2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)
October 6-10, 2024. Sarawak, Malaysia

Water Quality Anomaly Detection with Dual Sliding Windows and Convolutional LSTM

Jing Bi[1], Ming Yuan[1], Haitao Yuan[2], Ziqi Wang[1] and Junfei Qiao[1]

[1]College of Computer, Beijing University of Technology, Beijing 100124, China
[2]School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China

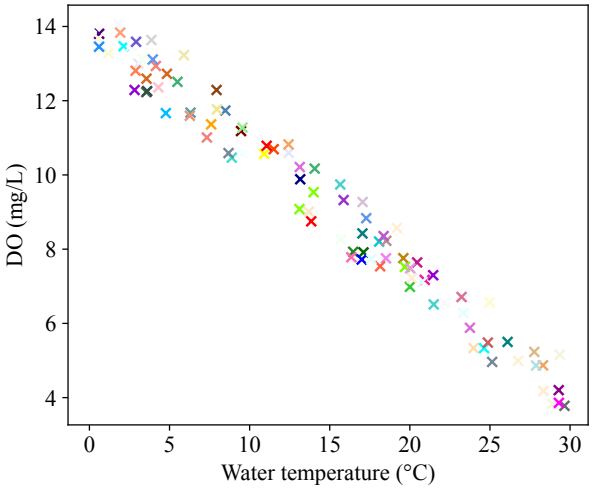
2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC) | 978-1-6654-1020-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/SMC54092.2024.10831202

***Abstract*—Water pollution is continuously increasing in water ecosystems across all continents. Surface water sensors can record data on water quality indicators at regular intervals, and the associated water quality sequences show abnormal trends when extreme weather or unusual industrial discharges occur. Therefore, governments can take timely actions to minimize damage and protect the water environment by detecting these abnormal trends promptly. However, current methods make it difficult to interpret different correlations among water quality parameters effectively. To solve this problem, this work proposes a parameter correlation-aware anomaly detection model, which integrates Dual sliding windows, Convolutional LSTM, and a Deep neural network with dropout, called for DCLD short. First, DCLD designs dual sliding windows to capture local and global patterns within the sequence of water quality. Second, DCLD adopts a stacked long short-term memory with a convolutional neural network to capture complex features and long-term dependencies in the time series. Third, DCLD uses a deep neural network incorporating the dropout algorithm to extract abstract features. DCLD mitigates overfitting risks and enhances the model's generalization capacity. Finally, DCLD is evaluated with two real-world water quality datasets, and its anomaly detection accuracy is improved by 5.41% and 0.79% on average over its peers.**

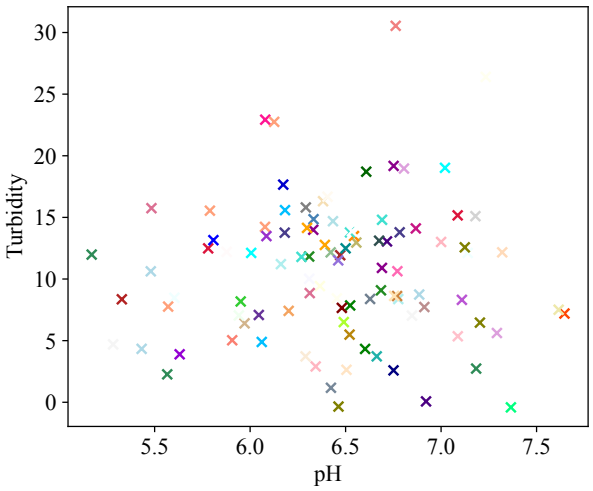
***Index Terms*—Water quality, anomaly detection, parameter correlations, LSTM, convolutional neural networks.**

I. Introduction

Water pollution has become one of the factors restricting social development. In water quality protection, many monitoring sensors are deployed in water environments worldwide, which monitor and upload water quality data in real-time. Those sensors have accumulated data about water quality over the years [1]. Moreover, detecting anomalies in the water quality time series is essential to provide timely actions for water quality data warnings and ultimately minimize losses incurred by water quality anomalies. However, several water quality indicators, including dissolved oxygen (DO), nitrogen content, *etc*., can impact the water quality data. As shown in Fig. 1(a), there is a strong correlation between DO and temperature, and in Fig. 1(b), the correlation between the Potential of Hydrogen (pH) and turbidity is weak. In addition, different water quality indicators may also have different degrees of correlation with each other. Some indicators may be highly correlated, while others may have low or no correlation. In that case, such differences may lead to challenges during anomaly detection. Therefore, choosing the best model to analyze the water quality series with different degrees of correlation is highly challenging.



