proceedings DOI: 10.21467/proceedings.156; Series: AIJR Proceedings; ISSN: 2582-3922; ISBN: 978-81-961472-7-3

streamflow dataset based on the idea of long-term dependencies that occur in data samples. This model combines the capabilities of long short-term memory (LSTM) and Autoencoder (AE). The result of the hybrid framework is compared with that of the simple autoencoder technique. As far as we know, no earlier research has utilized the LSTM Autoencoder technology for the anomaly identification of streamflow data.

2 Materials and Methods

2.1 Datasets

The study was carried out using the streamflow data of Thumpamon streamgauge station of Greater Pamba River basin, Kerala, spanning within a time period of 30 years (1985-2015) (collected from WRIS), to detect the anomalies present in the selected dataset. The streamflow data (in m^3/s) obtained for 25 years from the stream gauge station is plotted as in Figure 1.

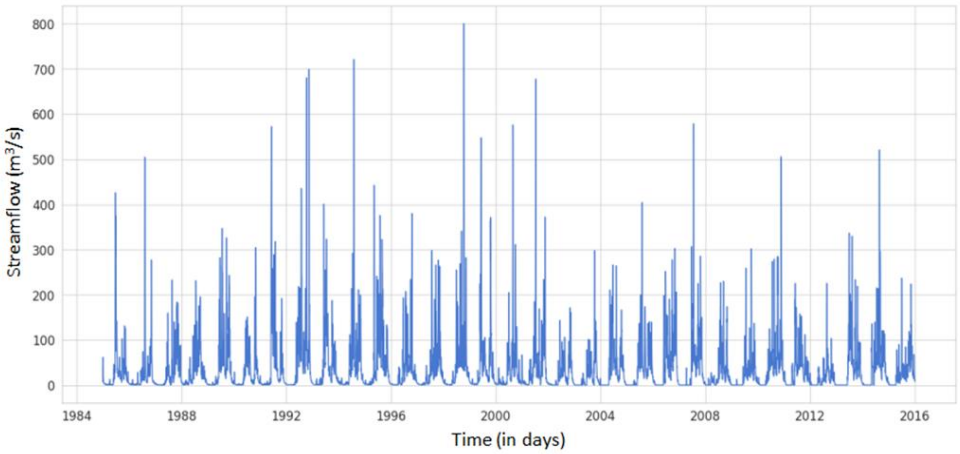

Figure 1: *Streamflow time series of Thumpamon for the period of 1985-2015*

2.2 Network Architecture

The autoencoder architecture involves an encoder phase which converts the input data into its compressed form and a decoder phase, where the reconstruction of the actual data is done. The reconstruction inaccuracy of the decoding step serves as the foundation for anomaly detection. The proposed hybrid deep learning model combines the LSTM network with the autoencoder for this purpose.

2.2.1 LSTM

LSTM networks can be described as a variant of Recurrent Neural Network (RNN) which exhibits the ability to recall the long-term dependencies within the input data. A basic LSTM network, depicted in Figure 2, is comprised of a cell and input, output and forget gates. The three gates control the flow of information into and out of the cell, and the cell remembers the values of the data points passed for all necessary time steps.

$$k_t = \sigma(P_k x_t + M_k h_t - 1 + n_f) \quad (1)$$

$$i_t = \sigma(P_i xt + I - 1 + n_i) \quad (2)$$

$$C_t = \tanh(P_c x_t + M_c h_t - 1 + n_c) \quad (3)$$

$$c_t = k_t * c_t - 1 + it * Ct \quad (4)$$

$$0_t = \sigma(P_o x_t + I_t - 1 + n_0) \quad (5)$$

$$h_t = 0_t * \tanh(c_t) \quad (6)$$

The weights of the input entering various gates are P and M. The gates are: input gate (it), input modulate gate (Ct), forget gate (kt), and output gate (0t). n is bias vectors, ct is cell state, and ht is hidden state. Each of these controllers controls the amount of information that is sent to the following state and the amount

that is received from the preceding loop.

