**Causanom: Anomaly Detection With Flexible Causal Graphs**

Sasha Strelnikoff  
Intelligent Systems Laboratory  
HRL Laboratories, LLC  
sstrelnikoff@hrl.com

Aruna Jammalamadaka  
Intelligent Systems Laboratory  
HRL Laboratories, LLC  
ajammaladaka@hrl.com

Tsai-Ching Lu  
Intelligent Systems Laboratory  
HRL Laboratories, LLC  
tlu@hrl.com

---

**Abstract**

Causality-based anomaly detection methods provide at least two significant theoretical benefits over purely statistical methods: 1. Improved robustness to non-anomalous out-of-distribution data, which implies a reduction in false-alarms; 2. A potential for failure localization due to the topological ordering of the causal graph. Recent studies have considered the utilization of causality-based methods for time series anomaly detection, however, these methods require the causal graph to be fixed; resulting such methods are not robust to incorrectly estimated causal graphs and are not able to natively model counterfactual scenarios. To address these limitations, we introduce Causanom: a graph-based encoder-decoder neural network for time series anomaly detection. Causanom utilizes a node conditional data-stream representation in conjunction with a weighted graph aggregation function in order to efficiently capture heterogeneous node dynamics whilst allowing for a flexible graphical structure. We show that Causanom can be trained along with auxiliary constraints in order to tune the causal graph and improve performance. Additionally, we show that Causanom can be used to produce counterfactual data, which we leverage to identify violated causal relationships. Using real and synthetic time series data respectively, we show that Causanom performs at least as well as state-of-the-art baselines in the anomaly detection task and outperforms existing methods in a causal attribution task.

Although both detection and attribution are required to recommend actions for full process recovery, the vast majority of work focuses only on detection (Vuković and Thulmann, 2022). Given a stream of multivariate time-series data, structural (graphical) models of dependencies between variables help us diagnose fault origins by tracing backwards through directed paths to anomalous node ancestors. Non-causal anomaly detection methods to estimate structure include using sensor similarity to determine edges in the graph (Deng and Hooi, 2021), or modeling the correlation matrix between variables (Zhang et al., 2019). Since the networks produced by these methods do not capture causal dependencies on their edges (e.g., the left and right engine of an airplane would have high correlation but no direct causal relationship), they are obviously not ideal for attributing causes to detected failures. Additionally, such correlation based methods are likely to produce false alarms in the anomaly detection task, since non-anomalous out-of-distribution data is likely to occur in many real-world applications.

Methods which leverage Bayesian networks to model structure can have causal interpretations, since they are generally assumed to be directed and acyclic. However, the existing literature focuses only on maximum likelihood estimates which score the network against the data (Krishnamurthy, Sarkar, and Tewari, 2014). Attribution is either performed simply by finding the lowest probability node (Diallo et al., 2018), or is not performed at all (Wunderlich and Niggemann, 2017).

While statistical models like Bayesian networks specify a single probability distribution, a causal model represents a set of distributions – one for each possible intervention. In this way, they are able to model failure-modes of real-world systems, where a failure is assumed to be an intervention, and “intervention recognition” (Li et al., 2022) is the process of identifying which node (or set of nodes) has failed. Recent work focuses on using neural approaches to estimating causal dependencies in multivariate time series data (Shah, Dang, and Zerfos, 2018; Yang, Zhang, and Hoi, 2022; Qiu et al., 2020; Li et al., 2022). In these, violations of the estimated causal dependencies constitute anomalies, and anomaly scores from individual nodes or edges are summed along the paths to root nodes. The highest scoring path is output as a diagnostic trace for attribution.