(a) Correlation between DO and temperature.



(b) Correlation between pH and turbidity.

Fig. 1. Correlation between different indicators.

To solve the above problems, researchers have proposed several methods for detecting anomalies in the time series.

This work was supported by the National Natural Science Foundation of China under Grants 62073005 and 62173013, the Beijing Natural Science Foundation under Grants 4232049 and L233005, and the Fundamental Research Funds for the Central Universities under Grant YWF-23-03-QB-015.

Traditional methods comprise four types: 1) statistical methods, *e.g.*, seasonal autoregressive integrated moving average, and seasonal decomposition; 2) distance-based algorithms, *e.g.*, K-nearest neighbor (KNN) [2] and local outlier factor (LOF) [3]; 3) clustering-based algorithms, *e.g.*, K-means algorithms [4]; and 4) machine learning-based algorithms, *e.g.*, support vector machine [5] and random forests [6]. These methods are suitable in cases where the amount of data is small, and the problem is relatively simple. However, different degrees of correlation between different water quality parameters complicate the problem of water quality anomaly detection. In addition, deep learning methods possess strong learning capabilities, enabling automatic acquisition of high-level feature representations from the raw data. They excel in capturing intricate data patterns and relationships, including nonlinear and time-varying relationships. Additionally, they exhibit superior adaptability to large-scale datasets. Therefore, the primary focus of time series anomaly detection research is progressively transitioning from conventional statistical machine learning approaches to more robust deep learning techniques.

Two common models in deep learning, including recurrent neural networks (RNNs) and convolutional neural networks (CNNs), are employed in identifying anomalies within water quality sequences and yielding promising outcomes. RNNs [7] can effectively capture time-series features and patterns of changes within water quality data, facilitating anomaly detection. Moreover, researchers have proposed a special variant of long short-term memory (LSTM) [8], [9] on this basis, which can better capture long-term dependencies. CNNs [10]–[12] are mainly adopted in two-dimensional images. However, they can also be used for localized feature extraction of one-dimensional water quality data, such as sudden water quality changes or unusual fluctuations. These deep learning models combine the characteristics of time-series data to capture temporal information and anomalous patterns better. However, RNN may not be able to handle long-term dependencies well, and CNN also localizes the problem of sensory field limitations. In this case, some researchers have attempted to combine these two models to deal with the time series anomaly detection problems. For example, Antonius *et al*. [13] propose a convolutional long short-term memory network (C-LSTM) to combine the advantages of CNN and LSTM and achieve good results in anomaly detection. However, this method is insufficiently flexible to address the constantly changing and complex real-world environments. In recent years, transformers [14] have been widely applied to anomaly detection due to their excellent ability to capture global information. Chen *et al*. [15] introduce graph neural networks to address anomaly detection problems. However, these models are too complex and time-consuming [16], failing to meet the timeliness requirements for the water quality anomaly detection.

To detect water quality anomalies more scientifically, this work proposes a new anomaly detection model that combines Dual sliding windows, Convolutional LSTM, and a Deep neural network with dropout, called for DCLD short. Experiments with two real-world water quality datasets demonstrate that DCLD outperforms state-of-the-art anomaly detection models. The major contributions of our work are two-fold:

1) DCLD integrates a dual sliding window analyzer, which employs two sliding windows for data processing and analysis. Primary and secondary windows are adopted from a cascade structure to extract multi-scale temporal features and improve feature richness.
2) DCLD combines DNN with the dropout mechanism, CNN, and stacked LSTM models. It utilizes different correlations of water quality parameters, improving the model's sensitivity and anomaly detection accuracy.

II. Model Framework

This section describes our proposed DCLD. First, the problem of water quality anomaly detection is introduced. Each module and the architecture of DCLD are explained.

A. Problem Definition

The purpose of detecting anomalies in water quality sequences is to assess if the current water quality is susceptible to contamination, leveraging past water quality samples collected from river sensors. X denotes a water quality time series and $X=\{x_1, x_2, \ldots, x_t\}$, where x_t represents the water quality data at time point t. Each point encompasses two factors, including DO and total phosphorus (TP). A function $f(x_t)$ represents the state of water quality at time point t, which maps x_t to a state of *normal* or *anomaly*, *i.e.*, $f(x_t) \rightarrow \{\text{normal}, \text{anomaly}\}$.

B. Dual Sliding Window

As shown in Fig. 2, DCLD comprises three primary components: dual sliding windows, C-LSTM, and DNN. The structure of the dual sliding window module is shown in Fig. 3. Specifically, two sliding windows are utilized. The primary and the secondary windows are adopted to process the data simultaneously, thereby converting the anomaly detection into a classification problem.