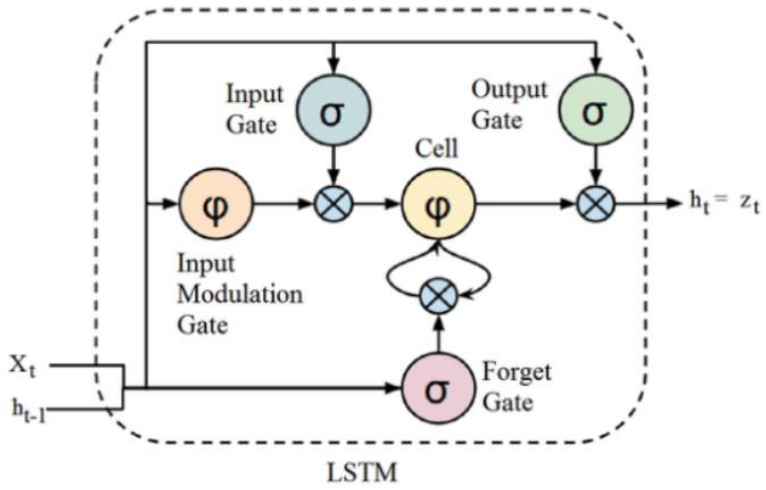

Figure 2: *Architecture of Long-Short Term memory Network*

2.2.2 Autoencoder

To learn effective coding of unlabeled data, an autoencoder is employed. By teaching the neural network to exclude extraneous data (often referred to as "noise"), it learns a representation for an input dataset. An input layer, an output layer, and multiple hidden layers make up a conventional autoencoder. The autoencoder working can be represented as in Figure 3.

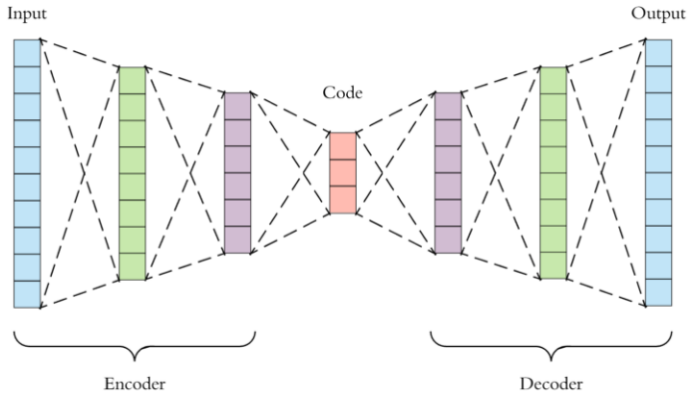

Figure 3: *Architecture of Autoencoder*

The encoding phase of the autoencoder results in the systematic reduction of complexity of the input data by multiple layers of a neural network. After this dimensionality reduction, the latter step involves the decoding or reconstruction phase. The decoding phase is just the reversal of the stage before. The compressed representation of the actual data is reconstructed back with a similar network structure. The optimal output from the autoencoder is a nearby depiction of the actual input. The anomalies present in the actual data can be determined from the reconstruction loss, which is computed using the deviation of reconstructed values from the original data.

2.3 Simple Autoencoder

The first stage of this model deals with the creation of a single fully-connected neural layer as encoder and as decoder phase and compile the model with Optimizer, Loss and Evaluation Metrics. The loss function used here depends on the mean-squared error between the output and the input, which is termed as the

Proceedings DOI: 10.21467/proceedings.156
ISBN: 978-81-961472-7-3

Series: AIJR Proceedings
ISSN: 2582-3922

Reconstruction Loss. It penalizes the network for reconstructing the series different from the input. After that, the model has to be fit with the test data.

2.4 LSTM - Autoencoder

The LSTM-Autoencoder that has been proposed makes use of the potentialities of both the LSTM and the Autoencoder, which assembles LSTM networks on the encoder and decoder stages. The high-dimensional input data sequence is obtained by the encoder as a fixed-size vector. The data handled by the former phase maintains dependencies between various data points in a time-series sequence while diminishing the high dimensional input vector depiction into low dimensional reduction using the memory cells of LSTM. Reconstruction error rates are used to define a threshold by the decoder LSTM in order to obtain the fixed-size input sequence within the compressed delineation of the input data. The actual dataset's abnormalities are found using this threshold.