While causality-based models are the most promising for accurately modeling the data generating process, the above

---

**Introduction**

In many real-world systems, rapidly detecting and diagnosing anomalies can save time and money spent on mitigation of potentially cascading failures. Moreover, since many relevant systems have complex temporal dependencies, the problem of time series anomaly detection and attribution has become an increasingly important task across a wide range of critical real-world applications, such as fraud detection (Kou et al., 2004; Pourhabibi et al., 2020), online service system failures (Li et al., 2022; Meng et al., 2020), and aviation safety assurance (Memarzadeh, Matthews, and Avrekh, 2020). By utilizing time-series anomaly detection and attribution methods, complex systems can be managed more efficiently and incidents can be identified in a timely manner.
methods require a fixed causal graph. This results in the inability to adapt (without fully retraining) to an incorrectly estimated or modified causal graph, as may occur when the initial structure learning is inaccurate, causal relationships are intervened upon, or additional variables need to be modeled. Additionally, all of the methods discussed above tackle the problem of attribution via statistical anomaly propagation. Theoretically, this method is flawed in light of the Principle of Independent Mechanisms, which states that “The causal generative process of a system’s variables is composed of autonomous modules that do not inform or influence each other. In the probabilistic case, this means that the conditional distribution of each variable given its causes (i.e., its mechanism) does not inform or influence the other mechanisms.” (Schlkopf et al., 2021). This implies that the effects of an intervention should be contained in a single local causal mechanism. For our purposes, this indicates that an anomaly that is the result of a single failure point should be detected in only a single node, assuming the causal dependencies are accurately modeled, and therefore anomaly propagation is not applicable.

In this paper, we propose a method for flexibly incorporating a causal graph for simultaneous anomaly detection and localized causal attribution. This graph flexibility not only allows us to quickly adapt to new causal structures but also enables us to perform intervention-based attribution, with which we identify a class of anomalous edges by comparing predictions from soft interventions to observed data. Our contributions are:

- An anomaly detection model which leverages a flexible causal graphical structure in such a way that the structure can be dynamically updated and easily incorporate supervision.
- An introduction of causal edge anomaly attribution and a method for attribution based on simulated interventions.
- Quantitative evaluations of improvements in detection and edge attribution across multiple synthetic and real-world datasets.

The remainder of this paper is structured as follows. The first section provides the mathematical problem formulation. The second section outlines our novel method for anomaly detection and attribution, along with its advantages. The third section describes our experiments and results. The final section concludes with a discussion and potential avenues for future work.

### Problem Formulation

We study a temporal system which, when not in an anomalous state, has a stationary generating causal structure $G$ with corresponding random variables $\{X_k(t)\}_{k=1}^N$ for all times $t$. Denoting $\mathbb{P}A(k)$ as the collection of parent nodes of $k$ in $G$, assume the unobserved local causal mechanisms, for each $k \in \mathcal{N}_c := \{n \in G \mid |\mathbb{P}A(n)| > 0\}$, are defined by the system of equations

$$X_k(t) = f_k (\{X_n(t') \mid n \in \mathbb{P}A(k), t' \leq t\}) + \eta_k^t, \quad (1)$$

where $\{f_k\}$ are arbitrary and the collection $\{\eta_k^t \sim \mathcal{N}(0, \sigma_k)\}$ is independent with respect to both $k$ and $t$.

An anomalous state of the system is one in which the generating process deviates from that described in (1) for at least one value of $k$. In line with prior research, we pose this as an unsupervised anomaly detection problem. In particular, given a collection of nominal data $D_{train}$ generated by a latent process described by (1), and labeled anomalous data $D_{test}$, containing both nominal data and data resulting from an anomalous state of the system, we aim to learn a model $M$ to detect the anomalies in $D_{test}$ using only $D_{train}$ as training data.

For the problem of edge-attribution, we focus on a subclass of additive causal mechanisms and anomalies arising from edge-perturbations. In this restricted context, we aim to identify the perturbed edges in $D_{test}$ using the pretrained model $M$. We leave the details of this to the Edge-Anomaly Attribution Section.

