Causal Graph Profiling via Structural Divergence for Robust Anomaly Detection in Cyber-Physical Systems

Arun Vignesh Malarkkan
arun.malarkkan@asu.edu
Arizona State University
Tempe, Arizona, USA

Haoyue Bai
haoyueba@asu.edu
Arizona State University
Tempe, Arizona, USA

Dongjie Wang
wangdongjie@ku.edu
University of Kansas
Lawrence, Kansas, USA

Yanjie Fu[†]
yanjie.fu@asu.edu
Arizona State University
Tempe, Arizona, USA

Abstract

With the growing complexity of cyberattacks targeting critical infrastructures such as water treatment networks, there is a pressing need for robust anomaly detection strategies that account for both system vulnerabilities and evolving attack patterns. Traditional methods—statistical, density-based, and graph-based models struggle with distribution shifts and class imbalance in multivariate time series, often leading to high false positive rates. To address these challenges, we propose CGAD: a Causal Graph-based Anomaly Detection framework designed for reliable cyberattack detection in public infrastructure systems. CGAD follows a two-phase supervised framework: causal profiling and anomaly scoring. First, it learns causal invariant graph structures representing the system's behavior under "Normal" and "Attack" states using Dynamic Bayesian Networks. Second, it employs structural divergence to detect anomalies via causal graph comparison by evaluating topological deviations in causal graphs over time. By leveraging causal structures, CGAD achieves superior adaptability and accuracy in non-stationary and imbalanced time series environments compared to conventional machine learning approaches. By uncovering causal structures beneath volatile sensor data, our framework not only detects cyberattacks with markedly higher precision but also redefines robustness in anomaly detection, proving resilience where traditional models falter under imbalance and drift. Our framework achieves substantial gains in F1 and ROC-AUC scores over best-performing baselines across four industrial datasets, demonstrating robust detection of delayed and structurally complex anomalies.

1 Introduction

Critical public infrastructures—including transportation systems, energy grids, water treatment facilities, healthcare services, and communication networks—form the backbone of societal functionality, public safety, and economic stability. These infrastructures are increasingly vulnerable to cyberattacks, which can lead to cascading service disruptions, severe economic damage, and threats to public health and safety. For example, in 2021, a cyberattack on the Oldsmar, Florida water treatment plant attempted to manipulate chemical levels, exposing critical vulnerabilities and prompting widespread, costly security upgrades across the sector [6]. Robust cyberattack detection in such infrastructures is essential to mitigate immediate threats and preserve system integrity. Yet, this task is particularly challenging due to the high-dimensional, temporally dependent nature of sensor data and severe class imbalance, where malicious behavior is rare compared to normal operations. These challenges are magnified in cyber-physical domains like industrial control systems and water treatment networks, where distributed sensors, dynamic operational contexts, and evolving attack surfaces complicate reliable anomaly detection. In this paper, we focus on *Water Treatment Networks (WTNs)*—a critical class of industrial cyber-physical infrastructure that manages water purification and distribution through interconnected sensors, actuators, and communication modules. WTNs generate *multivariate, high-dimensional time-series data* that reflect complex dependencies among physical and cyber components, making them a realistic and challenging testbed for cyberattack detection. We formalize the task of cyberattack detection in WTNs as the identification and classification of anomalous segments in multivariate time-series data collected from distributed sensors, using historical attack labels as supervision signals. The goal is to develop models that can detect both known and unseen cyberattacks with high precision and robustness.

Addressing cyberattack detection within the Water Treatment Networks critically introduces two seminal research challenges:
C1: Modeling invariant anomaly influence structures: How can we learn a robust representation of inter-sensor influence structures that remain invariant to attack strategies and class imbalance?
C2: Fast anomaly scoring: Given the learned structure, how can we efficiently quantify the degree of anomalous deviation in incoming time segments with training or supervision?

In defending WTNs against cyberattacks, prior literature includes both research-driven and applied anomaly detection approaches, each with specific limitations. **Applied techniques** focus on efficiency and simplicity: *1) Knowledge-based detection* uses expert-defined thresholds (e.g., chlorine levels, valve pressure), but lacks adaptability and scalability [30]. *2) Statistics-based detection* employs probabilistic or distance-based thresholds, yet struggles with nonlinear dependencies and manual tuning [2, 23]. *3) Unsupervised detection* methods such as clustering and anomaly scoring (e.g., k-means, Isolation Forests) scale well but suffer from high false positives due to limited context [5, 7]. *Supervised detection*

[†] Corresponding author.

Conference'17, July 2017, Washington, DC, USA
2018. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/XXXXXXX.XXXXXXX

methods like SVMs and random forests require labeled data and generalize poorly to novel attacks [7]. **Research-oriented methods** emphasize modeling complexity and structural reasoning: *4) Deep learning and graph-based detection* (e.g., LSTMs, GNNs) offer expressive temporal and relational modeling [4, 17, 21, 34], but suffer from high computational cost, limited interpretability, and reliance on heuristic graph construction. *5) Causal graph fusion approaches* (e.g., SMV-CGAD) combine dense and sparse views for improved robustness [18–20], but require domain priors and use deep graph classifiers that reduce transparency.

Our Insights: a causal graph-enabled profiling-scoring perspective. While prior work has explored correlation- or signal-based graph structures, few approaches model the causal mechanisms underlying WTN operations. We argue that **causal graphs**, represented as **Directed Acyclic Graphs (DAGs)**, offer a principled way to capture stable normal and anomaly dependency structures that remain invariant across shifting operational conditions and evolving attacks. This angle enhances robustness to distribution shifts while offering actionable insights by elucidating the propagation of anomalies across system components.

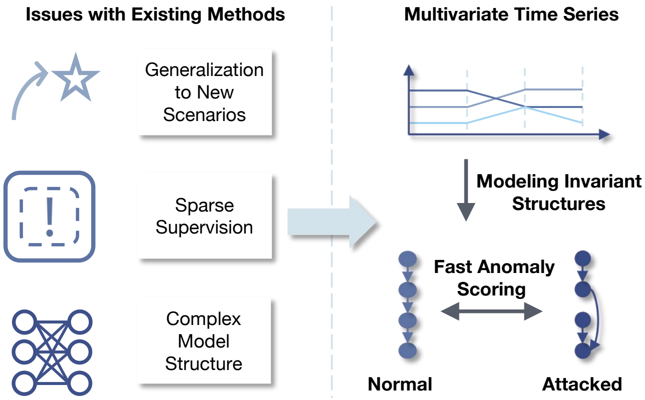


Figure 1: Leveraging causal structural differences between normal and attack conditions in WTNs for superior anomaly detection compared to existing traditional methods.

