Expert Systems With Applications 263 (2025) 125766

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

DAN: Neural network based on dual attention for anomaly detection in ICS

Lijuan Xu [a,b,c,*], Bailing Wang [a], Dawei Zhao [b,c], Xiaoming Wu [b,c]

[a] School of Computer Science and Technology, Harbin Institute of Technology, Weihai 264209, China
[b] Key Laboratory of Computing Power Network and Information Security, Ministry of Education, Shandong Computer Science Center (National Supercomputer Center in Jinan), Qilu University of Technology (Shandong Academy of Sciences), Jinan 250014, China
[c] Shandong Provincial Key Laboratory of Computer Networks, Shandong Fundamental Research Center for Computer Science, Jinan 250014, China

ARTICLE INFO

Keywords:
Industrial control systems
Anomaly detection
Multivariate time series
Dual attention

ABSTRACT

In the interpretability research on anomalies of Industrial Control Systems (ICS) with Graph Convolutional Neural Networks (GCN), the causality between the equipment components is a non-negligible factor. Nonetheless, few existing interpretable anomaly detection methods keeps a good balance of detection and interpretation, because of inadequate insufficient learning of causality and improper representation of nodes in GCN. In this paper, we propose a Dual Attention Network (DAN) for a multivariate time series anomaly detection approach, in which temporal causality based on attention is used for representing the relationship of device components. With this condition, the performance of detection is hardly satisfactory. In addition, in the existing graph neural networks, hyperparameters are used to construct an adjacency matrix, so that the detection accuracy is greatly affected. To address the above problems, we introduce a graph neural network based on an attention mechanism to further learn the causal relationship between device components, and propose an adjacency matrix construction method based on the median, to break through the constraint of hyperparameters. In terms of interpretation and detection effect, the performed experiments using the SWaT and WADI datasets from highly simulated real water plants, demonstrate the validity and universality of the DAN.[1]

1. Introduction

The industrial control system is an automatic control system consisting of a computer and industrial process control components, such as sensors and actuators. Attacks against industrial control systems have been frequent, and have caused significant economic losses and casualties (Li et al., 2016). Therefore, research on attack detection methods for industrial control systems has attracted more and more scholars (Liu et al., 2018; Zhu et al., 2018; Xu et al., 2021b; Chen et al., 2022). Among them, the development and wide application of Convolutional Neural Networks (CNN) (Sigaki et al., 2020; Surucu et al., 2021; Lopes et al., 2022; Ribeiro et al., 2023), which has a long history in industrial control systems, have injected impetus into the improvement of intrusion detection technology. Subsequently, more and more detection methods based on deep learning models have achieved greater success (Geiger et al., 2020; Chen et al., 2021; Li et al., 2021; Han and Woo, 2022; Gao et al., 2023). However, deep learning models are black boxes, they can identify features with large anomalous deviations but are unable to directly judge whether the correctly detected features are the ground-truth causes of anomalies. In network security, the interpretability of anomalies helps users rapidly discover attack points or attack targets, and take attack strategies to block attacks in time, lastly achieving the purpose of reducing the attack loss.

Making deep learning models interpretable, some scholars have conducted deep research in the field of image and medicine: The post hoc explanation methods develop interpretation models to further explain the results of deep learning models. Inherent interpretation methods, also known as transparent interpretation methods, construct models with functions of both detection and interpretation simultaneously. In network security, DeepAID (Han et al., 2021) is a typical post hoc method. It infers the input feature that produced the detection result, from the output neuron backward along the direction opposite to the direction of propagation.

In industrial control systems, devices collaborate to execute control tasks. Operations between equipments have a certain sequential relationship. DeepAID is less prone to interpreting actuator anomalies and more sensitive to interpreting sensor anomalies. GDN (Deng and Hooi, 2021), an inherent interpretation method using a graph neural network

* Corresponding author.
E-mail addresses: xulj@sdas.org (L. Xu), wbl@hit.edu.cn (B. Wang), zhaodw@sdas.org (D. Zhao), wuxm@sdas.org (X. Wu).
[1] https://github.com/xuljabc/DAN-source-code.git.

https://doi.org/10.1016/j.eswa.2024.125766
Received 31 December 2023; Received in revised form 10 June 2024; Accepted 9 November 2024
Available online 14 November 2024

based on attention mechanism, captures the correlation between sensors and actuators by cosine similarity to mine the correlation, and selects the top K edges with closer correlation for constructing the adjacency matrix. It demonstrates good detection performance, meanwhile having some abilities of interpretation. However, the status of device, such as temperature, pressure, water level, et al. implies causality, the causal relationship between components does not necessarily present a strong correlation. Furthermore, the selected hyperparameter K would affect the efficiency of detection and interpretation.

Consequently, in the research on the interpretability of ICS, the causality between the equipment components is an essential factor. Although, GDN conducts causal learning, the learning outcomes that its performance of interpretability is insufficient. The lack of causal learning ability of GDN in the process of constructing the adjacency matrix directly affects the balance between detection performance and interpretation ability.

For instance, in the real attack scenario on SWaT, the real attacked component is "HMI_P_302", and "HMI_FIT_401_PV" is the affected component. The above observations and predictions of attacked and affected components on the SWaT, are as shown in Fig. 1(a). While, GDN finds that "HMI_PIT_501_PV", with the highest outlier score(shown as a blue triangle), has no relationship with any actually attacked component (shown as a red triangle) or affected component as shown(shown as a yellow circle) in Fig. 1(b).

To bridge this gap, we use temporal causality based on attention instead of cosine similarity, to mine the causal relationship between components deeply and adopt the results after causal learning to represent the nodes of the graph, instead of embedding representation in GDN. Except for that, we introduce a novel algorithm to construct an adjacency matrix without hyperparameters.

In this paper, we propose DAN, a Dual Attention Network for multivariate time series anomaly detection approach, which uses temporal causality based on attention and graph attention networks, to discover the causality of components in ICS.

The main contributions of our work are summarized as follows.

- We propose a dual-attentional interpretability strategy, in which causality between different types of device components is learned by a temporal convolution and GCN based on attention, aiming at the problem of inaccurate interpretation for attacks caused by the difficulty in learning the causal relationship between device components, such as sensors and actuators.
- An adjacency matrix construction method based on the median is introduced, to break through the constraint of hyperparameters. In the existing graph neural networks, the adjacency matrix is constructed by various methods, such as all the learned edges (Chen et al., 2021), selecting the first K edge with higher relation value (Deng and Hooi, 2021), and selecting the edge with relation value greater than the threshold. In conclusion, hyperparameters are used to construct an adjacency matrix, leading to instability in the accuracy of detection.
- The universality and validity of DAN are verified by comparing with five state-of-the-art post hoc and inherent interpretability algorithms on both SWaT and WADI datasets from highly simulated real water plants. As a lack of quantifiable effectiveness evaluation in the interpretability research of attack point and attack mode detection, we introduce an evaluation method for the interpretability effectiveness of detection results. The experimental results demonstrate that the proposed algorithm generally improves by 49.4 %, compared with the classical GDN algorithm on the SWaT dataset in terms of the performance of both detection and interpretation. In additions, the detection precision and F1 score of DAN exceed those of state-of-the-art algorithms on the SWaT dataset.

The remainder of this paper is organized as follows. Section II reviews the existing anomaly detection and interpretability method. Section III provides preliminaries about anomaly detection. Section IV introduces our approach in detail. Section V describes experimental results and analysis using datasets from ICS. Conclusions and suggestions for future work are presented in Section VI.

2. Related work

With the growth of data quantity and dimension, anomaly detection methods based on Deep Neural Networks (DNN) have gradually risen and have reached a wealth of research results. For users, the deep learning model is a black box, and interpretability research on the model itself helps to judge the fidelity of the model and improve the classification or prediction performance. In addition, the interpretability of anomaly detection results can accurately locate the position of attacks and take measures to block attacks in time.

2.1. Detection for anomaly detection

In multivariate time series anomaly detection research, there are

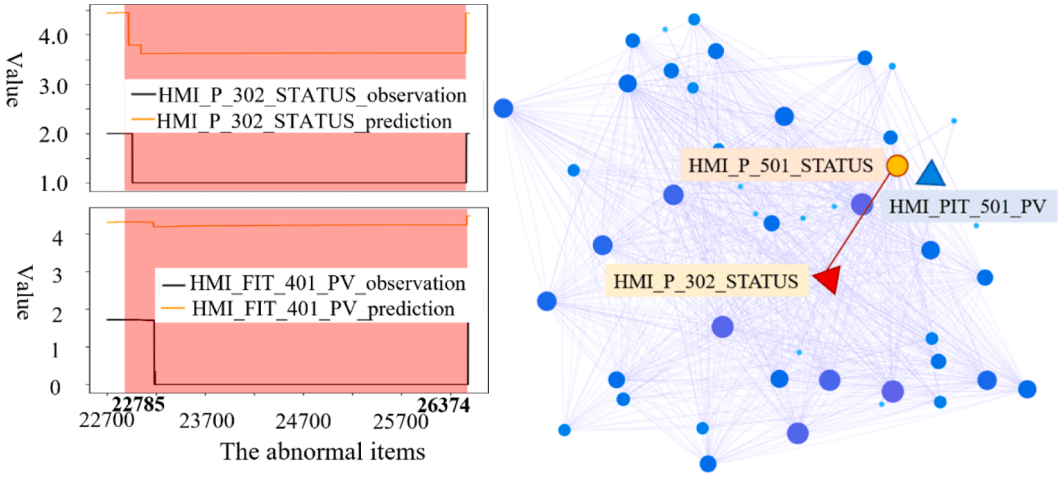


Fig. 1. Real attack scenario on SWaT and interpretation result of GDN.

many typical methods have been proposed, such as the methods based on linear model (Yang et al., 2006), the methods based on distance (Carcano et al., 2011), the methods based on one class support vector machines (Zhu et al., 2018), clustering (Xu et al., 2021b; Xu et al., 2021c) etc. Above methods acquire linear relationship between multivariate time series, which is not suitable for detecting equipment states anomalies with nonlinear relationships in real scenarios.