3 Results and Discussion

The effectiveness of the simple and hybrid models for streamflow anomaly identification was examined in this section. The datasets of streamflow collected for a period of 30 years is divided into training phase (70%) and testing phase (30%), by considering the previous literatures.

The trends of the loss (mean squared error) at various intervals obtained for the hybrid model are shown in Figure 4 below. The training loss measures the error rate of the model during training. From the figure, the training loss is found to be stabilized after around 3 epochs. The validation loss shows least value at the third epoch. This fits nicely, and our suggested model performs admirably.

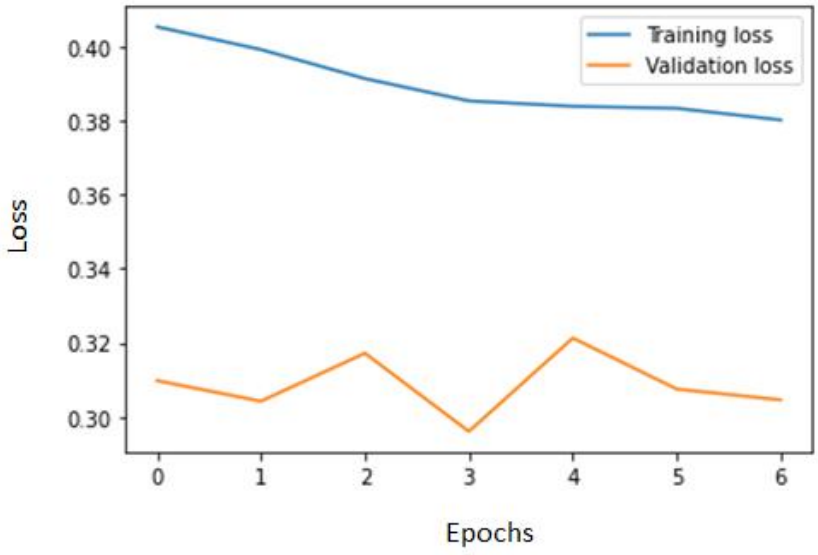

Figure 4: *Training and validation loss*

The next step deals with the detection of threshold value for identifying the anomalies present in the dataset. The reconstruction loss—the distinction between the original and rebuilt time series data—is used to determine it.The Mean Absolute Error (MAE) loss of the training dataset is determined and the maximum value among them is taken as the reconstruction error threshold. Each data point's recalculated loss is compared to the reconstruction error threshold, and data points with values higher than this are regarded as anomalies. The reconstruction error threshold obtained for the considered dataset in the hybrid LSTM-Autoencoder is around 0.915. The Figure 5 depicts the Loss - Threshold curve in which the data points with reconstruction loss values greater than the threshold are considered as the anomalies.