Summary of Proposed Method: CGAD – A Causal Profiling-Scoring Framework. To this end, we propose **CGAD**, a causal graph-based framework for anomaly detection from multivariate sensor time series. CGAD has two main components: 1) **Causal Anomaly Profiling:** We learn two causal DAGs to represent the normal and anomalous system states using observational time-series data. 2) **Causal Deviation Scoring:** We develop a DAG-to-DAG distance metric to compute an anomaly score for test data segments. Specifically, if the causal structure of a test segment closely resembles the anomaly DAG, it is flagged as anomalous; otherwise, it is considered normal. Our DAG-DAG distance metric accounts for both structural topology (e.g., edge presence/absence) and causal strength (e.g., edge weights), providing a holistic and efficient way to measure deviation between causal models. This framework enables fast, accurate, and robust detection of cyberattacks and is designed to be robust to class imbalance and distribution shifts commonly observed in real-world datasets.

Our Contributions. We address the pressing problem of cyberattack detection in Water Treatment Networks using multivariate sensor time series. Our main contributions are:

- We introduce **CGAD**, a novel causal graph-based framework that explicitly models and compares normal and abnormal causal structures in WTN data.
- We formulate a **two-phase profiling-scoring pipeline** that first learns DAGs to represent system states and then computes a **causal divergence score** via DAG-DAG comparison.
- We demonstrate that CGAD provides **robust and accurate detection** of cyberattacks in WTNs, significantly improving performance over existing baselines.

2 Problem Statement

Water Treatment Network (WTN): Consider a water treatment network W instrumented with K sensors monitoring various treatment process stages [33]. Continuous sensor streams are partitioned into N non-overlapping intervals, each with M time-aligned measurements, yielding a sensor stream data sequence $X = [X_1, \ldots, X_N]$, where $X_i \in \mathbb{R}^{M\times K}$. Each segment X_i is labeled by $y_i \in \{0, 1\}$, with $y_i = 1$ denoting an attack and $y_i = 0$ as normal operation.

The Detection Task: Let the multivariate time series data be segmented into N non-overlapping intervals, each denoted by $X_i \in \mathbb{R}^{M\times K}$, where M is the number of time steps per segment and K is the number of sensors. The dataset is given by $D = \{(X_1, y_1), \ldots, (X_N, y_N)\}$, where $y_i \in \{0, 1\}$ indicates whether segment X_i corresponds to an *Attack* (1) or *Normal* (0) system state. The AI task is to learn a model that can detect cyberattacks in WTN from multivariate time series. The model must account for both immediate disruptions and delayed attack effects, ensuring robust detection across short-term and delayed structural deviations.

Objective: We propose a robust, efficient framework for segment-level cyberattack detection via *structural causal learning*. By modeling inter-sensor causal links, our method detects deviations from normal causal dynamics, enhancing generalization to novel attacks.

3 Proposed Method

3.1 Framework Overview

Figure 2 illustrates the architecture of our proposed CGAD framework, which comprises two core phases:

(P1) Causal Profiling: We learn invariant causal structures from multivariate sensor time series by estimating Dynamic Bayesian Networks (DBNs) representing the system under distinct operational states. Specifically, two causal graphs are constructed: one for *Normal* operation and one for *Attack* conditions. This profiling captures stable, structurally insightful inter-sensor dependencies that reflect underlying system dynamics.

(P2) Anomaly Scoring: Streaming data is segmented temporally, and for each segment, a causal graph is inferred using the same procedure. We then compute the Structural Hamming Distance (SHD) between the segment's graph and each of the two reference graphs. The segment is classified as *Attack* if its causal structure is closer (lower SHD) to the *Attack* causal graph, and *Normal* otherwise.

This two-phase causal profiling and scoring pipeline leverages robust causal representations to detect anomalies efficiently and with increased transparency. Unlike traditional correlation-based

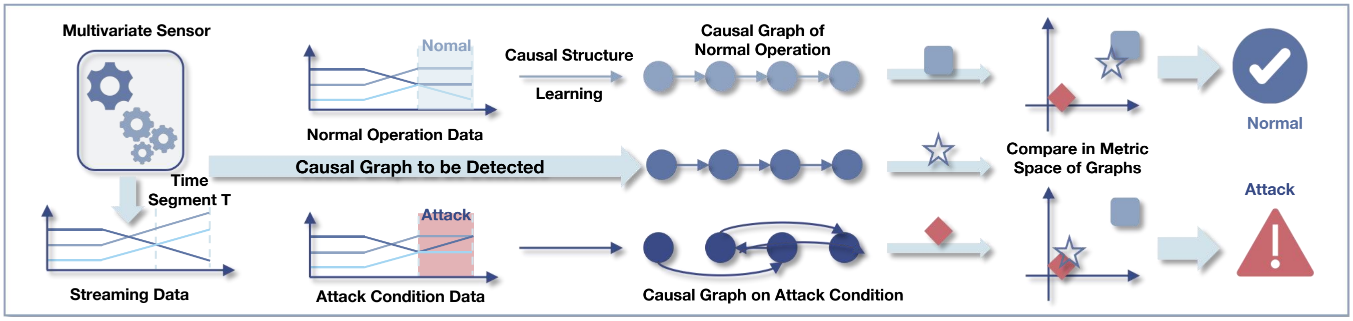

Multivariate Sensor
Time Segment T
Streaming Data
Nomal
Normal Operation Data
Causal Structure Learning
Causal Graph of Normal Operation
Causal Graph to be Detected
Attack
Attack Condition Data
Causal Graph on Attack Condition
Compare in Metric Space of Graphs
Normal
Attack

Figure 2: Overview of the CGAD Framework for Cyberattack Detection. Phase 1 - Causal Graph Learning: Causal Profiling learns causal graphs for "Normal" and "Attack" states using DYNOTEARS algorithm. Phase 2 - Anomaly Detection via Graph Comparison: Anomaly Scoring segments data, infers a graph for each segment, and classifies segments based on Structural Hamming Distance (SHD) to reference graphs.

methods, CGAD explicitly models causal mechanisms, enabling resilient detection across diverse attack patterns and operational shifts. Although only a single attack graph is used as a ground-truth, it captures recurring structural disruptions that are common across different attacks, allowing CGAD to generalize effectively to unseen threats through structural graph comparisons.