With the extensive research of machine learning and deep learning algorithms, researchers mostly implement multivariate time series anomaly detection based on reconstruction, prediction, or a combination of both. Reconstruction-based methods, such as auto encoder (Sakurada and Yairi, 2014), and generative adversarial network (Geiger et al., 2020), learn the representation of the time series by reconstructing the original input based on latent variables and then use the reconstruction error as the outlier score to detect anomalies. Prediction-based methods, including recurrent neural network, long short-term memory neural network(LSTM) (Lai et al., 2018), Gated recurrent unit (GRU) (Cho et al., 2014), Transformer (Chen et al., 2021; Xu et al., 2022), Autoformer (Wu et al., 2021), deep variational graph convolutional recurrent network (Chen et al., 2022; Gao et al., 2024) and other methods predict the state of equipment components in the next moment or a period and compare the predicted value with the actual value to achieve the purpose of detection. By combining both methods, the prediction-based methods are applied to a certain stage of the reconstruction algorithm. For instance, LSTM is applied to the feature extraction step of the generative adversarial network (Li et al., 2019), and GNN is applied to the stacking variational autoencoder (VAE) (Li et al., 2021).

2.2. Interpretations of anomaly detection based on DNN

There is much research relative to the interpretability of deep learning models in natural language processing, image recognition, medicine, and so on (Chen et al., 2019; Dwivedi et al., 2022; Mahapatra et al., 2023). According to the time of interpretation and detection, interpretable anomaly detection research is divided into two categories: post hoc interpretation and inherent interpretation (also called transparent interpretation).

2.2.1. Post hoc interpretation

Post-hoc interpretation refers to an interpretability approach that explains the reason for the anomaly occurrence according to the anomalies detected by anomaly detection models. By the implementation principle of the interpretation method, the post hoc interpretation is generally divided into approximation-based interpretation, perturbation-based forward propagation interpretation, and back propagation-based interpretation.

(1) Approximation based interpretation

Approximation-based interpretation uses a new and simple model to find a decision boundary of a specific sample for gaining on the original DNN. Furthermore, a simple model provide the interpretation of the sample.

A model-agnostic linear model (LIME) (Ribeiro et al., 2016) covers certain features in the vicinity of the instance to be explained for the classifier or regressor. A non-linear mixture regression model (LEMNA) (Guo et al., 2018) leverages fused Lasso to cope with the feature dependence problem in RNN. A unified framework (SHAP) (Lundberg and Lee, 2017), interprets predictions by a feature importance value assigned for a particular prediction by using game theory in supervised scenarios. Liu et al. (Liu et al., 2018) train a surrogate linear SVM from interpreted anomalies and normal data, supporting to interpret unsupervised models. Bhatt et al. (Bhatt et al., 2019) combine SHAP (Lundberg and Lee, 2017) and Integrated gradients (Sundararajan et al., 2017), with antecedent event influence to build post hoc local explanations and global patterns in supervised classification tasks. Based on the trained VAE, Ikeda et al. (Ikeda et al., 2019) explore a true latent distribution, through constructing an approximative probabilistic model. By maximizing the log-likelihood, they estimate which features contribute to determining data as an anomaly.

Giurgiu et al. (Giurgiu and Schumann, 2019) explain anomalies with a GRU-based autoencoder by extending SHAP. Zhang et al. (Zhang et al., 2019) provide insights into diagnosing which attributes significantly contribute to an anomaly by building a specialized linear model to locally approximate the anomaly score that a black-box model generates.

(2) Perturbation-based forward propagation interpretation

Perturbation-based forward propagation interpretation approaches add some perturbations to individual inputs or neurons and further observe the impact on later neuron.

Zeiler et al. (Zeiler and Fergus, 2014) visualize the change in the activation of later layers by perturbing different segments of an input image. Zintgraf et al. (Zintgraf et al., 2017) margin over each input patch, and then analyze the difference in a prediction. Kauffmann et al. (Kauffmann et al., 2020) necrotize predictions and use deep Taylor decomposition to obtain explanations. Yang et al. (Yang et al., 2021) perturb the inputs and further observe the distance changes to the nearest centroid in the latent space, for unsupervised interpretation of concept drift samples.

(3) Back propagation-based interpretation

Back propagation-based interpretation approaches infer key input reference backward through the layers from an output neuron to the input in one pass.

Simonyan et al. (Simonyan et al., 2014) adopt the gradient of the output pixels of an input image to compute a “saliency map” of the image. Sundararajan et al. (Sundararajan et al., 2017) attribute the prediction of a deep network to its input features by integrating gradients. Shrikumar et al. (Shrikumar et al., 2017) backpropagate the positive and negative contributions of all neurons in the network to the input features. Han et al. (Han et al., 2021) argue that existing explanatory methods are not directly applicable to deep learning-based anomaly detection in security applications, so, for tabular data, time series data, and graph data, they interpret anomaly based on gradient differences.

The methods of post hoc interpretation focus on the analysis of the deep learning model, which executes after the end of the attack detection. They have the defect of not being able to locate the components that suffering from attacks or the most affected components under attack, the inherent interpretation methods can bypass the above defect. Accordingly, we focus on the inherent interpretation method.

2.2.2. Inherent interpretation

In addition to certain models with the ability of inherent interpretation (Letham et al., 2015), recently, many scholars have introduced GNN and attention mechanism into the research of outliers’ interpretation to realize the transparent interpretation of anomalies.

Zhao et al. combine the two graph attention layers oriented to feature and time to learn the graph structure and used the prediction model and reconstruction model to calculate the joint loss, to realize the anomaly detection and interpretation of multivariate time series. Xu et al. (Xu et al., 2021a) propose a new attention-guided Triple Deviation network (ATON) for anomaly interpretation. ATON learns the embedding space directly and how to pay attention to each embedding dimension, capturing each dimension’s contribution to the query outliers. Li et al. (Li et al., 2021) propose to use graph neural networks and stack variational autoencoders to realize anomaly detection and interpretation of timing data. Deng et al. (Deng and Hooi, 2021) introduce the concept of a “Graph Deviation network”, named GDN, which uses a graph attention network for feature extraction, and attention weight to provide interpretability for anomaly detection. Chen et al. (Chen et al.,

2021) use Transformer and graph neural networks to detect and interpret data from the Internet of Things (IoT). Han et al. (Han and Woo, 2022) propose Fused Sparse Autoencoder and Graph Net (FuSAGNet), which jointly optimizes reconstruction and forecasting while explicitly modeling the relationships within multivariate time series. Casajús-Setién et al. (Casajús-Setién et al., 2022) propose a semi-supervised AD with Bayesian networks using generative-adversarial training and an evolutive strategy, aims to palliate the intrinsic lack of interpretability of deep neural networks. Ding et al. (Ding et al., 2023) propose a multi-modal spatial–temporal graph attention network (MST-GAT) to capture spatial–temporal relationships between univariate time series of different modalities. They provide graphical interpretable results and focus on reducing false negatives and false positives.

In summary, the research of attack detection based on graph neural networks focuses on the feature representation of device components and the description of the component relationship.

A series of research results have been obtained in the construction of an adjacency matrix. The device component features are represented by an embedding layer or one-dimensional convolution. The relationships between the components of networks are learned using cosine similarity (Deng and Hooi, 2021), Gumbel-Softmax sampling strategy (Chen et al., 2021), CNN (Wu et al., 2020), channel embedding (Xu et al., 2022), coefficient potential representation and regression feature embedding (Han and Woo, 2022). The adjacency matrix is constructed by using all the learned edges (Chen et al., 2021), selecting the top K edges with a higher relation value (Deng and Hooi, 2021), and selecting the edge with a relation value greater than the threshold, etc.

Most of the above studies excavate correlation rather than causality between component states. Although some components are attacked, their status is normal. For example, in a water treatment system, the status of the device components affected by the attacked component may be abnormal. However, these two causal device components do not necessarily show a strong correlation.

3. Preliminaries

In the ICS, N sensors collect the state data of multiple components of M ($M < N$) equipment, such as pumps, tankers, and so on.

The training dataset contains only the data of normal operation of the system, which is the multivariate time series $X_{train} = \left[X_{train}^{(1)}, \ldots, X_{train}^{(T_{train})}\right]$ obtained from N sensors in T_{train} time periods. At each time t, the value of the sensor $X_{train}^{(t)} \in R^N$ is an N-dimensional vector, in which each element from N sensors. The test dataset $X_{test} = \left[X_{test}^{(1)}, \cdots, X_{test}^{(T_{test})}\right]$ comes from the same N sensors, and there are multiple attacks in the T_{test} time period, with different components attacked each time.

According to whether the maximum outlier score $S = \left[S_{test}^{(1)}, \cdots, S_{test}^{(T_{test})}\right]$ exceeds the threshold, the algorithm outputs labels $Y_{test} = \left[Y_{test}^{(1)}, \cdots, Y_{test}^{(T_{test})}\right]$, where $Y_{test}^{(i)}$ indicates whether the component state at each time is normal or abnormal. $Y_{test}^{(i)} = 1$ indicates abnormal. The sensor corresponding to the maximum outlier score $S_{test}^{(i)}$ can indicate the position of the attack point (the attacked sensor, the affected sensor).

4. Methodology

4.1. Overview

The architecture of our proposed approach includes a neural network based on dual attention, time series forecasting, anomaly detection and interpretation, described as in Fig. 2.