In the primary window, the size of the sliding window is denoted by W_1. A zero padding strategy is employed to overcome the issue of insufficient data at the beginning of the series due to window size limitations, padding the initial part of the series with W_1-1 zero values. The extended series can be represented as X' and $X'=\{0, 0, \ldots, 0, x_1, x_2, \ldots, x_n\}$. X' contains W_1-1 zero values to ensure that each sliding window can acquire sufficient data points for processing. Thus, for each position t' ($t' \in [1, 2, \ldots, n]$) in the extended series X', the sliding window encompasses data points from $x'_{t'}$ to $x'_{t'+W_1-1}$, where x'_t denotes the data point t in X'. Therefore, the subsequence $X_{t'}^{(W_1)}$ corresponding to each sliding window can be defined as:

$$X_{t'}^{(W_1)}=\{x'_{t'}, x'_{t'+1}, \ldots, x'_{t'+W_1-1}\} \tag{1}$$

Then, all possible subsequences can be extracted sequentially to form an ordered sequence $S^{(W_1)}$, *i.e.*,

$$S^{(W_1)}=[X_1^{(W_1)}, X_2^{(W_1)}, \ldots, X_n^{(W_1)}] \tag{2}$$

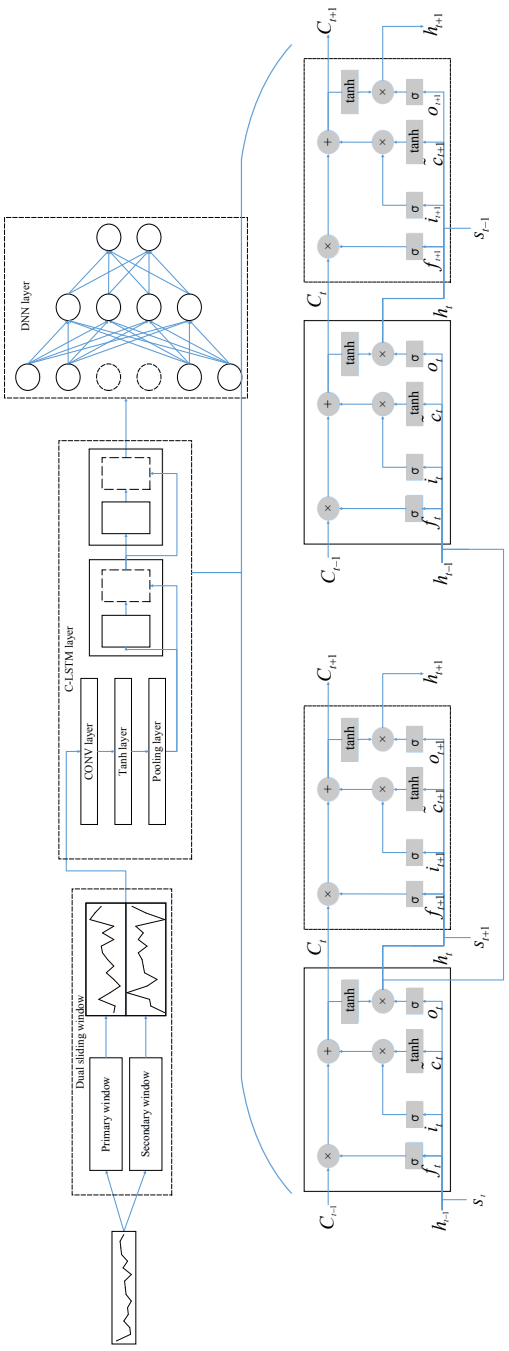


Fig. 2. Structure of DCLD.

For each data point in $S^{(W_1)}$, mean processing results in a new sequence is performed, and it is defined as:

$$\bar{S}^{(W_1)}=[\bar{X}_1^{(W_1)}, \bar{X}_2^{(W_1)}, \ldots, \bar{X}_n^{(W_1)}] \tag{3}$$

Following the same principle, a secondary window of size W_2 is constructed, and $W_2<W_1$. Thus, the sequences are shown as:

$$\begin{aligned} X_{t''}^{(W_2)}&=\{x''_{t''}, x''_{t''+1}, \ldots, x''_{t''+W_2-1}\} \\ S^{(W_2)}&=[X_1^{(W_2)}, X_2^{(W_2)}, \ldots, X_n^{(W_2)}] \\ \bar{S}^{(W_2)}&=[\bar{X}_1^{(W_2)}, \bar{X}_2^{(W_2)}, \ldots, \bar{X}_n^{(W_2)}] \end{aligned} \tag{4}$$