3.2 Causal Profiling - Causal DAG Learning for Time-Series Data

In this phase, we uncover the causal structure of multivariate sensor data to distinguish genuine anomalies from spurious correlations. Accurate modeling of these structures is critical, as cyberattacks in Water Treatment Networks (WTNs) often manifest through temporally extended and structurally coherent disruptions. To capture both instantaneous and time-lagged dependencies, we model the system as a Dynamic Bayesian Network (DBN), represented as a time-aware Directed Acyclic Graph (DAG). In this DAG, each **node** corresponds to a specific sensor reading (e.g., valve pressure, flow rate, chemical concentration) at a given time step. **Edges** represent directed causal influence: an edge from node A_t to node B_{t+1} implies that the value of sensor A at time t has a causal effect on sensor B at the next time step $t+1$. Intra-slice edges model instantaneous dependencies within a segment, while inter-slice edges capture delayed or sequential effects across time. Together, the DAG encodes the underlying operational rules and inter-sensor interactions governing system behavior. We employ the **DYNOTEARS** algorithm [25], a state-of-the-art approach for learning DBNs from time-series data. Unlike static DAG learning methods such as NOTEARS [42], DYNOTEARS explicitly incorporates temporal structure and can effectively model delayed, persistent, and cascading effects that characterize real-world cyberattacks [33]. Using this approach, we construct two causal graphs: one representing the normal operating state (G_{Normal}) and the other capturing the structural footprint of anomalous behavior (G_{Attack}). These graphs serve as causal profiles for the scoring phase, enabling structure-aware detection of deviations in unseen data segments.

Let $\mathcal{X} = [X_1, X_2, \ldots, X_n]$ denote the multivariate time-series data, segmented into n non-overlapping time windows. Each segment $X_i \in \mathbb{R}^{M\times K}$ contains M time steps across K sensors in a WTN. We design the temporal layout such that anomalous segments are bounded by normal segments both before and after the attack interval. Formally, we define the sequence as:

$$X_i = [N_1, N_2, A_3, A_4, \ldots, A_{m-2}, N_{m-1}, N_m],$$

where N_j denotes a segment in a *Normal* state, and A_j denotes a segment under an *Attack* state.

This structured segmentation allows us to construct two ground-truth causal graphs: G_{Normal} and G_{Attack}, which characterize inter-sensor dependencies under normal and attack conditions, respectively. To learn these graphs, we apply the **DYNOTEARS** algorithm [25], an extension of NOTEARS [42], designed to uncover both inter-slice (temporal lag) and intra-slice (instantaneous) causal dependencies from time-series data.

Let $X \in \mathbb{R}^{T\times K}$ denote the design matrix formed by concatenating the sensor readings from all segments within a class (either normal or attack), and let $Y \in \mathbb{R}^{T\times K}$ denote the corresponding time-lagged matrix. The Structural Equation Model (SEM) is defined as:

$$X = XW + YA + Z, \tag{1}$$

where $W \in \mathbb{R}^{K\times K}$ is the intra-slice (instantaneous) adjacency matrix, $A \in \mathbb{R}^{K\times K}$ is the inter-slice (temporal) adjacency matrix, and Z is a residual noise matrix.

The goal is to estimate sparse, interpretable matrices W and A such that W defines an acyclic structure. This constrained optimization objective is achieved by:

$$\min_{W,A} f(W, A) \quad \text{s.t.} \quad h(W) = 0, \tag{2}$$

with:

$$f(W, A) = \ell(W, A) + \lambda_W \|W\|_1 + \lambda_A \|A\|_1, \tag{3}$$

where $\ell(W, A)$ denotes the squared loss term, and λ_W, λ_A are hyperparameters controlling the sparsity via ℓ_1-norm regularization. To enforce acyclicity of the intra-slice graph, DYNOTEARS uses the continuous constraint:

$$h(W) = \text{tr}(e^{W\circ W}) - d = 0, \tag{4}$$

where $\circ$ denotes the Hadamard (element-wise) product and $d = K$ is the number of nodes in the graph.

At the end of this phase, we obtain the two causal graphs—G_{Normal} and G_{Attack}—which serve as reference structures for anomaly detection via graph comparison in the subsequent scoring phase. During training, we learn two reference graphs, G_{Normal} and G_{Attack}, by applying DYNOTEARS to concatenated normal and attack segments, respectively. At inference time, we apply DYNOTEARS independently to each unseen segment X_t to construct a corresponding causal graph G_t, which is then compared to the reference graphs using structural divergence for anomaly scoring.

3.3 Anomaly Scoring via Structural Divergence

Causal Graph Construction: In this phase, we perform causal profiling by learning a causal graph for each time segment of the test data and comparing it to reference causal graphs representing normal and attack states. This enables time-segmented anomaly detection, addressing limitations of traditional point-level detection methods. Specifically, point-level approaches are prone to false positives in imbalanced settings, overly sensitive to benign signal fluctuations, and often fail to detect temporally delayed or system-wide attack effects. In contrast, our causal graph-based strategy captures context-aware structural dependencies, allowing for more robust and interpretable anomaly detection. For each test segment T, we apply the DYNOTEARS algorithm to infer the segment-specific causal graph G_T, representing both instantaneous and lagged inter-sensor relationships within that segment.

Structural Comparison: Given the learned causal graph G_T for a test segment, we assess its similarity to the reference graphs G_{Normal} and G_{Attack} using a principled graph-theoretic metric: the Structural Hamming Distance (SHD). SHD quantifies the dissimilarity between two DAGs as the minimum number of edge additions, deletions, or direction reversals between them. This allows us to robustly measure deviations in causal structure without relying on distributional or feature-based matching.

Formally, given two graphs $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ with the same set of nodes V, the SHD is defined as:

$$\text{SHD}(G_1, G_2) = |E_1 \setminus E_2| + |E_2 \setminus E_1|, \tag{5}$$

where $E_1 \setminus E_2$ denotes the set of directed edges present in G_1 but not in G_2, and vice versa.

For the test graph G_T for time segment T, we compute:

$$SHD_{TA} = \text{SHD}(G_T, G_{\text{Attack}}), \quad SHD_{TN} = \text{SHD}(G_T, G_{\text{Normal}}) \tag{6}$$

representing the graph distance to the attack and normal references.

Decision Rule: We assign the predicted class label $\hat{y}_T$ by determining which reference graph the test graph more closely resembles:

$$\hat{y}_T = \begin{cases} 1, & \text{if } SHD_{TA} < SHD_{TN}, \\ 0, & \text{otherwise,} \end{cases} \tag{7}$$

where $\hat{y}_T = 1$ denotes an *Attack* state and $\hat{y}_T = 0$, a *Normal* state.