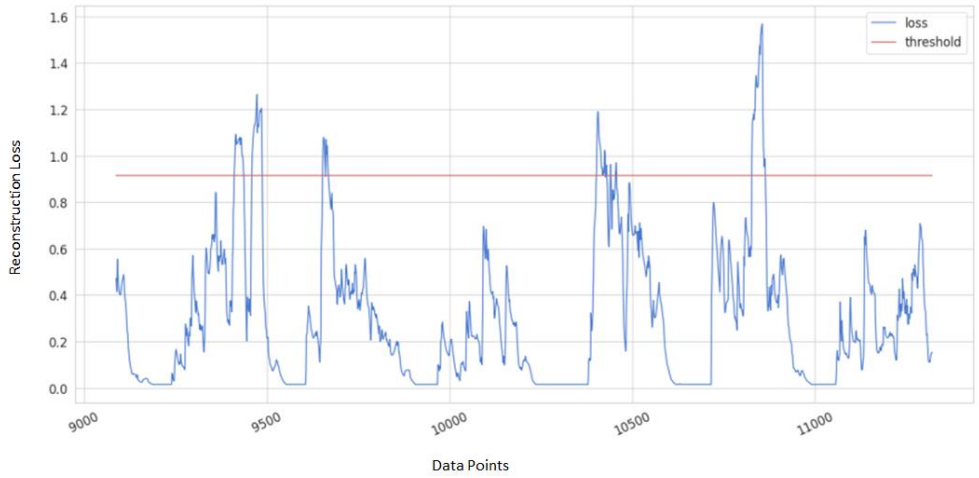

Figure 5: *Reconstruction Loss vs Threshold Curve*

The threshold value determined from the simple autoencoder is less than that obtained from the hybrid model (Threshold – 0.7). The anomaly points are determined afterwards by considering the threshold values. The total number of test samples = 3287 (Daily data from Januray 1, 2007 to December 31, 2015). The most deviated values are determined using binary segmentation multiple change point analysis using Rstudio, to be used as the base data for accuracy analysis of the autoencoder models. The change point analysis indicates 156 data points as anomalies and remaining 3131 points as normal samples. The simple autoencoder model established 108 data points and the hybrid model exhibits 140 points as anomalies. Among the 3131 standard data representations, both models accurately detected all 3131 normal data points (100% of the time). The Figure 6 and 7 depicts the obtained anomaly data points for simple and hybrid autoencoder models respectively.