A neural network based on dual attention proposed in this paper is a kind of GCN. Dual attention used to learn the causality of components, includes two factors, which are the **A**ttention mechanism to obtain **P**reliminary **R**epresentation of component features (**APR**), used as the

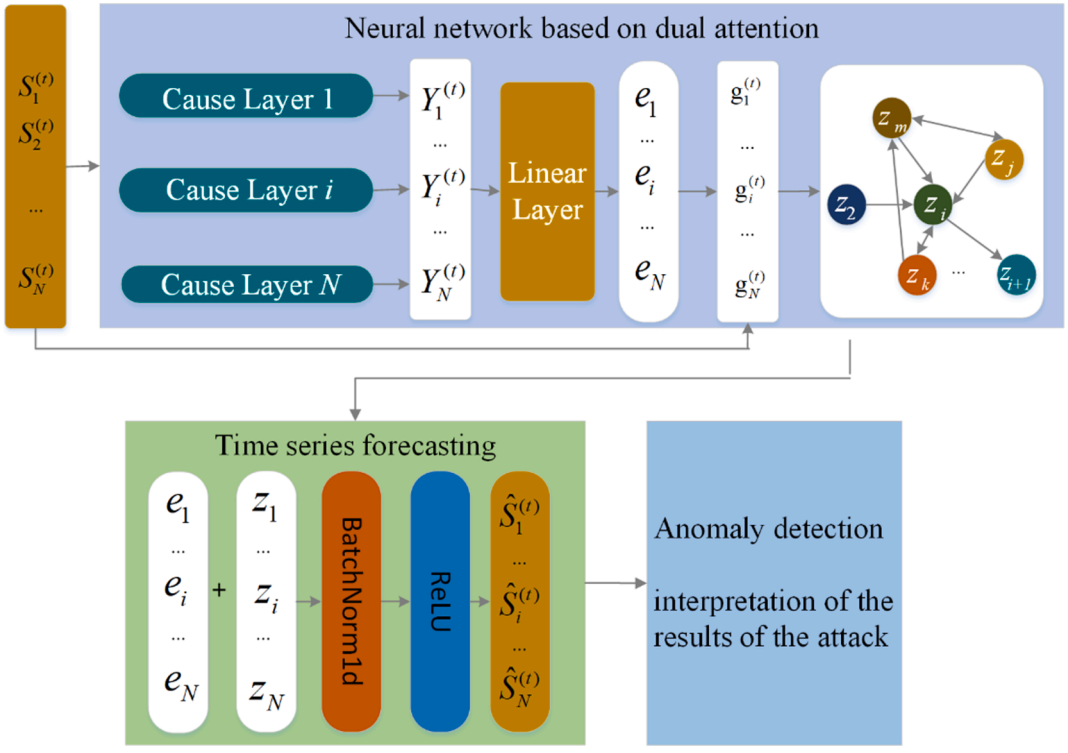


Fig. 2. The overall architecture of the DAN. DAN consists of three modules, i.e., the neural network based on dual attention module, the time series forecasting module, and the anomaly detection and interpretation module.

node representation of a GCN, and the **A**ttention mechanism to further **I**nfer the **C**ausality of GCN's nodes (**AIC**). Besides, the future component status is predicted with the help of the inference node features. The inferential anomalies and causality are the interpretability of the attacked components.

Time series forecasting inputs the outputs of APR and AIC, and outputs the predicted time series data.

The anomaly detection and interpretation module, compares the values of the real time series data with the predicted time series data, and interprets the attacked components or greatly affected components while implementing the detection.

In DAN, APR is used to represent the node of a GCN, and AIC is responsible for further inferring the causality of nodes. In a neural network based on dual attention, the nodes are the device components. This paper uses APR and AIC to describe the neural network based on dual attention.

4.2. APR

Dilated causal convolution has been used to learn the causality of multi-dimensional data, and has achieved good learning results (Bai et al., 2018). The causal convolution is a time-constrained model with unidirectional structure. Only with the previous cause can the subsequent effect be generated. Standard CNN can obtain a larger receptive field by adding a pooling layer, but there is definitely the problem of information loss after the pooling layer. To solve this problem, the standard convolution is injected with dilation, that is, the dilated convolution is formed to increase the receptive field. This paper uses the idea of dilated causal convolution to pre-train the causal relationship between components.

The attention weights between time series data learned after training were used to represent the initial relationship between components.

Let the sensor value at time t be $v^{(t)} = \left\{v_1^{(t)}, v_2^{(t)}, \cdots, v_N^{(t)}\right\}$, the sliding window size be ω, the historical time series data of N sensors before time t be $S^{(t)} = \left\{S_1^{(t)}, S_2^{(t)}, \cdots, S_N^{(t)}\right\}$, where $S_i^{(t)} = \left\{v_i^{(t-\omega)}, v_i^{(t-\omega+1)}, \cdots, v_i^{(t-1)}\right\}$.

At this point, $v^{(k)} = \left\{v_1^{(k)}, v_2^{(k)}, \cdots, v_N^{(k)}\right\}$, $v^{(k)} \in R^N$, $k \in \{t-\omega, t-\omega+1, t-1\}$.

Let the dilation size be d_s, the dilation coefficient be d_c, and the number of dilated convolution layers be L. We have $d_s = d_c^{\,l}$, in which $l = \{0, 1, \ldots, L\}$.

As shown in Fig. 3, to maintain the size of the output data across the dilated convolution layer, padding nodes are required to be added before the input nodes. In the end, the number of padding nodes should be $(knl-1)^* d_s$, when kernel size is knl.

$\alpha_i = [\alpha_{i1}, \alpha_{i2}, \cdots, \alpha_{iN}]$ is attention scores.

The softmax function is applied to the attention scores during the network training process, such that $\sum_{i=1}^{N} \alpha_{ij} = 1$.

$$a_{ij} = Softmax(\alpha_{ij}) = \frac{exp^{\alpha_{ij}}}{\sum_{p=1}^{N} exp^{\alpha_{ip}}} \tag{1}$$

$$Y_i^{(t)} = F\left(a_{i1}S_1^{(t)} + a_{i2}S_2^{(t)} + \cdots + a_{ij}S_j^{(t)} + \cdots + a_{iN}S_N^{(t)}\right) \tag{2}$$

In which, $A_i = [a_{i1}, a_{i2}, \cdots, a_{ij}, \cdots, a_{iN}]$.

The function $F(\bullet)$ takes a_i and $\widetilde{S}_i^{(t)}$ as input, and $Y_i^{(t)}$ as output during the first causality learning process. Fig. 3 describes the details of the cause layer i from Fig. 2. First, $\widetilde{S}_i^{(t)}$ is input to a dilated convolution with $N \times N$. Second, since the data generated by the previous layer is larger than the original data, add a clipped layer to remove the redundant data. Third, a series of temporal layers are used to learn temporal causal information between components, in which, each temporal layer is composed of a dilated convolution with $N \times N$ and clipped layer, besides, a resnet module is added for resolving the problem that the training effects become worse when the number of layers is too deep.

This paper uses the mean square deviation between the predicted value $Y_i^{(t)}$ and the original data value $S_i^{(t)}$ as the loss function.

$$Loss_1 = \frac{1}{N}\sum_{i=1}^{N}\left(Y_i^{(t)} - S_i^{(t)}\right)^2 \tag{3}$$

To make graph nodes retain original data characteristics, $Y_i^{(t)}$ learned in Formula (3) as the parameter, is used to represent graph nodes.

The calculation formula of the graph node is,

$$e_i = W_i Y_i^{(t)} + B_i, e_i \in R^d, i \in \{1, 2, \cdots, N\} \tag{4}$$

Fig. 4 describes the overall process of constructing the adjacency matrix.

We introduce an adjacent matrix calculation process based on intervals median of attention scores, as described in Algorithm 1. The input is the attention scores A_i of $Y_i^{(t)}$. *atdList* is a set of features that influence the change of $Y_i^{(t)}$ and their corresponding attention score List is *atdWList*.

The output is a $N \times N$ adjacent matrix *adjMatrix*, input to the graph neural network. $\widehat{A}_i$ is obtained by sorting $\widehat{A}_i$ from large to small, and indices are the corresponding sequence number in sorted A_i. When the member in $\widehat{A}_i$ is less than 1, we stop calculating the intervals of adjacent members of $\widehat{A}_i$, as the initial value of all members of A_i is set to 1, since an attention score of less than 1 means that the feature corresponding to

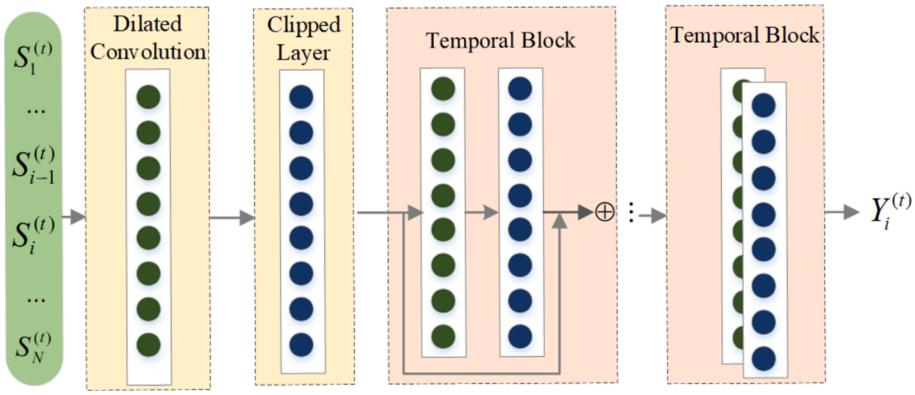


Fig. 3. Causality layer.

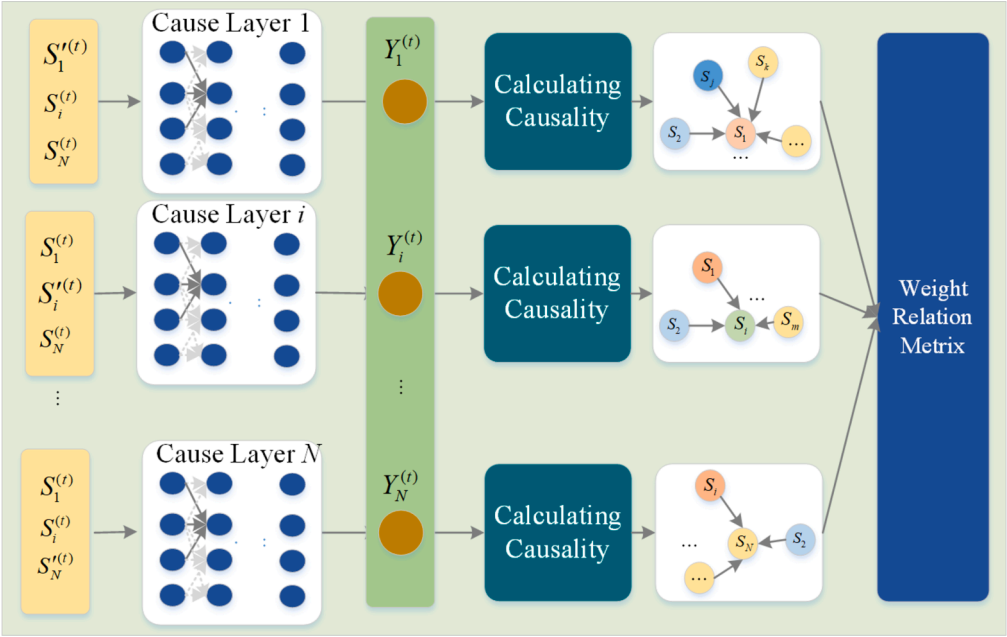


Fig. 4. The overall process of constructing the adjacency matrix.

this value is less causal.