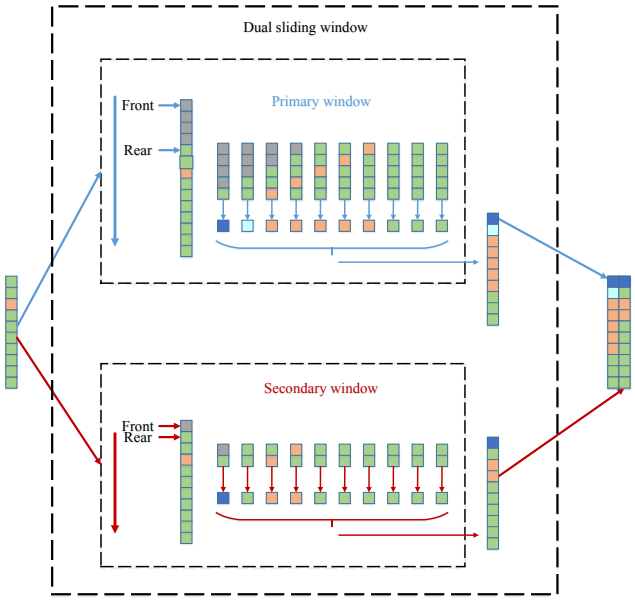


Fig. 3. Structure of the dual sliding window module.

By concatenating $\bar{S}^{(W_1)}$ and $\bar{S}^{(W_2)}$, a new sequence X_c is obtained. Using a dual-stage sliding window, the one-dimensional time series data is transformed into a two-dimensional dataset, facilitating the extraction of temporal features across two dimensions.

C. C-LSTM Layer

As shown in Fig. 2, the C-LSTM layer consists of a CNN network and two stacked LSTM layers to help the model learn water quality features. X_c is input to the CNN, which undergoes the convolution operation, the activation function, and the pooling operation to obtain the pooled output feature map (Y_{p}), *i.e.*,

$$Y_{\mathrm{p}}=\mathrm{Pool}(\mathrm{Tanh}(X_c*\Theta)) \tag{5}$$

where Θ is the convolutional kernel. $*$ represents the convolution operation, Tanh$(\cdot)$ is the hyperbolic tangent function, and Pool$(\cdot)$ is the pooling operation. Afterwards, Y_{p} is reshaped into X_l.

Each LSTM layer contains multiple LSTM cells, each of which is given as:

$$\begin{aligned} f_t&=\sigma(W_f\cdot[h_{t-1}, x_{lt}]+b_f) \\ i_t&=\sigma(W_i\cdot[h_{t-1}, x_{lt}]+b_i) \\ \tilde{C}_t&=\tanh(W_c\cdot[h_{t-1}, x_{lt}]+b_c) \\ C_t&=f_t\cdot C_{t-1}+i_t\cdot\tilde{C}_t \\ o_t&=\sigma(W_o\cdot[h_{t-1}, x_{lt}]+b_o) \\ h_t&=o_t\cdot\tanh(C_t) \end{aligned} \tag{6}$$

where x_{lt} represents the input at time step t. f_t, i_t, and o_t denote the forget, input, and output gate outputs at time step t, respectively. h_{t-1} and h_t represent the hidden states at time steps $t-1$ and t. $\tilde{C}_t$ represents the new information proposed to update the memory cell at time step t. C_t denotes the

content of the memory cell at time step t. W_i, W_f, W_o, and W_c are the weight matrices for the input gate, the forget gate, the output cell, and the candidate gate, respectively. b_i, b_f, b_o, and b_c are the bias vectors for the input gate, the forget gate, the output cell, and the candidate gate, respectively. Therefore, the output of the stacked LSTM is given as:

$$H=\text{LSTM}_2(\text{LSTM}_1(X_l)) \tag{7}$$

where LSTM_1 and LSTM_2 represent the first and the second LSTM layers in a stacked LSTM, respectively.

D. DNN layer

The dropout mechanism is utilized to drop inputs during the training process of the DNN randomly, *i.e.*,

$$H^{(l)}=\tanh\left(\text{dropout}\left(H^{(l-1)}\cdot W^{(l)}+b^{(l)},p\right)\right) \tag{8}$$

where $H^{(l)}$ represents the output of layer l. $b^{(l)}$ and $W^{(l)}$ are the bias vector and weight matrix of layer l, respectively, and p is the dropout rate. Finally, the output of DCLD, O, is obtained as:

$$\text{O}=\text{sigmoid}\left(H^{(L)}\cdot W^{(L+1)}+b^{(L+1)}\right) \tag{9}$$

where L refers to the number of hidden layers.