In essence, if the causal structure G_T of the test segment is more similar to the attack reference graph G_{Attack} than to the normal graph G_{Normal}, we classify it as anomalous. Otherwise, it is deemed normal. This approach enables CGAD to generalize to previously unseen attack types, as many cyberattacks introduce recurring causal disruptions, even if their surface signal patterns differ.

This framework effectively maps each test segment to an operational state using global structural characteristics rather than raw features. Notably, **SHD focuses on structural rather than distributional differences, making CGAD resilient to transient noise, nonstationarity, and high class imbalance, three defining challenges in cyber-physical anomaly detection.**

4 Experiments

We conduct experiments to answer the following questions:
RQ1: Does our method improve anomaly detection compared with baseline methods?
RQ2: What are the impacts of causal graph learning and graph divergence metrics in our framework? Is structural hamming distance the most effective graph comparison metric for anomaly detection?
RQ3: Is our method robust over different data conditions?
RQ4: How sensitive is our method to key hyperparameters?
RQ5: What is the computational cost of our method?

4.1 Experimental Setup

4.1.1 Datasets. We use four real-world datasets: **1) SWaT** [22]: A 11-day water treatment plant testbed data collected from 51 interconnected sensors. The dataset contains 16 fault events over the period, which had 7 days of "Normal" system status and 4 days of "Attack" system status. **2) WADI** [1]: Water Distribution testbed dataset collected over 16 days from 123 actuators/sensors. 15 attack events were recorded in the last 2 days of the collection period with "Attack" system status. **3) Tennessee Eastman (TE)** [8]: A chemical process simulation dataset with 52 sensors and 20 anomalies. Training spans 25 hours, testing 48 hours, with measurements every three minutes. Table 1 shows dataset statistics, train-test split, and causal graph node count. **4) Server Machine Dataset (SMD) [31]** is a server monitoring dataset over 5 weeks monitoring 28 server machines from 38 sensors. It is one of the largest public datasets for anomaly detection in multivariate time-series data.

Table 1: DATASET STATISTICS.

Dataset	Status	# Features	# Normal Data	# Attack Data	Normal to Attack Ratio
SWaT	Normal	51	495000	0	0
	Attack	51	395298	54621	7:1
WADI	Normal	123	1048571	0	0
	Attack	123	162824	9977	16:1
TE	Normal	52	450000	0	0
	Attack	52	222500	22800	10:1
SMD	Normal	38	708405	0	0
	Attack	38	678950	29470	25:1

4.1.2 Evaluation Metrics. **Point-adjusted F1-Score [10]:** In multivariate time-series data, attack events often form continuous segments, making pointwise anomaly detection less effective [11]. Our method requires a time-interval segment to capture the underlying causal structure. We use the point-adjusted F1-score, where a segment is labeled anomalous if any point within it is an anomaly, and the F1-score is computed based on the performance of our method on the entire segment. **ROC-AUC:** AUC, the area under the Receiver Operating Characteristic (ROC) curve, provides a measure of a model's ability to distinguish positive from negative samples. **PRC-AUC:** PRC-AUC focuses on the performance of the model on the anomalous class. This is particularly useful when the dataset is imbalanced, like in anomaly detection.

4.1.3 Baseline Algorithms. We compare our method with the following baseline algorithms: **1) One-Class Support Vector Machine (OC-SVM)** [38] – Supervised anomaly detection algorithm employing a kernel-based hyperplane decision boundary for classification of anomaly samples. **2) Isolation Forest** [36] – Ensemble supervised anomaly detection isolating anomalies via subsampling on streaming data. **3) Deep Support Vector Data Description (Deep SVDD)** [40] – Deep learning anomaly detection via hypersphere learning. **4) Hybrid CNN-LSTM** [29] – An efficient unsupervised anomaly detection framework. **5) Spatio-Temporal Outlier Detection (STOD)** [33] – A spatio-temporal outlier detection. **6) Angle Based Outlier Detection (ABOD)** [15] – Outlier detection using angle metrics. **7) Empirical Cumulative Distributed Functions for Outlier Detection (ECOD)** [16] – A parameter-free, interpretable unsupervised anomaly detection method. **8) Lightweight On-Line Detector of Anomalies (LODA)** [27] – Efficient unsupervised ensemble of weak detectors. **9) SMV-CGAD** [20]-Spectral multi-view causal graph anomaly detection with dense/sparse graph structures and deep graph convolution.

Implementation Details. **Assumptions.** We assume anomalous events are rare relative to normal behavior, introducing class imbalance that impairs conventional detection methods. **Data Setup.** Time-series data are split into non-overlapping 15-minute segments. Models are trained on historical data and tested on future segments to preserve causal validity. **Causal Learning.** We use DYNOTEARS from `CausalNex` [3] to learn Dynamic Bayesian Networks, with time-lags: 4 (SWaT), 3 (WADI), 4 (TE), 1 (SMD). Gaussian noise is added to attack data to regularize graph learning. **Baselines.** Competing methods use `PyOD` [41] with default settings. **Hardware.** Experiments were conducted on Intel i9-12900HK CPU, 32 GB RAM, and NVIDIA RTX 4090 GPU. The code repository is available in https://anonymous.4open.science/r/CGAD-4E18/

4.2 Experimental Results

4.2.1 RQ1: Overall Performance. To answer **RQ1** , Table 2 shows our method (**CGAD**) in overall outperforms baseline methods on the **SWaT**, **WADI**, **TE** and **SMD** datasets in terms of five metrics: $F1_{PA}$, ROC-AUC, and PRC-AUC. The experiments demonstrate four insights: 1) CGAD outperforms correlation-based and density-based methods by capturing stable cause-effect patterns that reflect true system dynamics. 2) Despite its lightweight design, CGAD surpasses deeper models like SMV-CGAD, highlighting the strength of structural divergence over fused deep representations. 3) CGAD maintains high F1 and ROC-AUC across all datasets, demonstrating resilience to distribution shifts and data imbalance. 4) CGAD offers structurally derived alerts based on causal deviation, crucial for actionable insights in high-stakes infrastructure systems. While CGAD-DYNOTEARS shows strong overall performance, its effectiveness relies on the assumption that causal structures can be reliably and robustly estimated from segmented time-series data. In highly nonstationary settings where causal discovery algorithms fail to recover meaningful structures, CGAD's performance may degrade. But SMV-CGAD, by integrating dense and sparse views with deep representations, can be more robust in such cases by capturing implicit patterns even when explicit structures are unreliable.