Algorithm 1. Calculating Adjacent Matrix
Input: The attention scores $[A_1, A_2, \cdots, A_N]$.
Output: A $N \times N$ adjacent matrix *adjMatrix*
1: *intervals*←∅, *interval*←0, *adjMatrix*←0;
2: **for** each A_i in $[A_1, A_2, \cdots, A_N]$ **do**
3: *atdList*←∅, *atdWList*←∅, *adjMatrix*[*N*][*i*] = [0, 0, ⋯, 0]
4: $\widehat{A}_i$= Sort(A_i);
5: indices = argSort(A_i);
6: *j*←0;
7: **while** *j* < *len*($\widehat{A}_i$) −1 **do**
8: if $\widehat{a}_{i,j} < 1$ then
9: break;
10: end if
11: interval = $\widehat{a}_{i,j} - \widehat{a}_{i,j+1}$;
12: intervals.add(interval);
13: *j*←*j* + 1;
14: **end while**
15: sortItvs = Sort(intervals);
16: itvsMedian = median(intervals);
17: *j*←0;
18: **while** *j* < *len*(*intervals*) **do**
19: largestItv = sortItvs[j];
20: idx = intervals.index(largestItv);
21: *ind*← −1;
22: **if** *largestItv* < *itvsMedian* and *idx* > 0 **then**
23: ind = idx;
24: break;
25: **end if**
26: *j*←*j* + 1;
27: **end while**
28: **if** *idx* > 0 then
29: *atdList*←*indices*[: *ind* +1];
30: *atdWList*←$\widehat{A}_i$[: *ind* +1];
31: **for** each atd in atdList do
32: *adjMatrix*[*atd*][*i*]←1;
33: **end for**
34: **end if**
35: **end for**

Finally, we obtain the split point whose interval value is less than the median of the intervals and use this point as the position of the abnormal score to intercept, which expands the range of causal learning in the original Temporal Convolution Network (TCN), without causing excessive resource consumption due to the excessive selection of threshold value as in the causal selection based on threshold value.

4.3. AIC

In this section, Graph Convolutional Neural Network (GCN) further learns the causal relationship between nodes, and the updated graph node is adopted to predict the value of the sensor at time *t*.

GCN takes graph node vector e_i of Formula (4) and original data $S^{(t)}$ as input, predicted vector $v^{(t)}$ at time *t* as output.

The graph node vector and the original data $S^{(t)}$ compute the joint vector $g_i^{(t)}$, so that the graph node learns the original data features again.

$$g_i^{(t)} = e_i \oplus W_i^2 S_i^{(t)} \tag{5}$$

Next, LeakyReLU function calculates the attention coefficient.

$$\pi(i,j) = LeakyReLU\left(W_i^3\left(g_i^{(t)} \oplus g_j^{(t)}\right)\right) \tag{6}$$

The aggregated node feature $z_i^{(t)}$ is expressed as,

$$z_i^{(t)} = \mathrm{ReLU}\left(\pi(i,i)W_i^2 S_i^{(t)} + \sum_{j \in N\{i\}} \pi(i,j) W_i^2 S_j^{(t)}\right) \tag{7}$$

4.4. Time series forecasting with DAN

In the final prediction algorithm, we take the dot product of $z_i^{(t)}$ with the initial node vector e_i, and then perform batch normalization to get the predicted data.

$$\widehat{S}^{(t)} = \mathrm{ReLU}\left(BatchNorm1d\left(e_1 \circ z_1^{(t)}, e_2 \circ z_2^{(t)}, \cdots, e_N \circ z_N^{(t)}\right)\right) \tag{8}$$

Formula (9) defines the loss function.

$$Loss_2 = \frac{1}{T_{train} - \omega} \sum_{t=\omega+1}^{T_{train}} \left|\left|S^{(t)} - \widehat{S}^{(t)}\right|\right|_2^2 \tag{10}$$

The final loss function of the model is:

$$Loss = Loss\_1 + Loss\_2 \tag{11}$$

4.5. Detection of attacked component

This paper compares the predicted and ground-truth values of all sensors at time t to calculate the abnormal score of each sensor at time t. If the abnormal score is larger than the threshold, we consider that an attack occurs at time t. In this case, the sensor with the highest score is selected as the attacked sensor or the sensor receiving an attack.

The absolute value of the difference between the predicted value and the ground-truth value of sensor i at time t is defined as the abnormal score.

The value ranges of data collected by sensors are different. As a result, to compare the abnormal score of the same sensor and the deviation of abnormal score between different sensors, the median and interquartile are used to normalize the abnormal score here. In addition, we choose Simple Moving Average (SMA) to solve the problem of inaccurate detection results caused by abnormal score mutation (Hundman et al., 2018). Finally, the extreme value theorem (Siffer et al., 2017) is used to select the anomaly threshold. When the anomaly score after the above processing is greater than the threshold, we consider that an attack occurs at time t.

5. Evaluation

We conducted our experiments on NVIDIA GeForce RTX 3080. The experiments were carried out using Python version 3.7.13, PyTorch version 0.12.1, and cuda 11.3. The maximum of epochs is set to 100. We observe that the *Loss* is almost stable before epochs reaching maximum, so we have implemented an early stopping mechanism. The training process breaks when the *Loss* does not decrease for 10 times, thereby preventing overfitting. Through experiments, we find that when *batch* is set to 128, L is 2 and d_c is 2, we can achieve the optimal detection and interpretation effect. For the meanings represent by L and d_c, please refer to 4.2.

5.1. Testbed and datasets

This paper selects datasets generated on the water treatment test-bed called Secure Water Treatment (SWaT) (Mathur et al., 2016) and Water Distribution (WADI) (Ahmed et al., 2017), coordinated by the iTrust Laboratory at the Singapore's Public Utility Board in the field of industrial control, to demonstrate the experiment results. The iTrust Laboratory focuses on research in the security of industrial control systems and networks, actively working towards creating practical testing environments and datasets to support related studies. Therefore, SWaT and WADI are not artificially simulated data. Since the construction of the water treatment test bed and the attack scenarios simulated on it are based on real water plants, the test dataset including the attack, generated by them not only contains normal and abnormal labels but also marks the name of the specific equipment components under attack, which plays an extremely essential role in evaluating the interpretation ability of the model.

It takes 5–6 h for the system to start up before it enters normal operating status, therefore, we remove the first 21,600 samples of the training dataset. To speed up the training, similar to GDN, we replace every 10 (10 s) of data with the mean value to reduce the original dataset. Accordingly, the labels that account for the majority of the 10 labels are used to set the new data. Table 1 shows the processed features of the dataset, the size of the training dataset, the size of the test dataset, and the proportion of exceptions in the test dataset.

Table 1
Dataset Statistics.

Data set	Features	Train	Test	Anomalies
SWaT	51	496,800	449,919	11.98 %
WADI	127	1,048,571	172,801	5.99 %

5.2. Evaluation measures

5.2.1. Evaluation measure of detection methods

Precision(P), Recall(R), and F1 score (F1) are used to evaluate the performance of anomaly detection with detection methods and inherent interpretability detection methods respectively with SWaT and WADI.

5.2.2. Evaluation measure of interpretability

The performance of the interpretability method can be reflected by the agreement between ground-truth components under attack and the correctly detected anomalies. Let $C = C_1, C_2, \cdots, C_n$ refer to the set of detected anomalies which is the ground-truth ones. For an anomaly $C_i (i \in [1, n])$ in C, if the components identified as attacked by the interpretability model are exactly the components that are actually attacked or are most affected by the ground-truth attacked components, manifested in the graph nodes, which are the tail nodes with the identified node as the head node and the strongest causality with the identified node, then we consider C_i is correctly interpreted by the interpretability model. Let $n_i \leq n$ be the number of correctly interpreted anomalies. Generally, researchers adopt Recall of Interpretability (*IR*) as a performance measure to quantify the ability of the interpretability model to explain the valid anomaly detection results. *IR* can be expressed as Formula (11).

$$IR = \frac{n_i}{n} \tag{11}$$

5.2.3. Joint evaluation measure of detection and interpretability

According to Formula (11), the value of *IR* relates to the number of ground-truth abnormal samples detected by the anomaly detection algorithm. When the number of detected ground-truth samples is the same, the more the number of abnormal samples correctly interpreted, the higher the value of *IR*. A high *IR* value indicates that the model has a good ability of interpretability when we have the same interpreted set C. However, this situation is less likely to exist. In most cases, the set C obtained by the detection method is different from each other. Since Recall of detection methods refers to the ratio of correctly detected abnormal samples to all ground-truth abnormal samples, we suggest that the ratio *TIR* of correctly interpreted abnormal samples to all ground-truth abnormal samples is used as a measure to evaluate the explanatory ability of the interpretation algorithm. Let n_g be the number of all ground-truth abnormal samples, $TIR = \frac{n_i}{n_g}$ and $Recall = \frac{n}{n_g}$. Consequently, *TIR* is defined as Formula(12).

$$TIR = IR^*R \tag{12}$$

5.3. Performance comparison and analysis

This paper compares the performance of the proposed approach DAN to that of several existing interpretability approaches. Five algorithms are chosen as baselines, i.e., LIME (Ribeiro et al., 2016), Occlusion (Zeiler and Fergus, 2014), IG (Sundararajan et al., 2017), DeepAID (Han et al., 2021), GDN (Deng and Hooi, 2021), corresponding to approximation-based interpretation, perturbation-Based forward propagation interpretability methods, back propagation-based approaches, inherent interpretation. Since post hoc interpretability approaches interpret anomaly detection results generated by specific detection algorithms, this paper chooses four representative anomaly detection algorithms used for anomaly detection for LIME, Occlusion, IG, DeepAID algorithms, i.e., the Auto encoder (AE) (Hawkins et al., 2002), Deep Auto encoding Gaussian Mixture Model (DAGMM) (Zong et al., 2018), Long Short Term Memory Networks (LSTMAD) (Malhotra et al., 2015), and LSTM-based Encoder-Decoder (LSTMED) (Malhotra et al., 2016). The authors have described the details of them (Xu et al., 2021b). The above four anomaly detection algorithms traverse all thresholds, take the threshold with the best test performance on the test dataset as the

selected threshold, and select the optimal effect as the best detection effect. The threshold-selecting method is not reasonable in real abnormal detection. To maintain the same contrast environment, we let these algorithms use the same extreme value theorem (Siffer et al., 2017) as ours to select the anomaly threshold values.