III. Experimental Evaluation

A. Experimental Setup

The DCLD and comparative models are executed using PYTORCH in a server with an Intel Xeon 6248R CPU and a GTX3090 GPU. Moreover, DCLD is evaluated with two real-world water quality datasets collected from stations in the Beijing-Tianjin-Hebei region, China. The first dataset collects 9,779 data from Apr. 2, 2014, to Sept. 17, 2018, at the Gubeikou site in the Beijing-Tianjin-Hebei region, China. Another collects 6,511 samples from Aug. 31, 2018, to Dec. 12, 2021, at the Wucun site. Data points in these datasets are gathered at four-hour intervals. Moreover, DO and TP are chosen as the detection indicators. The two datasets are illustrated in Table I. It is worth noting that the experimental datasets are real. Thus, the number of outliers is low, and there is a data imbalance issue. Four types of anomalous data are added and shown in Fig. 4. The ratio of the training, validation, and test sets is set to 7:1:2.

TABLE I
SUMMARY OF DATASETS

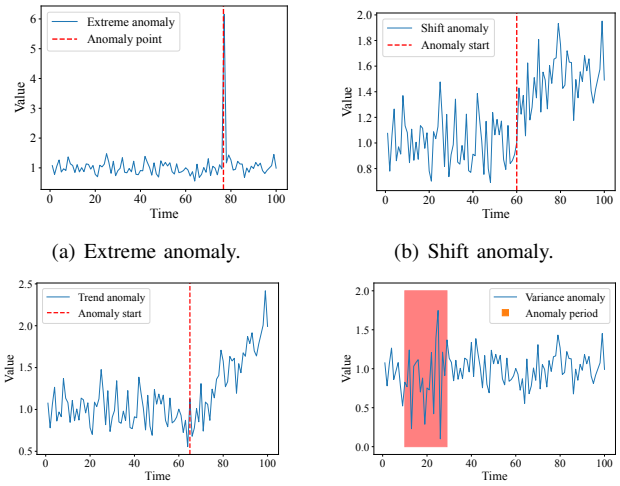
Parameter	Gubeikou	Wucun
Time Span	Apr. 2, 2014–Sept. 17, 2018	Aug. 31, 2018–Dec. 12, 2021
Time Interval	4 hours	4 hours
Indicator	DO	TP

B. Evaluation Metrics

Anomaly detection is treated as a classification task. Models are evaluated with metrics including Accuracy, Precision, Recall, and F1-score. Accuracy is the proportion of correctly identified samples by the model out of the total sample number. Precision quantifies the fraction of samples the model accurately classifies as positive out of those labeled as positive. Recall calculates the percentage of positive instances (true positives) correctly identified from all the actual positive samples in the dataset, and F1-score offers a balanced mean of Precision and Recall. They are calculated as:

$$\begin{aligned}\text{Accuracy}&=\frac{\text{TruePositive}+\text{TrueNegative}}{\text{TotalPopulation}}\\ \text{Precision}&=\frac{\text{TruePositive}}{\text{TruePositive}+\text{FalsePositive}}\\ \text{Recall}&=\frac{\text{TruePostive}}{\text{TruePositive}+\text{FalseNegative}}\\ \text{F1-score}&=2\times\frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}\end{aligned} \tag{10}$$

where TruePositive represents when the number of positive samples correctly identified. FalsePositive denotes the number of false negative samples. TrueNegative represents the number of negative samples that are correctly identified. FalseNegative denotes the number of underreported positive samples. TotalPopulation represents the total number of samples the model processes in the test set.



(a) Extreme anomaly. (b) Shift anomaly.

(c) Trend anomaly. (d) Variance anomaly.

Fig. 4. Four types of time series anomalies.

C. Hyperparameter setting in DCLD

DCLD has several important hyperparameters, including the window size of the dual sliding windows, the discard rate, and the number of DNN layers. Based on the experimental results in Figs. 5 and 6, the number of DNN layers is 2, the dropout discard rate is 0.3, the primary window size is 60, and the second window size is 10.