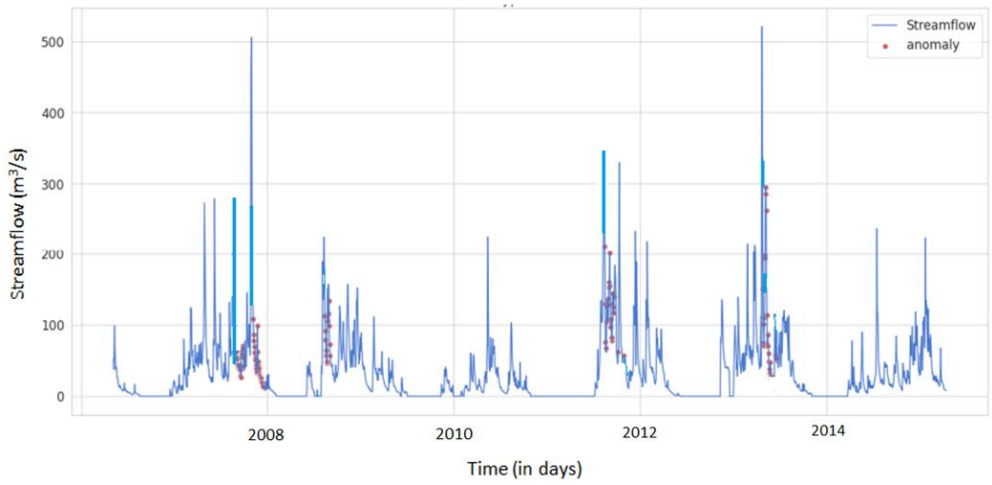

Figure 6: *Anomaly data points obtained from Simple Autoencoder*

Proceedings DOI: 10.21467/proceedings.156
ISBN: 978-81-961472-7-3

Series: AIJR Proceedings
ISSN: 2582-3922

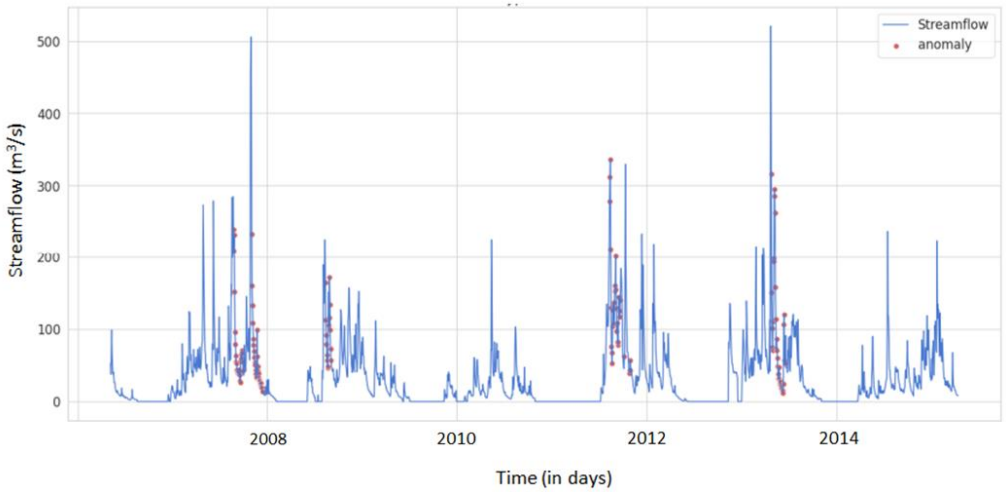

Figure 7: *Anomaly data points obtained from Hybrid LSTM Autoencoder*

The Figure 8 represents the confusion matrix obtained for simple and hybrid autoencoder models. The accuracy, recall, precision and F1- scores of simple and hybrid models are determined and are represented using a radar plot as in Figure 9. The accuracy obtained for simple and hybrid models are 98.5% and 99.52% respectively. The precision values obtained for both the models reach 100%. The recall and F1-Scores obtained are 89.89% , 69.23% and 94.73%, 81.8% respectively for hybrid and simple autoencoders. The evaluation's overall findings demonstrate that the suggested model outperforms the basic autoencoder model in terms of accuracy in detecting abnormalities. On the basis of the entire time series testing dataset, this conclusion was reached.