In addition, to study the necessity of each portion of our approach, we replace the portions of DAN with original methods to observe how the model performance degrades. First, we evaluate the importance of the representation of graph nodes by substituting it with an embedding layer, denoted by "T + EMB", where the causality is learned between embedding nodes. Second, we evaluate the necessity of expanding the range of causal learning by using the original causal learning algorithm of TCN, denoted by "T + CAS".

We test different parameter configurations to obtain the best detection performance for each anomaly detection method. In the post hoc interpretation approach, since different dimensions in interpretation results demonstrate different recall of interpretation results, we set the value of dimensions K as an integer, changing from 1 to 10, to evaluate the performance interpretation approaches. In particular, as long as one of the interpretation results corresponding to the components supposed to be under attack, is the ground truth, we consider the whole interpretation result is correct. In GDN and DAN series algorithms, such as DAN, T + EMB, and T + CAS, the feature with the highest anomaly score or the closest causality to it is considered as the component feature under attack. Therefore, we suggest K is 2 in baselines to maintain a consistent interpretation of the principle.

Table 2 summarizes the detection results obtained by all compared anomaly detection algorithms.

Table 3 shows the *IR* comparisons with baseline methods in interpretability, in which, the interpretation results for LIME, Occlusion, IG, and DeepAID are computed in the case of $K = 2$.

The following observations can be derived from the results in Tables 2 and 3.

- The detection precision and F1 score of DAN exceed those of the other six algorithms with the SWaT dataset. In terms of precision, F1, and recall, DAN has achieved relatively the best performance of detection in all inherent interpretation methods with the SWaT dataset. Although the recall of AE, DAGMM, LSTMED, and LSTMAD is higher than that of DAN, in terms of precision and F1, the performance of those four methods, especially DAGMM, is worse than that of DAN. With the WADI dataset, low recall and F1 reduce the overall performance of GDN, T + CAS. As a higher precision and F1, DAN is better than T + EMB. Inherent interpretation methods are generally lower than pure detection methods in detection efficiency. A possible explanation for this may be that causal learning of component attributes increases the likelihood of using redundant information.
- Solely observing the value of IR, although the general interpretation ability of DAN is slightly weaker than that of GDN and T + CAS with the WADI dataset, DAN has a superior power with SWAT dataset, and the relative improvement of DAN over the best baseline methods in terms of IR ranges from 12.29 % to 99.7 %. This observation proves that, with the SWaT dataset, DAN can learn more information about the attacked components that are strongly correlated with the highest anomaly score from the graph. With both datasets, DeepAID can find more attacked components than any other post hoc methods, however, the best *IR* value is still 0.3 percentage points lower than that of DAN.

Table 2
Comparisons of Detection Performance with Baselines.

Methods	SWaT			WADI		
	P(%)	R(%)	F1	P(%)	R(%)	F1
AE	40.6	67.4	0.506	93.0	29.5	0.448
DAGMM	14.7	97.5	0.256	39.3	38.1	0.387
LSTMED	41.0	69.6	0.516	93.0	29.1	0.443
LSTMAD	84.1	66.7	0.744	86.3	29.6	0.441
GDN	98.3	60.8	0.751	86.1	18.6	0.305
T + EMB	96.2	59.4	0.734	51.4	24.5	0.331
T + CAS	99.3	61.4	0.759	90.5	15.4	0.262
DAN	**99.4**	61.5	**0.760**	81.5	21.2	0.336

5.4. Joint evaluation of interpretation performance by combining detection efficiency with *IR*

TIR indicates joint evaluation of interpretation performance by combining detection efficiency with *IR*.

Fig. 5 shows the comparisons of *TIR* with baselines. It indicates that the joint performance of DAN is superior to that of all compared methods with the SWaT dataset, while the joint performance of DAN is a little worse than that of Occlusion and DeepAID with the WADI dataset. The relative improvement of DAN over the best baseline methods in terms of *TIR* ranges from 11.98 % to 61.29 % with SWaT, while with WADI, the *TIR* of DAN is reduced by 8.7 percentage points compared with that of the best baseline method, Occlusion on DAGMM. In summary, the improvement degree of joint performance with SWaT greatly exceeds the reduction degree of that with WADI. In addition, we conclude that the overall joint performance of DAN is better than that of inherent interpretability algorithms, i.e., GDN and ablation algorithms.

5.5. Impacts of the dimensions *K*

We have just mentioned that the rise of K will impact the *TIR*. As our proposed approach and GDN do not involve variable K, they do not participate in the comparison of the performance of interpretability. Fig. 6 shows the impacts of K on the performance of interpretability with baselines.

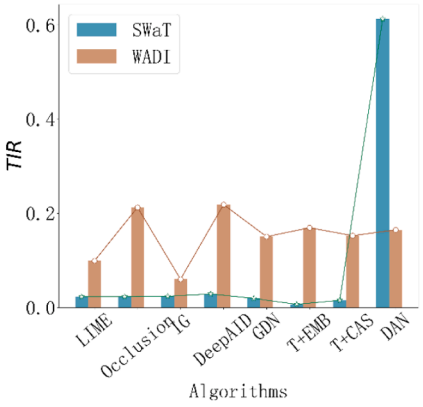
Obviously, as shown in Fig. 6, with the increase of K, the *TIR* of interpretation algorithms, including LIME, Occlusion, IG, and DeepAID, demonstrate an overall upward trend with both SWaT and WADI datasets. *TIR* of interpretation algorithms with SWaT reaches a maximum of 0.2516, lower than that with WADI with a maximum of 0.2869, which illustrates that the performance of interpretability with baselines on SWaT is worse than that on WADI. *TIR* of Occlusion, IG, and DeepAID with SWaT have similar increasing trends in the aspect of AE, DAGMM, and LSTMAD, shown in Fig. 6(a), 6(b), and 6(d). While *TIR* of LIME is lower than the above three algorithms in general, and, when $K > 3$, the increase of that is so small that it is almost negligible with both SWaT and WADI datasets. Expect for Fig. 6(b) and 6(d), When $K \geq 3$, *TIR* of Occlusion is the highest in all of the algorithms, and DeepAID is inferior to Occlusion. Also in all Figures, when $K > 3$, the performance of IG exceeds that of LIME and rises to third place. In the aspect of interpreting for detection results of DAGMM and LSTMAD on SWaT, shown in Fig. 6 (b) and 6(d), DeepAID has a better ability of interpretation than Occlusion. On WADI, when $K = 1$, *TIR* of IG is the lowest in all of the algorithms, with a minimum of 0.002, substantially lower than that of the other three algorithms, and as the rise of K, it has a large increase, gradually exceeding that of LIME, reaching the level of DeepAID and IG, except for Fig. 6(e). In conclusion, Occlusion has the best explanatory performance, followed by DeepAID, and the explanatory performance of LIME changes the least as K increases. It is considered that Occlusion is more suitable for interpreting the anomaly detection with SWaT and WADI datasets.

5.6. Comparison and analysis of graphical interpretation results

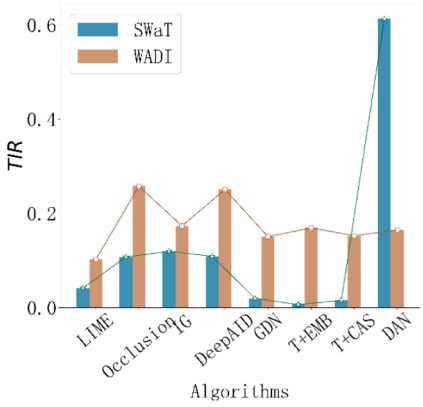
To demonstrate the interpretability qualitatively, we visually compare the proposed method with the interpretability methods described above in some specific attack scenarios.

Table 3
Comparisons of Interpretability Performance with Baselines.

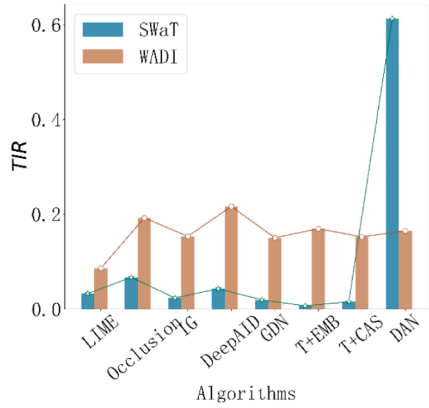
Type	Methods	SWaT				WADI			
		AE	GAGMM	LSTMED	LSTMAD	AE	GADMM	LSTMED	LSTMAD
Post Hoc	LIME	0.034	0.043	0.047	0.026	0.337	0.268	0.297	0.349
	Occlusion	0.034	0.110	0.096	0.026	0.721	0.676	0.662	0.736
	IG	0.035	0.123	0.034	0.018	0.204	0.455	0.528	0.603
	DeepAID	0.043	0.112	0.062	0.027	0.742	0.658	0.745	0.736
Inherent	GDN	0.032				0.811			
	T + EMB	0.011				0.693			
	T + CAS	0.025				**0.994**			
	DAN	**0.997**				0.779			



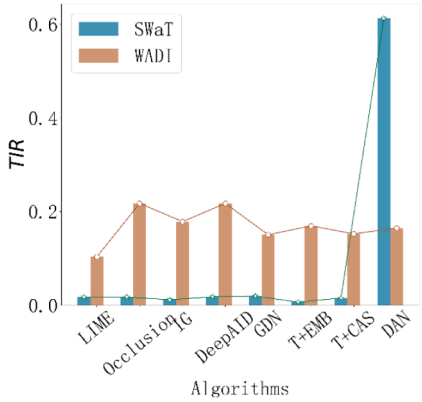
(a) *TIR* comparison with baselines in AE.