D. Comparison Experiments

To validate the detection performance of DCLD, DCLD is compared with several benchmark models, including KNN [2], Autoencoder [17], K-Means [4], LSTM [18], LOF [3], Random Forest [6], Transformer [2], and VAE [19]. It is shown in Table II that DCLD exhibits superior performance across most metrics, notably excelling in accuracy and F1-score on both datasets. This proves the remarkable efficiency of DCLD in anomaly detection. Fig. 7 shows the

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT MODELS

Methods	Gubeikou_DO				Wucun_TP			
	Accuracy	Precision	Recall	F1-score	Accuracy	Precision	Recall	F1-score
KNN	0.670	0.8167	0.2945	0.4329	0.749	0.8363	0.3970	0.5385
Autoencoder	0.776	0.7239	0.7725	0.7474	0.750	1.0000	0.3243	0.4897
K-Means	0.503	0.4411	0.6034	0.5096	0.262	0.2737	0.6071	0.3773
LSTM	0.565	0.4909	0.4180	0.4515	0.717	1.0000	0.2343	0.3797
LOF	0.669	0.9846	0.2308	0.3739	0.689	0.7846	0.2143	0.3366
Random Forest	0.669	0.6252	0.5673	0.5948	0.683	0.8587	0.1660	0.2782
Transformer	0.534	0.4578	0.4743	0.4659	0.736	1.0000	0.2845	0.5382
VAE	0.531	0.4661	0.6431	0.5405	0.717	1.0000	0.2343	0.3797
DCLD	0.814	0.7694	0.8072	**0.7878**	0.768	1.0000	0.3724	**0.5427**

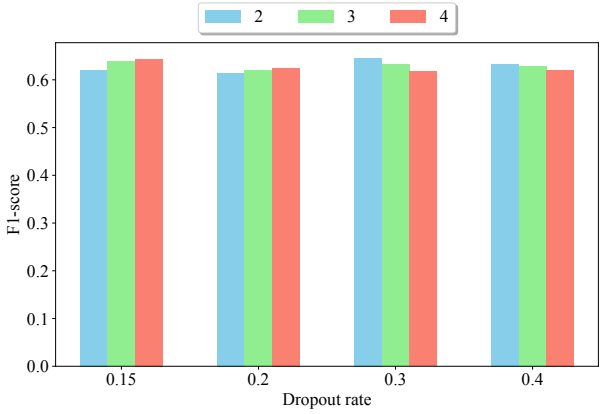


Fig. 5. F1-score for different layer numbers with different discard rates.

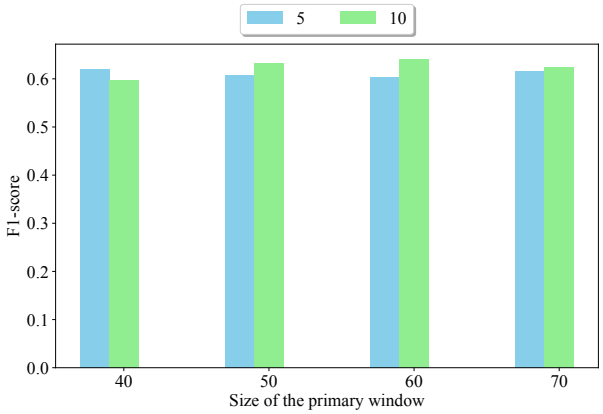
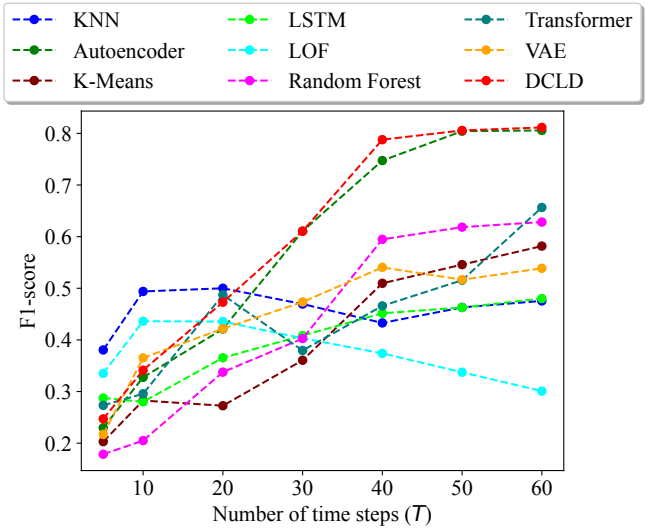
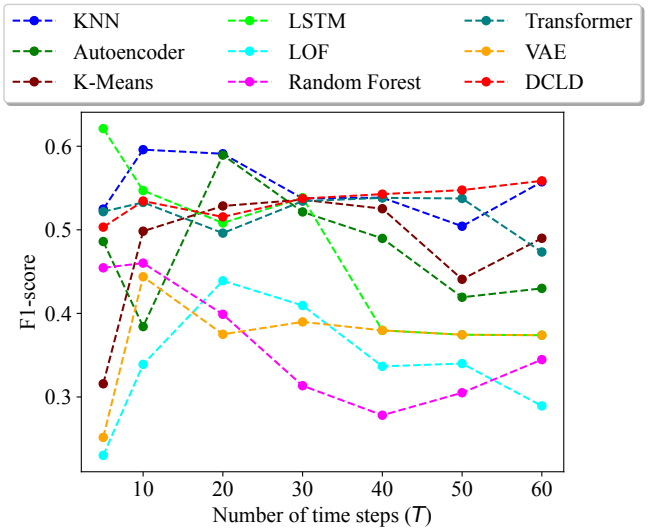