4.2.2 RQ2: Study of Causal Graph Learning and Graph Divergence Metrics. To answer **RQ2**, we develop an ablation study to examine two technical components.

Effect of Causal Graph Learning (Phase 1): Our method considers temporal causal structures from multivariate time series. The baseline method, DAGs with NO TEARS [42], performs poorly due to ignoring temporal dependencies, latent confounders, and delayed attack effects in cyber-physical systems. In contrast, DYNOTEARS, which models regressive temporal dynamics, significantly improves AUC and $F1_{PA}$, confirming the importance of temporal modeling for robust anomaly detection."

Effect of Graph Divergence Metrics (Phase 2): We detect anomalies by measuring structural change between causal graphs using a graph comparison metric. We compare several alternatives, including **Jaccard similarity** (edge set overlap), and **Laplacian spectral distance**. We observe that while these metrics yield comparable trends in anomaly localization, they fail to capture subtle structural deviations. However, Structural Hamming Distance balances accuracy with efficiency and achieves the best trade-off between detection fidelity and runtime cost.

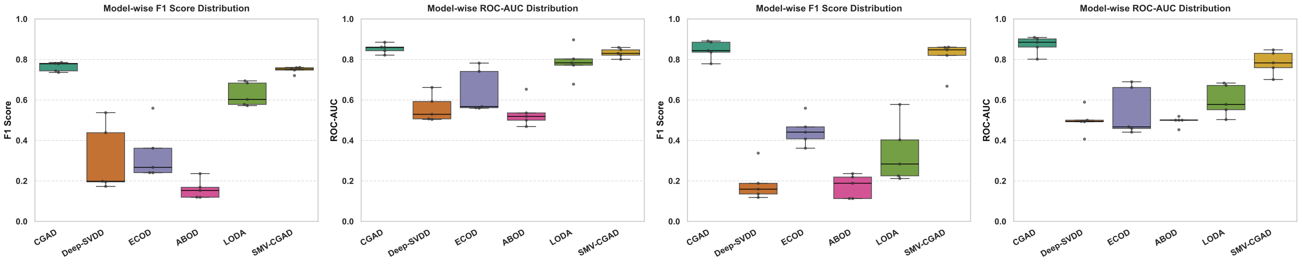
4.2.3 RQ3: Robustness Check. To answer RQ3, we evaluate the robustness of CGAD by assessing whether CGAD can consistently detect anomalies via a causal perspective across multiple subsets of temporal streams on the SWaT and WADI datasets. Figure 3 shows that CGAD achieves stable performance across multiple balanced data subsets, indicating the robustness of causal graphs learned from different samples of the same underlying distribution. However, a notable drop in all evaluation metrics is observed when a data subset is significantly imbalanced. This degradation underscores a known limitation of the DYNOTEARS algorithm—its sensitivity to the availability of high-quality, causally relevant data during structure learning. Additionally, we observe that the omission of Gaussian noise during the causal graph construction for the 'Attack' state further reduces performance. This suggests that introducing controlled noise into the attack data aids in capturing the stochastic nature of attack-induced perturbations, thereby improving the generalization capacity of the learned causal representations. These findings highlight the importance of data quality, balance, and controlled regularization for robust causal discovery in adversarial cyber-physical environments.

4.2.4 RQ4: Impact of Time-Lag Parameter in DYNOTEARS. The time-lag parameter in the DYNOTEARS algorithm is critical in modeling the temporal dependencies that characterize delayed effects of cyberattacks. It defines the maximum historical window considered when estimating temporal causal relationships. To answer RQ4, we conduct a sensitivity analysis by manually tuning the time-lag parameter across multiple values and evaluating the detection performance. Figure 4 shows that optimal performance is achieved with a time-lag of 4, 3, 4, and 1 for the SWaT, WADI, TE, and SMD datasets, respectively. Smaller lag values tend to miss delayed causal effects, while larger values risk overfitting and introducing instability in the learned structures. This analysis confirms that the effectiveness of DYNOTEARS in CGAD is highly dependent on appropriate lag selection, which must be tailored to the specific temporal characteristics of the underlying system dynamics.

Table 2: Overall Performance Across All Datasets

Methods	SWAT			WADI		
	$F1_{PA}$	ROC-AUC	PRC-AUC	$F1_{PA}$	ROC-AUC	PRC-AUC
STOD	0.6955	0.8420	0.7013	0.4126	0.5176	0.5660
Deep-SVDD	0.1729	0.6618	0.2927	0.1591	0.4938	0.2798
CNN-LSTM	0.6129	0.6881	0.5997	0.7891	0.8138	0.7798
ECOD	0.2403	0.7819	0.3320	0.4077	0.6897	0.5137
LODA	0.6942	0.8972	0.7154	0.2251	0.5516	0.2920
ABOD	0.1194	0.5000	0.2321	0.1126	0.5000	0.05
One-class SVM	0.7385	0.6632	0.7441	0.5343	0.4208	0.6121
Isolation Forest	0.7412	0.8426	0.8122	0.6434	0.6487	0.6850
SMV-CGAD	0.7532	0.8211	0.7355	0.6679	0.7831	0.7077
CGAD - DAGNOTEARS	0.1156	0.5427	0.2322	0.5879	0.7120	0.5470
CGAD-DYNOTEARS	**0.7807**	**0.8611**	**0.7388**	**0.8913**	**0.9015**	**0.8504**

Methods	TE			SMD		
	$F1_{PA}$	ROC-AUC	PRC-AUC	$F1_{PA}$	ROC-AUC	PRC-AUC
STOD	0.6421	0.7280	0.7106	0.6779	0.8076	0.6840
Deep-SVDD	0.4801	0.5679	0.4680	0.3911	0.5448	0.4794
CNN-LSTM	0.6778	0.8328	0.6990	0.6490	0.5030	0.5720
ECOD	0.4003	0.5000	0.4222	0.5827	0.5100	0.6100
LODA	0.4228	0.5007	0.5102	0.2656	0.5030	0.2866
ABOD	0.1827	0.4730	0.2560	0.6244	0.5027	0.5009
One-class SVM	0.8125	0.8231	0.8441	0.8443	0.8288	0.8624
Isolation Forest	0.7421	0.8102	0.7684	0.8501	0.8734	0.8681
SMV-CGAD	0.7389	0.8002	0.7556	0.8395	0.8030	0.8112
CGAD - DAGNOTEARS	0.3512	0.5011	0.3323	0.4670	0.5802	0.5017
CGAD-DYNOTEARS	**0.8297**	**0.8516**	**0.7888**	**0.8626**	**0.8542**	**0.9004**



(a) SWaT: F1 Score Distribution (b) SWaT: ROC-AUC Distribution (c) WADI: F1 Score Distribution (d) WADI: ROC-AUC Distribution

Figure 3: Model robustness analysis across test subsets on the SWaT and WADI datasets. Each boxplot illustrates the distribution of detection performance across five temporal subsets for each model.