(b) *TIR* comparison with baselines in DAGMM.



(c) *TIR* comparison with baselines in LSTMED.



(d) *TIR* comparison with baselines in LSTMAD.

Fig. 5. Joint comparison of interpretation performance by combining detection efficiency with *IR,* recall of interpretability, on SWaT and WADI datasets in four detection algorithms, including AE, DAGMM, LSTMED, and LSTMAD. The x-axis represents the DAN and baselines, i.e., LIME, Occlusion, IG, DeepAID, GDN, T + EMB, T + CAS, and the y-axis represents the joint evaluation of *TIR,* which is interpretation performance.

- Illustration of attack scenario on SWaT. For attacking SWaT, when "P302" ("HMI_P_302_STATUS") was on, the attacker attacked for 11:15:27 by closing it, as a result, the inflow of tank "T-401"("HMI_FIT_401_PV") was stopped. Fig. 7(b) and 7(c) respectively show the observations and predictions of attacked "HMI_P_302_STATUS" and "HMI_FIT_401_PV".
- Illustration of attack scenario on WADI. The stealthy attack was performed on WADI for 29.0 min. In detail, attackers aimed to drain elevated reservoir "2_LIT_002"("2_LT_002_PV"). This is done by controlling and manipulating tank level draining and filling speed. Moreover, attackers change the reading seen by the water quality sensor "1_AIT_001"("1_AIT_001_PV"), and this causes the raw water tank to drain. Fig. 7(d) and 7(e) respectively show the observations and predictions of attacked "2_LT_002_PV" and "1_AIT_001_PV".

As shown in Fig. 5, the performance of DAN exceeds all other baseline algorithms on the SWaT dataset, in terms of interpretation. The reason is DAN can explain some abnormal situations which account for a large proportion of ground-truth anomalies, while other algorithms cannot correctly recognize this explanation. Fig. 8(a), 8(b), and 8(c) show graphical interpretation results on the SWaT dataset when the actually attacked component is "HMI_P_302" marked red triangle and labeled "HMI_P_302_STATUS". The component marked blue triangle is the component with the highest outlier score, which is interpreted as the component under-attacked or having the strongest correlation with the

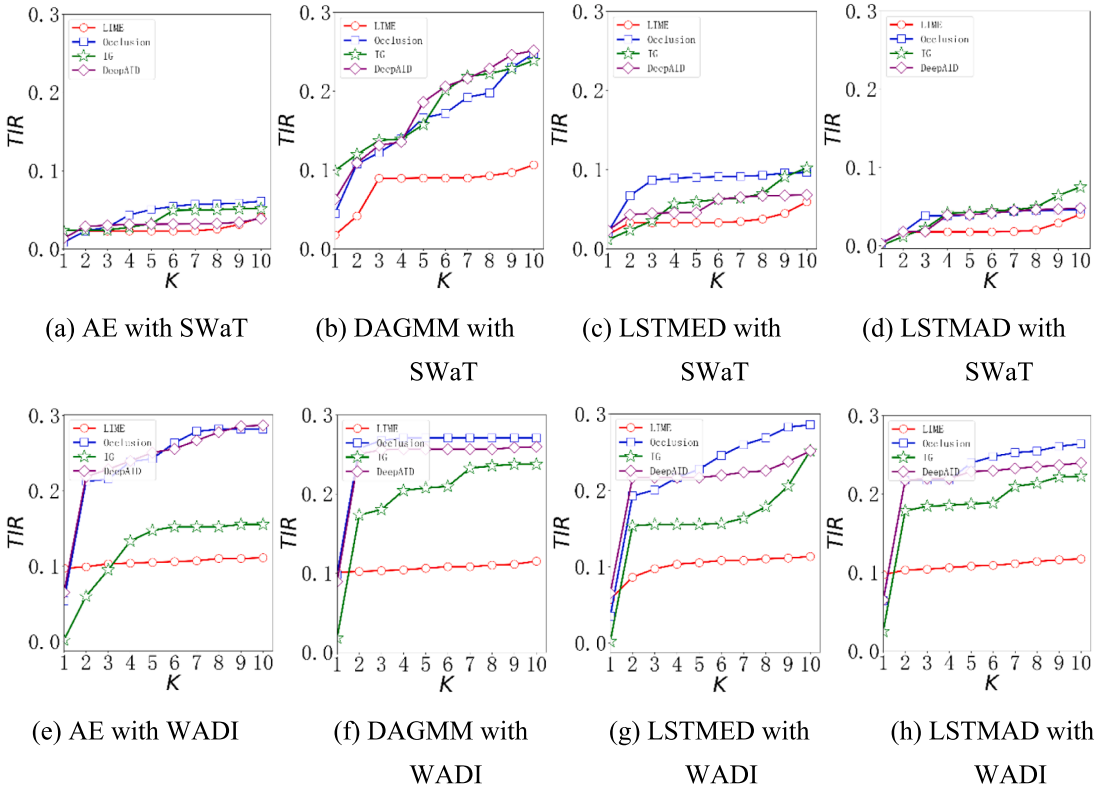


Fig. 6. The impacts of dimensions K in interpretation results on the performance of interpretability with baselines. The x-axis represents the dimensions K in interpretation results, and the y-axis represents the joint evaluation of interpretation performance *TIR*. The subfigures (a)-(d) represent the comparison of interpretation algorithms using four detection algorithms on SWaT dataset, and the subfigures (e)-(h) represent the comparison of that on WADI dataset.

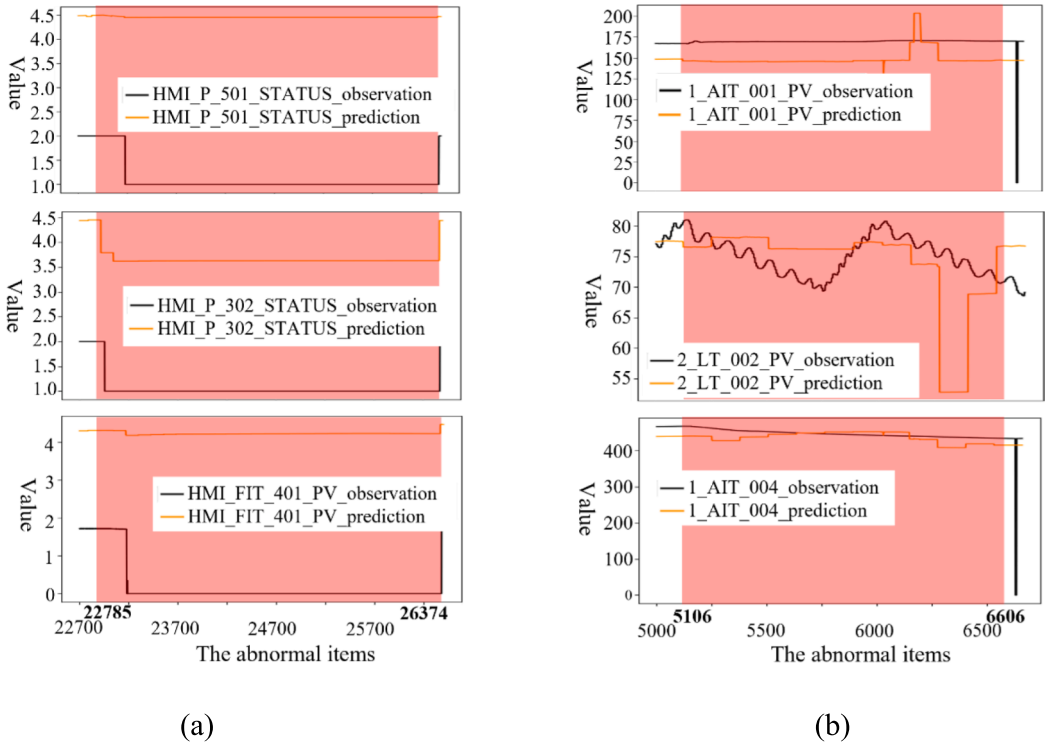


Fig. 7. Observations and predictions of attacked and affected components. The subfigure (a) represents the trend of attacked and affected components on the SWaT dataset, and the subfigure (b) represents the trend of that on the WADI dataset in red shade. The x-axis represents the range of abnormal items, and the y-axis represents observations and predictions of three typical components, i.e., HMI_P_501, HMI_P_302, and HMI_FIT_401_PV. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