Fig. 6. F1-score for different window sizes.

trend of the F1-score for different models as a function of the time step size. Fig. 7(a) presents the performance of different models with the Gubeikou dataset. It is shown that traditional machine learning algorithms such as KNN and LOF perform well in the short-term time series. There is no significant gap between traditional anomaly detection and deep learning models. However, deep learning methods achieve a significant advantage in the medium to long-term time series. Among them, the autoencoder is already



(a) F1-score with respect to T for the GubeikouDO dataset.



(b) F1-score with respect to T for the WucunTP dataset.

Fig. 7. Performance comparison with respect to different T.

capable of feature extraction, but DCLD is more effective in handling the temporal dependencies of the time series data. The dual sliding windows capture data features at different temporal scales, providing contextual information. Moreover, combining CNN and stacked LSTM allows the model to effectively process these features and retain critical information over the long-term time series. Furthermore, the dropout helps prevent the model from overfitting to specific time steps, ensuring generalization across the entire time series. The above characteristics enable DCLD to capture more contextual information and maintain flexibility in adapting to new data, leading to performance improvement as the time step increases. However, comparative models do not have this design to continuously enhance their modeling capabilities for sequence data, leading to less significant performance improvements in long-sequence prediction than DCLD.

Similarly, the performance of different models with the Wucun dataset is presented in Fig. 7(b). It is shown that DCLD performs the best at most time steps. However, it experiences a slight decrease in performance in 10 to 20 steps. This is because DCLD encounters outliers or noise at specific sequence lengths, affecting its performance. Compared to Fig. 7(a), the F1-score of the compared models exhibits greater fluctuations, indicating that the WucunTP dataset contains more complex anomaly patterns, affecting the performance of all models. Given both datasets, DCLD demonstrates a trend of performance improvement with increasing time step size, showing its ability to learn better and adjust its anomaly detection mechanism with more data.

IV. Conclusions

Accurate water quality anomaly detection is imperative to protect the water environment and reduce water pollution hazards. The goal of water quality anomaly detection is to determine whether there is a risk of contamination in the current water quality based on water quality samples. Water quality parameters with different levels of correlation in complex water environments bring a main challenge. In this work, a water quality anomaly detection model is proposed that integrates Dual sliding windows, Convolutional LSTM, and a Deep neural network with dropout, abbreviated as DCLD. A double sliding window is designed to initially capture different ranges in the series. Moreover, a C-LSTM module is proposed to handle the long-term dependencies of the water quality series, effectively utilizing the features of the water quality series. Furthermore, abstract features are further extracted with a DNN network to improve the generalization ability and detection performance. Compared to eight baseline models with two real-world datasets, experimental results show that DCLD's accuracy is improved by 5.41% and 0.79% on average with the two datasets, respectively.

In the future, first, we plan to investigate the cooperative relationship between the double sliding window and neural networks to improve feature utilization and better coping with complex water environments. Second, we intend to investigate relationships in CNNs and LSTM to adaptively select the most important features and better deal with sequences of different relevance.

References

[1] J. Qiao, Y. Lin, J. Bi, H. Yuan, G. Wang, and M. Zhou, "Attention-Based Spatiotemporal Graph Fusion Convolution Networks for Water Quality Prediction," *IEEE Transactions on Automation Science and Engineering*, doi: 10.1109/TASE.2023.3285253.

[2] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, "Efficient KNN Classification with Different Numbers of Nearest Neighbors," *IEEE Transactions on Neural Networks and Learning systems*, vol. 29, no. 5, pp. 1774–1785, May 2018.

[3] L. Chen, W. Wang, and Y. Yang, "CELOF: Effective and Fast Memory Efficient Local Outlier Detection in High-Dimensional Data Streams," *Applied Soft Computing*, vol. 102, pp. 1–10, Jan. 2021.