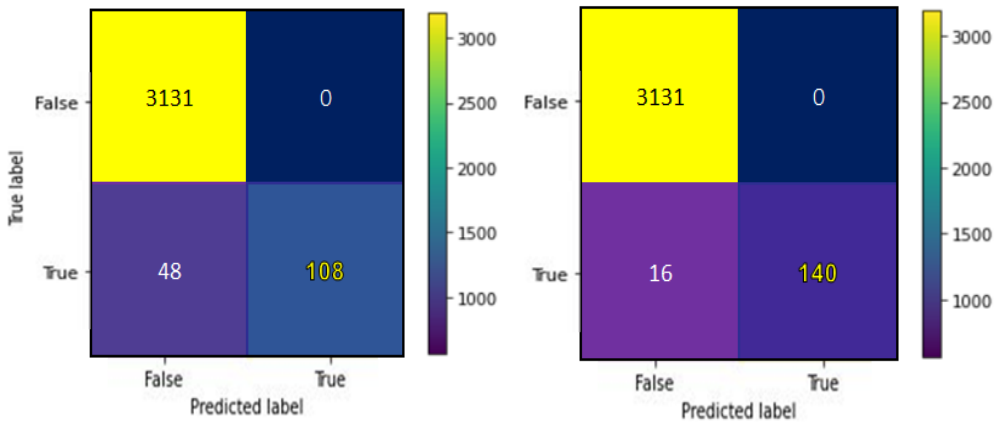

Figure 8: *Confusion Matrix obtained for Simple and Hybrid LSTM Autoencoder*

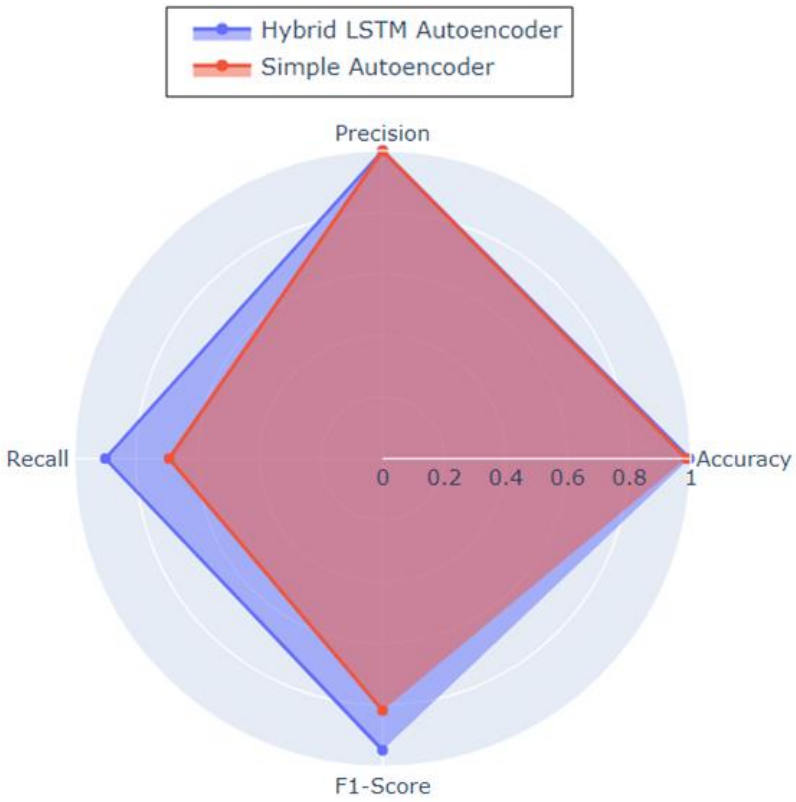

Figure 9: *Radar Plot for Model Performance Evaluation*

4 Conclusions

This paper focuses on anomaly detection of streamflow dataset, which experiences only a few researches in the past. In this study, a hybrid LSTM Autoencoder model is established for streamflow anomaly detection. In the hybrid model, two separate LSTM networks—each of which has multiple LSTM units—are taken into account as the encoder and decoder and have the capacity to recognise long-term correlational relationships that are present in a time series sequence. In addition to maintaining the long-term dependencies established by the LSTM encoder and providing outputs to match the input through the LSTM decoder, autoencoder is regarded to generate encoded features of the input series representation. The trained model's maximum reconstruction loss is used as a threshold and is passed into the anomaly detector. Each data sample from the testing set is flagged as an anomaly by the anomaly detector if the reformation loss result is higher than the selected limit. The model evaluation results represent that the hybrid model is efficient for anomaly detection than the simple autoencoders. The further scope of the study deals with the inclusion of stacked or bidirectional LSTM for autoencoder working.

5 Declarations

5.1 Competing Interests

The authors confirm that they have no known financial or interpersonal conflicts that would have appeared to have an impact on the research presented in the study.

5.2 Publisher's Note

AIJR remains neutral with regard to jurisdictional claims in institutional affiliations.

How to Cite

Arathy & Adarsh (2023). Anomaly Detection of Streamflow Time Series using LSTM Autoencoder. *AIJR Proceedings*, 112-119. https://doi.org/10.21467/proceedings.156.16

References

[1] L. Kulanuwat *et al.*, "Anomaly Detection Using a Sliding Window Technique and Data Imputation with Machine Learning for Hydrological Time Series," *Water* , vol. 13, no. 13, p. 1862, Jul. 2021, doi: 10.3390/W13131862.

[2] M. Gupta, J. Gao, C. C. Aggarwal, and J. Han, "Outlier Detection for Temporal Data: A Survey," *IEEE Trans Knowl Data Eng*, vol. 26, no. 9, pp. 2250–2267, Sep. 2014, doi: 10.1109/TKDE.2013.184.

[3] D. J. Hill and B. S. Minsker, "Anomaly detection in streaming environmental sensor data: A data-driven modeling approach," *Environmental Modelling & Software*, vol. 25, no. 9, pp. 1014–1022, Sep. 2010, doi: 10.1016/J.ENVSOFT.2009.08.010.

[4] Y. Qin and Y. Lou, "Hydrological time series anomaly pattern detection based on isolation forest," in *Proceedings of 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference, ITNEC 2019*, Institute of Electrical and Electronics Engineers Inc., 2019, pp. 1706–1710. doi: 10.1109/ITNEC.2019.8729405.

[5] M. Safaei *et al.*, "A Systematic Literature Review on Outlier Detection in Wireless Sensor Networks," *Symmetry (Basel)*, vol. 12, no. 3, p. 328, Feb. 2020, doi: 10.3390/SYM12030328.

[6] B. Hwang and S. Cho, "Characteristics of autoassociative MLP as a novelty detector," in *Proceedings of the International Joint Conference on Neural Networks*, IEEE, 1999, pp. 3086–3091. doi: 10.1109/IJCNN.1999.836051.

[7] S. Russo, A. Disch, F. Blumensaat, and K. Villez, "Anomaly Detection using Deep Autoencoders for in-situ Wastewater Systems Monitoring Data," *Electrical Engineering and Systems Science*, Feb. 2020, Accessed: Mar. 23, 2023. [Online]. Available: https://arxiv.org/abs/2002.03843v3