4.2.5 RQ5: Runtime Analysis and Computational Efficiency. To answer RQ5, we analyzed CGAD's computational efficiency by comparing its training and inference times against all baselines on the SWaT and WADI datasets. Figure 5 demonstrates CGAD's competitive training time, comparable to LODA and significantly faster than Deep-SVDD and ABOD. This efficiency stems from DYNOTEARS' score-based optimization, which avoids exhaustive graph search and scales effectively with variables. Inference time, which involves learning causal graphs for each test segment, is moderately higher due to the segment-wise causal discovery. However, the trade-off is justified by the model's superior transparency and accuracy, especially in the presence of complex or delayed attack patterns. Overall, CGAD offers a practical balance between computational efficiency and detection robustness, making it well-suited for deployment in real-time industrial monitoring environments.

4.2.6 Discussions: Scalability Considerations and Future Directions. Causal graph learning, by design, seeks to uncover underlying cause-effect relationships among features—in this case, the sensors within cyber-physical systems. A key advantage of the CGAD framework is that as the data scale increases, the resulting causal representations become more expressive, capturing richer and more generalized system behaviors. This scalability potential distinguishes CGAD from traditional methods that struggle to model such complexity effectively. Our runtime analysis demonstrates that CGAD achieves relatively low training time compared to several baseline models. While test-time inference is moderately slower due to the need to learn causal structures per time segment, the DYNOTEARS algorithm's score-based optimization remains tractable even for high-dimensional datasets. This computational efficiency makes CGAD a viable option for deployment in real-world

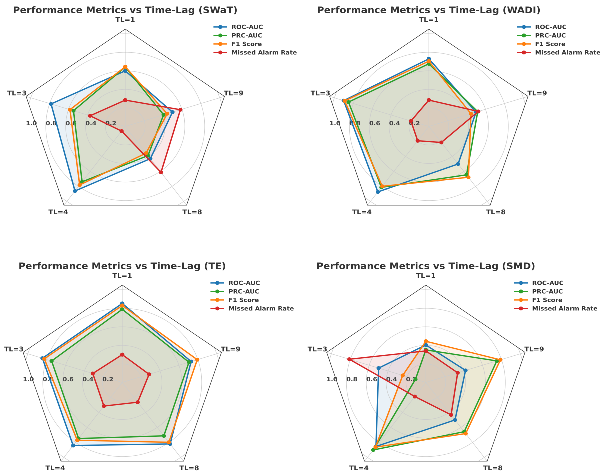
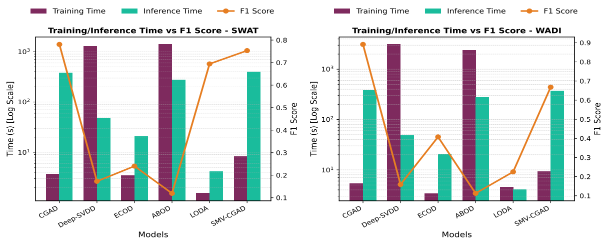


Figure 4: CGAD Performance vs. Time-lag parameter.



(a) SWaT: Train-Test time comparison (b) WADI: Train-Test time comparison

Figure 5: Runtime analysis for training and inference phases.

settings that demand real-time monitoring over time-series data. Nonetheless, several scalability challenges remain, consistent with broader limitations of machine learning in industrial settings:

- **Computational Cost:** Causal graph learning over large datasets can be computationally intensive. To mitigate this, we propose continual causal learning, selective modeling on representative subsets, domain-informed constraints, and early stopping to reduce overhead at scale.
- **Data Quality:** Noisy, inconsistent, or biased sensor data may compromise causal discovery. Robust preprocessing, outlier handling, and noise-tolerant graph learning are essential for accurate structure estimation.
- **Operational Constraints:** Deployment may be hindered by regulatory and organizational barriers. These require interpretable models and collaboration with domain stakeholders.

To address these concerns, we are actively pursuing industry collaborations to validate CGAD in pilot deployments. As a forward-looking extension, we aim to develop a *continuous causal learning* module that incrementally updates the causal graph in response to real-time sensor feedback. Such a mechanism would enhance adaptability and resilience to concept drift, ensuring sustained performance in dynamic operational environments.

5 Related Work

Our framework aligns at the intersection of *causal discovery in time-series*, *graph-based reasoning*, and *robust anomaly detection.*

Causal Discovery in Time-Series. Learning temporal causality in CPS requires uncovering both lagged and instantaneous dependencies, often obscured by noise. Early VAR-based Granger causality methods lack scalability and fail under nonlinearity. Recent advances like TCDF [24] provide neural or score-based causal discovery, modeling multivariate dynamics with time lags. Gong et al. [12] survey modern methods (e.g., PCMCI, attention-based causal transformers [39]) that achieve state-of-the-art results. However, many still rely on strong stationarity assumptions.

Graph-Based Reasoning in CPS. Graph abstractions represent complex infrastructure states in CPS. Gorawski et al. [13] model smart city infrastructure as sensor networks, while Kirchheim et al. [14] show structured representations improve anomaly explanation. Deep graph models (e.g., GCFormer [35], GDN [9]) capture spatio-temporal dependencies but rely on predefined topologies, limiting transparency. Recent work explores causal interventions to improve cross-graph generalization [26], highlighting the value of structural invariance in anomaly-prone environments.

Anomaly Detection in Cyber-Physical Systems. Anomaly detection in CPS is challenging due to scarce labels and high-dimensional, noisy streams. Traditional statistical or density-based methods [7] suffer from high false positives. Deep models like Deep SVDD [28] or Anomaly Transformer [37] improve recall but struggle with explainability and drift. Semi-supervised frameworks [32] reduce supervision needs but often lack robustness to evolving attack patterns. CGAD addresses these by modeling causal invariants across system states, enabling reliable detection with actionable insights.