[4] R. Ghezelbash, A. Maghsoudi, and E. Carranza, "Optimization of Geochemical Anomaly Detection Using A Novel Genetic K-Means Clustering (GKMC) Algorithm," *Computers & Geosciences*, vol. 134, pp. 1–11, Oct. 2020.

[5] Y. Liu, R. Zhang, Z. He, Q. Huang, R. Zhu, H. Li, and Q. Fu, "The Study of Hydraulic Machinery Condition Monitoring Based on Anomaly Detection and Fault Diagnosis," *Measurement*, vol. 230, pp. 1–12, Mar. 2024.

[6] H. Akouemo, and R. Povinelli, "Probabilistic Anomaly Detection in Natural Gas Time Series Data," *International Journal of Forecasting*, vol. 32, no. 3, pp. 948–956, Jun. 2016.

[7] Y. Liu, C. Gong, L. Yang, and Y. Chen, "DSTP-RNN: A Dual-Stage Two-Phase Attention-Based Recurrent Neural Network for Long-Term and Multivariate Time Series Prediction," *Expert Systems with Applications*, vol. 143, pp. 1–12, Apr. 2020.

[8] J. Bi, Y. Lin, Q. Dong, H. Yuan, and M. Zhou, "Large-Scale Water Quality Prediction with Integrated Deep Neural Network," *Information Sciences*, vol. 571, pp. 191–205, Sept. 2021.

[9] J. Bi, Z. Chen, H. Yuan, and J. Zhang, "Accurate Water Quality Prediction with Attention-Based Bidirectional LSTM and Encoder–Decoder," *Expert Systems with Applications*, vol. 238, pp. 1–10, Oct. 2023.

[10] W. Ullah, T. Hussain, F. Ullah, M. Lee, and S. Baik, "TransCNN: Hybrid CNN and Transformer Mechanism for Surveillance Anomaly Detection," *Engineering Applications of Artificial Intelligence*, vol. 123, pp. 1–11, Apr. 2023.

[11] F. Abir, M. Chowdhury, M. Tapotee, A. Mushtak, A. Khandakar, S. Mahmud, and M. Hasan, "PCovNet+: A CNN-VAE Anomaly Detection Framework with LSTM Embeddings for Smartwatch-Based COVID-19 Detection," *Engineering Applications of Artificial Intelligence*, vol 122, pp. 1–15, Jun. 2023.

[12] J. Bi, L. Xu, H. Yuan, and J. Zhang, "Web Traffic Anomaly Detection Using a Hybrid Spatio-Temporal Neural Network," *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Honolulu, Oahu, HI, USA, 2023, pp. 5009–5014.

[13] F. Antonius, J. Sekhar, V. Rao, R. Pradhan, S. Narendran, R. Borda, and S. Silvera-Arcos, "Unleashing the Power of Bat Optimized CNN-BiLSTM Model for Advanced Network Anomaly Detection: Enhancing Security and Performance in IoT Environments," *Alexandria Engineering Journal*, vol. 84, pp. 333–342, Nov. 2023.

[14] X. Wang, D. Pi, X. Zhang, H. Liu, and C. Guo, "Variational Transformer-Based Anomaly Detection Approach for Multivariate Time Series," *Measurement* vol. 191, pp. 1–17, Feb. 2022.

[15] A. Deng, and B. Hooi, "Graph Neural Network-Based Anomaly Detection in Multivariate Time Series," *Proc. of the AAAI Conference on Artificial Intelligence*, Vancouver, British Columbia, Canada, 2021. pp. 5076–5084.

[16] J. Bi, Z. Wang, H. Yuan, J. Zhang, and M. Zhou, "Cost-Minimized Computation Offloading and User Association in Hybrid Cloud and Edge Computing," *IEEE Internet of Things Journal*, doi: 10.1109/JIOT.2024.3354348.

[17] Z. Chen, C. Yeo, B. Li, and C. Lau, "Autoencoder-Based Network Anomaly Detection," *2018 Wireless telecommunications symposium*, Phoenix, AZ, USA, 2018, pp. 1–5.

[18] R. Chen, G. Shi, W. Zhao, and C. Liang, "A Joint Model for IT Operation Series Prediction and Anomaly Detection," *Neurocomputing*, vol. 448, pp. 130–139, Apr. 2021.

[19] A. Terbuch, P. O'Leary, N. Khalili-Motlagh-Kasmaei, P. Auer, A. Zöhrer, and V. Winter, "Detecting Anomalous Multivariate Time-Series Via Hybrid Machine Learning," *IEEE transactions on instrumentation and measurement*, vol. 72, no. 2503711, pp. 1–11, Jan. 2023.