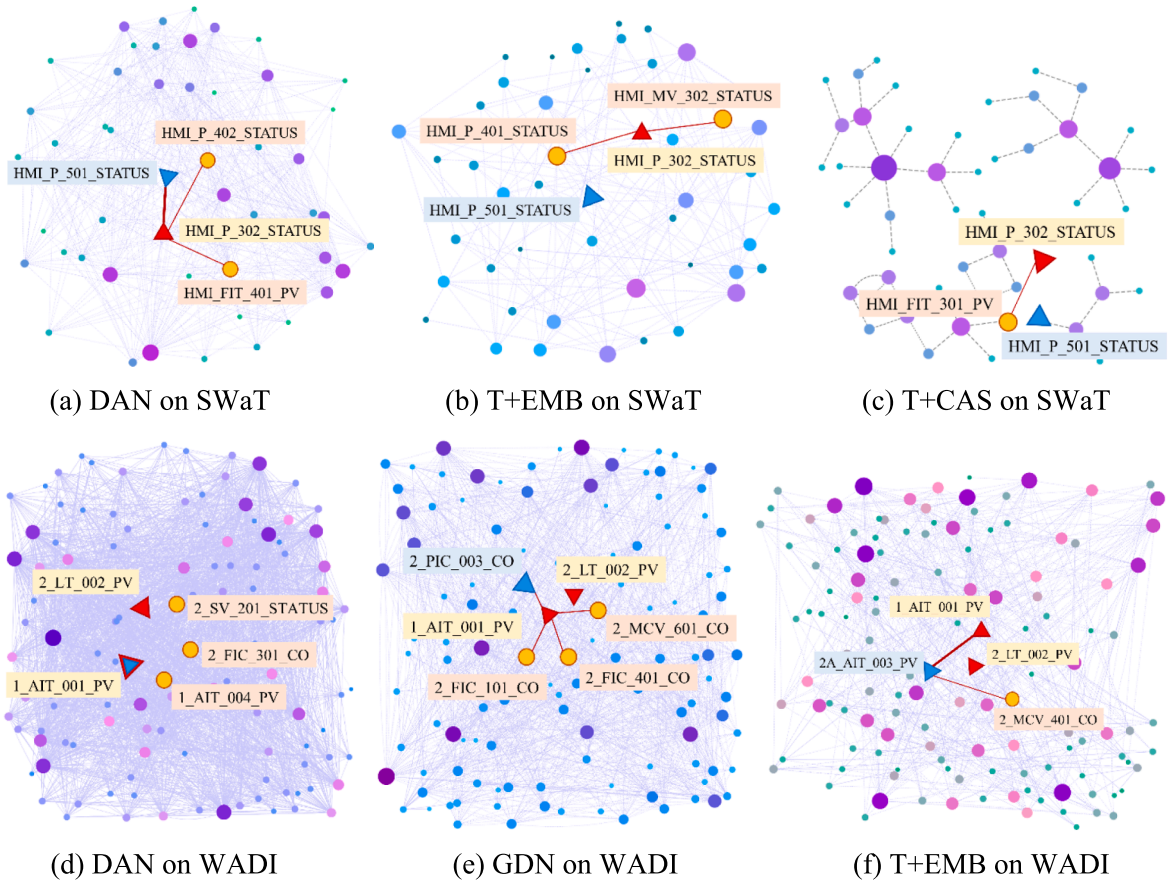


Fig. 8. Comparison of graphical interpretation results on SWaT and WADI datasets. The figures (a)-(c) represent graphical interpretation results on the SWaT dataset using DAN and baselines, i.e., T + EMB, T + CAS, and the figures (d)-(f) represent that on the WADI dataset using DAN and baselines, i.e., GDN, and T + EMB.

attacked component. The yellow dots represent the components having a stronger correlation with the attacked component and are the tail nodes of the node corresponding to the attacked component in the Graph.

As shown in Fig. 8(a), DAN correctly interprets that "HMI_P_501" with the highest outlier score (trend during attack phase shown as Fig. 7 (a)), having a stronger correlation with attacked component "HMI_P_302", "HMI_P_402" labeled "HMI_P_402_STATUS" and "HMI_401_PV" labeled "HMI_FIT_401_PV" are further recognized as two components having the strong correlation with attacked component. The above interpretation is consistent with the real attack scenario on the SWaT dataset. GDN finds that "HMI_PIT_501_PV", with the highest outlier score, has no relationship with the actually attacked component in Fig. 1(b). Similarly, Fig. 8(b) and (c) indicate that neither T + EMB nor T + CAS can give a reasonable interpretation.

By similar analysis with SWaT, shown in Fig. 8(d), 8(e), and 8(f), it is observed that the ability of DAN, T + EMB are better than that of GDN in terms of interpreting attacked "1_AIT_001_PV" and "2_LT_002_PV" with WADI dataset. Concretely speaking, DAN interprets "1_AIT_001_PV" as attacked, consistent with the attack scenario on WADI. In addition, as described in the attack result, "1_AIT_004_PV" is observed as 0 at 6606 after attacking in Fig. 7(b). T + EMB considers "2_AIT_003_PV" has a stronger correlation with attacked components "1_AIT_001_PV". While T + CAS can not detect the attack against "2_LT_002_PV" and "1_AIT_001_PV". "2_PIC_003_CO" has no relationship with attacked components in GDN.

6. Conclusion

In this paper, we have proposed DAN, a neural network based on dual attention, in which causality between different types of device components is learned by a temporal convolution based on attention and GCN based on attention, for detecting and interpreting attacks against ICS simultaneously.

To resolve problems caused by improper representation of graph nodes, DAN directly represents the time series after causality learning as graph nodes, and later leverages a GCN based on attention to predict series for guaranteeing detection precision, and finally locate attacked components by calculating attention weights from graph attention network. Experimental results obtained using two acknowledged datasets extracted from test beds highly simulating actual water plants, demonstrate the effectiveness of DAN.

There are primarily two limitations to this study. First, we do not solve the issue that the difficulty of interpreting discrete types of components such as valves, switches, etc. which reduces the accuracy of existing interpretation methods. Second, the industrial control scenario used in our research is relatively changeless, in some cases, such as when the industrial control architecture or components change, the scene is no longer stable, leading to that the corresponding established graph neural network model is no longer applicable to the current scene, and the model needs to be retrained. In the future, how to use the sparse attention mechanism to deduce the real attacked discrete component from the continuous variable type component, how to use the idea of incremental learning to build a deep learning model with low resource consumption and fast update speed to adapt to the ever-changing industrial control network environment is worthy of in-depth research. There are other aspects we intend to explore in the future. These include locating attacked components accurately when multiple components are attacked at the same time, discovering the attack degree, as well as promoting the performance of anomaly detection on multivariate time

series by leveraging interpretation results.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported in part by the National Key R&D Program of China (2023YFB3107305), in part by the Young Innovation Team of Colleages and Universities in Shandong Province (2021JK001), in part by the National Natural Science Foundation of China(62172244), in part by the Natural Science Foundation of Shandong Province(ZR2020YQ06 and ZR2021MF132), in part by the Innovation Ability Pormotion Project for Small and Medium sized Technology-based Enterprise of Shandong Province (2022TSGC2098), in part by the Pilot Project for Integrated Innovation of Science, Education and Industry of Qilu University of Technology (Shandong Academy of Sciences) (2022JBZ01-01), in part by the Taishan Scholars Program (tsqn202211210).

Data availability

The data that has been used is confidential.

References

Ahmed, C. M., Palleti, V. R., & Mathur, A. P. (2017). *WADI: a water distribution testbed for research in the design of secure cyber physical systems*. In Proceeding of the 3rd International Workshop on Cyber-Physical Systems for Smart Water Networks, Pittsburgh, Pennsylvania, USA.

Bai, S., Kolter, J. Z., & Koltun, V. (2018). *An empirical evaluation of generic convolutional and recurrent networks for sequence modeling.* In Proceeding of International Conference on Learning Representations (ICLR) Workshop, Vancouver, BC, Canada.

Bhatt, U., Ravikumar, P., & Moura, J. (2019). *Towards aggregating weighted feature attributions*. In Proceeding of the AAAI Workshop on Network Interpretability for Deep Learning, Hawaii, USA.

Carcano, A., Coletta, A., & Guglielmi, M. (2011). A multidimensional critical state analysis for detecting intrusions in scada systems. *IEEE Transactions on Industrial Informatics, 7*, 179–186. https://doi.org/10.1109/TII.2010.2099234

Casajús-Setién, J., Bielza, C., & Larrañaga, P. (2022). *Evolutive adversarially-trained bayesian network autoencoder for interpretable anomaly detection*. In Proceeding of the 11th International Conference on Probabilistic Graphical Models, Almeria, Spain.

Chen, C., Li, O., Barnett, A., Su, J., & Rudin, C. (2019). *This looks like that: deep learning for interpretable image recognition*. In Proceeding of the Conference and Workshop on Neural Information Processing Systems. Vancouver Canada.

Chen, W., Tian, L., Chen, B., Dai, L., Duan, Z., & Zhou, M. (2022). *Deep variational graph convolutional recurrent network for multivariate time series anomaly detection.* In Proceedings of the 39th International Conference on Machine Learning. Baltimore, Maryland, USA.

Chen, Z., Chen, D., Yuan, Z., Cheng, X., & Zhang, X. (2021). Learning graph structures with transformer for multivariate time series anomaly detection in iot. *IEEE Internet of Thing Journal*, 9, 9179–9189. https://doi.org/ 10.1109/JIOT.2021.3100509.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Y. Bengio. (2014). *Learning phrase representations using RNN encoder-decoder for statistical machine translation*. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Doha, Qatar.

Deng, A., & Hooi, B. (2021). *Graph neural network-based anomaly detection in multivariate time series*. In Proceeding of AAAI Conference on Artificial Intelligence, Vancouver, Canada.

Ding, C., Sun, S., & Zhao, J. (2023). MST-GAT: A multimodal spatial–temporal graph attention network for time series anomaly detection. *Information Fusion, 89*, 527–536. https://doi.org/10.1016/j.inffus.2022.08.011

Dwivedi, R., Dave, D., Naik, H., Singhal, S., Rana, O., Patel, P., Qian, B., Wen, Z., Shah, T., & Morgan, G. (2022). Explainable AI (XAI): core ideas, techniques and solutions. *ACM Computing Surveys*, 55, 1–33. https://doi.org/ 10.1145/3561048.

Gao, C., Zhu, J., Zhang, F., Wang, Z., & Li, X. (2023). A novel representation learning for dynamic graphs based on graph convolutional networks. *IEEE Transactions on Cybernetics, 53*, 3599–3612. https://doi.org/10.1109/TCYB.2022.3159661

Gao, C., Liu, H., Huang, J., Wang, Z., Li, Z., & Li, X. (2024). Regularized spatial-temporal graph convolutional networks for metro passenger flow prediction. *IEEE Transactions on Intelligent Transportation Systems*. Early access. https://doi.org/10.1109/tits.2024.3365179.

Geiger, A., Liu, D., Alnegheimish, S., Cuesta-Infante, A., & Veeramachaneni, K. (2020). December). *TadGAN: Time series anomaly detection using generative adversarial networks. In Proceeding of IEEE International Conference on Big Data, virtual*.

Giurgiu, I., & Schumann, A. (2019). *Additive explanations for anomalies detected from multivariate temporal data*. In Proceeding of the 28th ACM International Conference on Information and Knowledge Management. New York, NY, USA.

Guo, W., Mu, D., Xu, J., Su, P., Wang, G., & Xing, X. (2018). *LEMNA: Explaining deep learning based security applications*. In Proceeding of the 2018 ACM SIGKDD Conference on Computer and Communications Security, Toronto, Canada.