6 Conclusion Remarks

In this work, we addressed the critical challenge of detecting cyberattacks in Water Treatment Networks by proposing a novel causal graph-based anomaly detection framework, **CGAD**. It operates in two phases: (i) *Causal Profiling*, employing DYNOTEARS to learn ground-truth causal graphs for "Normal" and "Attack" system behaviors; and (ii) *Causal Scoring*, where segmented sensor data's causal graphs are inferred and compared to references via structural divergence. Through extensive experimentation on four real-world cyber-physical datasets, we demonstrate CGAD achieves high detection accuracy and robustness to class imbalance, distributional shifts, and delayed attack manifestations. By leveraging causal stability over correlation, CGAD offers explainable, efficient, and generalizable anomaly detection in complex time-series environments. While promising, the framework presents scalability challenges, particularly in causal graph learning for high-dimensional or noisy data. Addressing these requires efficient graph learning strategies and adaptation for evolving system behaviors. Looking ahead, we envision extending CGAD for real-time deployment in large-scale industrial and critical infrastructure systems. Domains like Finance, Healthcare, and Cybersecurity can greatly benefit from CGAD's explainable causal reasoning and structural anomaly detection capabilities. Future work will focus on continuous causal learning and collaborative testing with industry partners to ensure robustness and adaptability in dynamic operational settings.

7 GenAI Usage Disclosure

Generative AI tools were used to refine sections of this paper for improved clarity, coherence, and grammatical accuracy. All core ideas, algorithms, experiments, and analyses were conceived, implemented, and validated solely by the authors.

References

[1] Chuadhry Mujeeb Ahmed, Venkata Reddy Palleti, and Aditya P. Mathur. 2017. WADI: a water distribution testbed for research in the design of secure cyber physical systems. In *Proceedings of the 3rd International Workshop on Cyber-Physical Systems for Smart Water Networks* (Pittsburgh, Pennsylvania) *(CySWATER '17)*. Association for Computing Machinery, New York, NY, USA, 25–28. https://doi.org/10.1145/3055366.3055375

[2] Mohiuddin Ahmed, Abdun Naser Mahmood, and Jiankun Hu. 2016. A survey of network anomaly detection techniques. *Journal of Network and Computer Applications* 60 (2016), 19–31.

[3] Paul Beaumont, Ben Horsburgh, Philip Pilgerstorfer, Angel Droth, Richard Oentaryo, Steven Ler, Hiep Nguyen, Gabriel Azevedo Ferreira, Zain Patel, and Wesley Leong. 2021. *CausalNex*. https://github.com/quantumblacklabs/causalnex

[4] Paul Boniol and Themis Palpanas. 2020. Series2Graph: graph-based subsequence anomaly detection for time series. *Proceedings of the VLDB Endowment* 13, 12 (Aug. 2020), 1821–1834. https://doi.org/10.14778/3407790.3407792

[5] Markus M Breunig, Hans-Peter Kriegel, Raymond T Ng, and Jörg Sander. 2000. LOF: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*. 93–104.

[6] James Cervini, Aviel Rubin, and Lanier Watkins. 2022. Don't drink the cyber: Extrapolating the possibilities of Oldsmar's water treatment cyberattack. In *International conference on cyber warfare and security*, Vol. 17. Academic Conferences International Limited, 19–25.

[7] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly detection: A survey. *ACM computing surveys (CSUR)* 41, 3 (2009), 1–58.

[8] Xiaolu Chen. 2019. Tennessee Eastman simulation dataset. https://doi.org/10.21227/4519-z502

[9] Xiaoyun Deng, Yujie Zhou, Wei Hu, Yadong Wang, and Lichao Zhang. 2021. Graph neural network-based anomaly detection in multivariate time series. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 2829–2838.

[10] Falih Gozi Febrinanto, Kristen Moore, Chandra Thapa, Mujie Liu, Vidya Saikrishna, Jiangang Ma, and Feng Xia. 2023. Entropy Causal Graphs for Multivariate Time Series Anomaly Detection. arXiv:2312.09478 [cs.LG]

[11] Astha Garg, Wenyu Zhang, Jules Samaran, Ramasamy Savitha, and Chuan-Sheng Foo. 2022. An Evaluation of Anomaly Detection and Diagnosis in Multivariate Time Series. *IEEE Transactions on Neural Networks and Learning Systems* 33, 6 (2022), 2508–2517. https://doi.org/10.1109/TNNLS.2021.3105827

[12] Chang Gong, Di Yao, Chuzhe Zhang, Wenbin Li, and Jingping Bi. 2023. Causal Discovery from Temporal Data: An Overview and New Perspectives. arXiv:2303.10112 [cs.LG]

[13] Michał Gorawski and Krzysztof Grochla. 2019. Graph Representation of Linear Infrastructure in Smart City IoT Systems. In *2019 11th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*. 1–4. https://doi.org/10.1109/ICUMT48472.2019.8970674

[14] Konstantin Kirchheim. 2023. Towards Deep Anomaly Detection with Structured Knowledge Representations. In *International Conference on Computer Safety, Reliability, and Security*. Springer, 382–389.

[15] Hans-Peter Kriegel, Matthias Schubert, and Arthur Zimek. 2008. Angle-Based Outlier Detection in High-Dimensional Data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Las Vegas, Nevada, USA) *(KDD '08)*. Association for Computing Machinery, New York, NY, USA, 444–452. https://doi.org/10.1145/1401890.1401946

[16] Zheng Li, Yue Zhao, Xiyang Hu, Nicola Botta, Cezar Ionescu, and George H. Chen. 2023. ECOD: Unsupervised Outlier Detection Using Empirical Cumulative Distribution Functions. *IEEE Transactions on Knowledge and Data Engineering* 35, 12 (2023), 12181–12193. https://doi.org/10.1109/TKDE.2022.3159580

[17] Qin Lin, Sridha Adepu, Sicco Verwer, and Aditya Mathur. 2018. TABOR: A Graphical Model-Based Approach for Anomaly Detection in Industrial Control Systems. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security* (Incheon, Republic of Korea) *(ASIACCS '18)*. Association for Computing Machinery, New York, NY, USA, 525–536. https://doi.org/10.1145/3196494.3196546

[18] Arun Vignesh Malarkkan, Haoyue Bai, Xinyuan Wang, Anjali Kaushik, Dongjie Wang, and Yanjie Fu. 2025. Rethinking spatio-temporal anomaly detection: A vision for causality-driven cybersecurity. *arXiv preprint arXiv:2507.08177* (2025).

[19] Arun Vignesh Malarkkan, Dongjie Wang, Haoyue Bai, and Yanjie Fu. 2025. Incremental Causal Graph Learning for Online Cyberattack Detection in Cyber-Physical Infrastructures. *arXiv preprint arXiv:2507.14387* (2025).