Han, D., Wang, Z., Chen, W., Zhong, Y., Wang, S., Zhang, H., Yang, J., Shi, X., & Yin, X. (2021). *DeepAID: Interpreting and improving deep learning-based anomaly detection in security applications*, In Proceeding of ACM SIGSAC Conference on Computer and Communications Security, New York, NY, USA.

Han, S., & Woo, S. S. (2022). *Learning sparse latent graph representations for anomaly detection in multivariate time series*. In Proceeding of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington DC, U.S.

Hawkins, S., He, H., Williams, G., & Baxter, R. (2002). *Outlier detection using replicator neural networks*. In Proceeding of the International Conference on Data Warehousing and Knowledge Discovery, Aix-en-Provence, France.

Hundman, K., Constantinou, V., Laporte, C., Colwell, I., & Soderstrom, T. (2018). *Detecting spacecraft anomalies using lstms and nonparametric dynamic thresholding*, In Proceedings of the 24th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, London, UK.

Ikeda, Y., Tajiri, K., Nakano, Y., Watanabe, K., & Ishibashi, K. (2019). *Estimation of dimensions contributing to detected anomalies with variational autoencoders*, In Proceeding of the AAAI Workshop on Network Interpretability for Deep Learning, Hawaii, USA.

Kauffmann, J., Müller, K. R., & Montavon, G. (2020). Towards explaining anomalies: A deep taylor decomposition of one-class models. *Pattern Recognition, 101*, Article 107198. https://doi.org/10.1016/j.patcog.2020.107198

Lai, G., Chang, W. C., Yang, Y., & Liu, H. (2018). *Modeling long- and short-term temporal patterns with deep neural networks*. In Proceeding of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor USA.

Letham, B., Rudin, C., Mccormick, T. H., & Madigan, D. (2015). Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *Annals of Applied Statistics, 9*, 1350–1371.

Li, D., Chen, D., Jin, B., Shi, L., Goh, J., & Ng, S.-K. (2019). *MAD-GAN: Multivariate anomaly detection for time series data with generative adversarial networks*. In Proceeding of the 28th International Conference on Artificial Neural Networks, Munich.

Li, W., Hu, W., Chen, T., Chen, N., & Feng, C. (2021). Stacking VAE with graph neural networks for effective and interpretable time series anomaly detection. *AI Open, 3*, 101–110. https://doi.org/10.1016/j.aiopen.2022.07.0001

Li, W., Xie, L., Deng, Z., & Wang, Z. (2016). False sequential logic attack on ACADA system and its physical impact analysis. *Computers & Security, 58*, 149–159. https://doi.org/10.1016/j.cose.2016.01.001

Liu, N., Shin, D., &, Xia, H. (2018). *Contextual outlier interpretation*. In Proceeding of the 27th International Joint Conference on Artificial Intelligence, Stockholm, Sweden.

Lopes, D. D., Cunha, B. R. D., Martins, A. F., Gonalves, S., Lenzi, E. K., Hanley, Q. S., Perc, M. & Ribeiro, H. V. (2022). Machine learning partners in criminal networks. *Scientific Reports*. 12, Article 15746. https://doi.org/10.1038 /s41598-022-20025-w.

Lundberg, S. M., & Lee, S. I. (2017). *A unified approach to interpreting model predictions*. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach, California, USA.

Mahapatra, D., Poellinger, A., & Reyes, M. (2023). Graph node based interpretability guided sample selection for active learning. *IEEE transactions on medical imaging*, 42, 661-673. https://doi.org/ 10.1109/TMI.2022.3215017.

Malhotra, P., Ramakrishnan, A., Anand, G., Vig, L., Agarwal, P., & Shroff, G. (2016). *Lstm-based encoder-decoder for multi-sensor anomaly detection*. Anomaly Detection Workshop at 33rd International Conference on Machine Learning (ICML 2016), New York, NY.

Malhotra, P., Vig, L., Shroff, G., & Agarwal, P. (2015). *Long short term memory networks for anomaly detection in time series*. In Proceeding of the 23rd European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges, Belgium.

Mathur, A. P., & Tippenhauer, N. O. (2016), *SWaT: a water treatment testbed for research and training on ics security*. In Proceeding of the 2016 International Workshop on Cyber-physical Systems for Smart Water Networks (CySWater), Vienna, Austria.

Ribeiro, M., Sameer, S., Carlos, G. (2016). *Why shoud i trust you?: Explaining the predictions of any classifier*. In proceeding of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco USA.

Ribeiro, H. V., Lopes, D. D., Pessa, A. B., Martins, A. F., Cunha, B. R., Gonçalves, S., Lenzi, E. K., Hanley, Q. S., & Perc, M. (2023). Deep learning criminal networks. *Chaos, Solitons and Fractals: Applications in Science and Engineering: An Interdisciplinary Journal of Nonlinear Science., 172*, Article 113579. https://doi.org/10.1016/j.chaos.2023. 113579

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, Gold Coast, Australia.

Shrikumar, A., Greenside, P., & Kundaje, A. (2017). *Learning important features through propagating activation differences*. In Proceeding of the International Conference on Machine Learning, Sydney, Australia.

Siffer, A., Fouque, P. A., Termier, A., & Largouet, C. (2017). *Anomaly detection in streams with extreme value theory*. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax NS, Canada.

Sigaki, H. Y. D., Lenzi, E. K., Zola, R. S., Perc, M., & Ribeiro, H. V. (2020). Learning physical properties of liquid crystals with deep convolutional neural networks. *Scientific Reports, 10*, 1–10. https://doi.org/10.1038/s41598-020-63662-9

Simonyan, K., Vedaldi, A., & Zisserma, A. (2014). *Deep inside convolutional networks visualising image classification models and saliency maps*. In Proceeding of the 2th International Conference on Learning Representations, Banff National Park, Canada.
Sundararajan, M., Taly, A., & Yan, Q. (2017). *Axiomatic attribution for deep networks*. In Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia.
Surucu, M., Isler, Y., Perc, M., & Kara, R. (2021). Convolutional neural networks predict the onset of paroxysmal atrial fibrillation: Theory and applications. *Chaos: An Interdisciplinary Journal of Nonlinear Science, 31*, Article 113119. https://doi.org/10.1063/5.0069272
Wu, H., Xu, J., Wang, J., & Long, M. (2021). *Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting*. In Proceeding of the 35th Conference on Neural Information Processing Systems, Montreal, Canada.
Wu, Z., Pan, S., Long, G., Jiang, J., Chang, X., & Zhang, C. (2020). *Connecting the dots: Multivariate time series forecasting with graph neural networks*. In Proceeding of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, New York, NY, USA.
Xu, H., Wang, Y., Jian, S., Huang, Z., Wang, Y., Liu, N., & Li, F. (2021a). *Beyond outlier detection: Outlier interpretation by attention-guided triplet deviation network*. In Proceeding of the International Conference of World Wide Web, Ljubljana Slovenia.
Xu, L., Wang, B., Wu, X., Zhao, D., Zhang, L., & Wang, Z. (2021b). Detecting semantic attack in scada system: A behavioral model based on secondary labeling of states-duration evolution graph. *IEEE Transactions on Network Science and Engineering, 9*, 703–715. https://doi.org/10.1109/TNSE.2021.3130602
Xu, L., Wang, B., Yang, M., Zhao, D., & Han, J. (2021c). Multi-mode attack detection and evaluation of abnormal states for industrial control network. *Journal of Computer Research and Development, 58*, 2333–2349. https://doi.org/10.7544/issn1000-1239.2021.20210598
Xu, J., Wu, H., Wang, J., & Long, M. (2022). *Anomaly Transformer: Time series anomaly detection with association discrepancy*. In Proceeding of the International Conference on Learning Representations, virtual.
Yang, D., Usynin, A., & Hines, J. W. (2006). *Anomaly-based intrusion detection for SCADA systems.* In Proceeding of the 5th International Topical Meeting on Nuclear Plant Instrumentation, Control and Human Machine Interface Technologies, Illinois, United States.
Yang, L., Guo, W., Hao, Q., Ciptadi, A., Ahmadzadeh, A., Xing, X., & Wang, G. (2021). *CADE: Detecting and explaining concept drift samples for security applications.* In Proceeding of the 30th USENIX Security Symposium, Vancouver, B.C., Canada.
Zeiler, M., & Fergus, R. (2014). *Visualizing and understanding convolutional networks*. In Proceeding of the 13th European Conference on Computer Vision, Zurich, Switzerland.
Zhang, X., Marwah, M., Lee, I. T., Arlitt, M., & Goldwasser, D. (2019). *An anomaly contribution explainer for cyber-security applications*. In Proceeding of 2019 IEEE International Conference on Big Data, Los Angeles, CA, USA.
Zhu, J., An, P., & Wan, M. (2018). Intrusion detection method of RST-SVM for abnormal behavior in industrial control network. *Journal of Electronic Measurement and Instrumentation, 32*, 8–14.
Zintgraf, L. M., Cohen, T. S., Adel, T., & Welling, M. (2017). *Visualizing deep neural network decisions: Prediction difference analysis*. In Proceeding of the 5th International Conference on Learning Representations, Toulon France.
Zong, B., Song, Q., Min, M. R., Cheng, W., Lumezanu, C., Cho, D., & Chen, H. (2018). *Deep autoencoding gaussian mixture model for unsupervised anomaly detection*. In Proceeding of the 6th International Conference on Learning Representations, Vancouver, BC, Canada.

LijuanXu received Ph.D. degree from the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China, in 2023. She is currently an Associate Professor with Shandong Computer Science Center (National Supercomputer Center in Jinan), China. Her main research interests include network security, industrial internet security, and computer forensics.

Bailing Wang received his Ph.D. degree from the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China, in 2006. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology. His main research interests include financial security, information security, and cyber security.

DaweiZhao received the Ph.D. degree in cryptology from the Beijing University of Posts and Telecommunications, Beijing, China, in 2014. He is currently a Professor with Shandong Computer Science Center (National Supercomputer Center in Jinan), China. His main research interests include network security, complex network, and epidemic spreading dynamics.

XiaomingWu received the Ph.D. degree in software engineering from Shandong University of Science and Technology, Qingdao, China, in 2017. He is currently a Professor with Shandong Computer Science Center (National Supercomputer Center in Jinan), China. His main research interests include network security, industrial internet security, and wireless sensor network.