[20] Arun Vignesh Malarkkan, Dongjie Wang, and Yanjie Fu. 2024. Multi-view Causal Graph Fusion Based Anomaly Detection in Cyber-Physical Infrastructures. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management* (Boise, ID, USA) *(CIKM '24)*. Association for Computing Machinery, New York, NY, USA, 4760–4767. https://doi.org/10.1145/3627673.3680096

[21] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, Puneet Agarwal, et al. 2015. Long short term memory networks for anomaly detection in time series.. In *Esann*, Vol. 2015. 89.

[22] Aditya P. Mathur and Nils Ole Tippenhauer. 2016. SWaT: a water treatment testbed for research and training on ICS security. In *2016 International Workshop on Cyber-physical Systems for Smart Water Networks (CySWater)*. 31–36. https://doi.org/10.1109/CySWater.2016.7469060

[23] Douglas C Montgomery. 2007. *Introduction to statistical quality control*. John Wiley & Sons.

[24] Meike Nauta, Darius Bucur, and Christin Seifert. 2019. Causal discovery with attention-based convolutional neural networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2299–2307.

[25] Roxana Pamfil, Nisara Sriwattanaworachai, Shaan Desai, Philip Pilgerstorfer, Paul Beaumont, Konstantinos Georgatzis, and Bryon Aragam. 2020. DYNOTEARS: Structure Learning from Time-Series Data. arXiv:2002.00498 [stat.ML]

[26] Zhiqiang Pan, Chen Gao, Fei Cai, Wanyu Chen, Xin Zhang, Honghui Chen, and Yong Li. 2025. On the Cross-Graph Transferability of Dynamic Link Prediction. In *Proceedings of the ACM on Web Conference 2025* (Sydney NSW, Australia) *(WWW '25)*. Association for Computing Machinery, New York, NY, USA, 4101–4110. https://doi.org/10.1145/3696410.3714712

[27] Tomáš Pevnỳ. 2016. Loda: Lightweight on-line detector of anomalies. *Machine Learning* 102 (2016), 275–304.

[28] Lukas Ruff, Robert Vandermeulen, Nico Görnitz, Lucas Deecke, Malik Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. 2018. Deep one-class classification. In *International Conference on Machine Learning*. PMLR, 4393–4402.

[29] Ali Kivanc Sahin, Bora Cavdar, Ramazan Ozgur Dogan, Selen Ayas, Busra Ozgenc, and Mustafa Sinasi Ayas. 2023. A Hybrid CNN-LSTM Framework for Unsupervised Anomaly Detection in Water Distribution Plant. In *2023 Innovations in Intelligent Systems and Applications Conference (ASYU)*. 1–6. https://doi.org/10.1109/ASYU58738.2023.10296546

[30] Robin Sommer and Vern Paxson. 2010. Outside the closed world: On using machine learning for network intrusion detection. In *2010 IEEE symposium on security and privacy*. IEEE, 305–316.

[31] Ya Su, Youjian Zhao, Chenhao Niu, Rong Liu, Wei Sun, and Dan Pei. 2019. Robust Anomaly Detection for Multivariate Time Series through Stochastic Recurrent Neural Network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Anchorage, AK, USA) *(KDD '19)*. Association for Computing Machinery, New York, NY, USA, 2828–2837. https://doi.org/10.1145/3292500.3330672

[32] Miryam Elizabeth Villa-Pérez, Miguel Á. Álvarez Carmona, Octavio Loyola-González, Miguel Angel Medina-Pérez, Juan Carlos Velazco-Rossell, and Kim-Kwang Raymond Choo. 2021. Semi-supervised anomaly detection algorithms: A comparative summary and future research directions. *Knowledge-Based Systems* 218 (2021), 106878. https://doi.org/10.1016/j.knosys.2021.106878

[33] Dongjie Wang, Pengyang Wang, Jinbo Zhou, Leilei Sun, Bowen Du, and Yanjie Fu. 2020. Defending Water Treatment Networks: Exploiting Spatio-Temporal Effects for Cyber Attack Detection. In *2020 IEEE International Conference on Data Mining (ICDM)*. 32–41. https://doi.org/10.1109/ICDM50108.2020.00012

[34] S. Xing, J. Niu, and T. Ren. 2023. GCFormer: Granger Causality based Attention Mechanism for Multivariate Time Series Anomaly Detection. In *2023 IEEE International Conference on Data Mining (ICDM)*. IEEE Computer Society, Los Alamitos, CA, USA, 1433–1438. https://doi.org/10.1109/ICDM58522.2023.00187

[35] Shijie Xing, Jing Niu, and Tao Ren. 2023. GCFormer: Granger Causality based Attention Mechanism for Multivariate Time Series Anomaly Detection. In *2023 IEEE International Conference on Data Mining (ICDM)*. IEEE, 1433–1438. https://doi.org/10.1109/ICDM58522.2023.00187

[36] Dong Xu, Yanjun Wang, Yulong Meng, and Ziying Zhang. 2017. An improved data anomaly detection method based on isolation forest. In *2017 10th international symposium on computational intelligence and design (ISCID)*, Vol. 2. IEEE, 287–291.

[37] Wenqian Xu, Yingbo Zhao, Jian Pei, Xiaoyun Lin, and Yanming He. 2022. Anomaly transformer: Time series anomaly detection with association discrepancy. *Advances in Neural Information Processing Systems* 35 (2022), 28858–28872.

[38] Shen Yin, Xiangping Zhu, and Chen Jing. 2014. Fault detection based on a robust one class support vector machine. *Neurocomputing* 145 (2014), 263–268.

[39] Xiaotian Zhang, Yuan Luo, Bin Hu, and Hailiang Wang. 2023. A Survey on Attention-based Causal Discovery. *arXiv preprint arXiv:2306.00557* (2023).

[40] Zheng Zhang and Xiaogang Deng. 2021. Anomaly detection using improved deep SVDD model with data structure preservation. *Pattern Recognition Letters* 148 (2021), 1–6. https://doi.org/10.1016/j.patrec.2021.04.020

[41] Yue Zhao, Zain Nasrullah, and Zheng Li. 2019. PyOD: A Python Toolbox for Scalable Outlier Detection. *Journal of Machine Learning Research* 20, 96 (2019),

1–7. http://jmlr.org/papers/v20/19-011.html

[42] Xun Zheng, Bryon Aragam, Pradeep Ravikumar, and Eric P. Xing. 2018. DAGs with NO TEARS: Continuous Optimization for Structure Learning. arXiv:1803.01422 [stat.ML]