### Methods

#### Anomaly Detection Model

The definition of an anomalous state introduced above suggests a natural definition for a score, $s$, of anomaly associated with the $k$th dimension of $x = (x_1, ..., x_N)$ as

$$s(x_k) = -\mathbb{P}X_k|\mathbb{P}A(X_k) (x_k \mid x_n, n \in \mathbb{P}A(k)) \quad (2)$$

and the anomaly score of $x$ as the maximum anomaly score over each of its components,

$$S(x) = \max_k (s(x_k)) \quad (3)$$

Since the probability densities $\mathbb{P}X_k|\mathbb{P}A(X_k)$ and graph structure $G$ are unknown, our aim is to estimate these latent features in order to produce an estimated anomaly score $\hat{S}(x)$ for an observed value $x$. We assume an initial estimate of the causal structure, $\hat{G}$, with adjacency matrix $A \in \mathbb{R}^{N \times N}$, which may be informed via a subject matter expert (SME) or any data-driven structure learning methodology (such as the PC algorithm (Colombo, Maathuis, and others, 2014)).

In order to model $\mathbb{P}X_k|\mathbb{P}A(X_k)$, we leverage a simple generative method augmented with a variant of a Graph Convolutional Network (GCN) which we use to incorporate the graph structure $\hat{G}$. Our proposed pipeline is illustrated in Figure 1. For each time step $t$, we compute the data embedding using a learned node embedding vector concatenated with the input datastream, $\tilde{x}_k^t(t) = (x_k(t - \tau), ..., x_k(t))$, for a fixed time window $\tau$. We then compute the graph weighted average over all of the data embeddings to obtain a parent-node embedding for each node $k \in \mathcal{N}_c$. Finally, this embedding

![Figure 1: Causalom pipeline.](image_url)
We denote this vector as $W$. The aim of the prediction network is to estimate the conditional distribution of each of the non-root nodes $z_k$, any prediction-based method of detecting conditional values ($W$ simply requiring $Z_k$, $k$-tuple $W$) can additionally be used for root-node modeling by the initial weight matrix, we can simply set $W_{θ} = \{ W_k \mid k \in N_c \}$, and these mechanisms are independent, maximizing the likelihood of $θ$ is equivalent to minimizing sum of the log losses,

$$L(x(t), M(x^z(t); θ)) = \sum_{k \in N_c} l(μ_k(t), σ_k(t), x_k(t)), \quad (4)$$

where $l(μ, σ, x) = \log(σ) + \frac{1}{2} \left( \frac{x - μ}{σ} \right)^2$.

We consider the baseline method to be one in which the parameters in $W$ remain fixed, however in the subsequent experiments, we investigate the impact of fine-tuning $W$. Fine-tuning could be done most simply in an unconstrained manner, however, this would not necessarily preserve the directed acyclicity of $W$. In NOTEARS structure learning (Zheng et al., 2018), the authors introduce a constraint function $h(W) = tr(W^2) - l$, for $W \in \mathbb{R}^{l \times l}$, and prove $h(W) = 0$ iff $W$ is a DAG. To enforce this constraint, we add $γb(W)$ to (4) with weighting $γ > 0$.

**Edge-Anomaly Attribution**

For purposes of this section, restrict the characterization in (1) to causal systems of the form

$$X_k(t) = f_k \left( \sum_{n \in \mathbb{P}(k)} g_{kn}(X_n(t)) \right) + \eta_k^t, \quad (5)$$

where $\{ f_k \}, \{ g_{kn} \}$ are arbitrary measurable functions. We say there is an edge anomaly at time $t$ if there exists $(n, k) \in G$ with $g_{kn}(t) \rightarrow \hat{g}_{kn}(t)$ for some function $\hat{g}_{kn} \neq g_{kn}$. In this setting, we formulate the edge-anomaly attribution problem as that of identifying the edge(s) $(n, k)$ corresponding to an edge anomaly. In the following, we only consider edge anomalies which are the result of a single edge anomaly in which $\hat{g}_{kn} = 0$. We refer to this as a broken edge anomaly.

Given a trained Causanom model, $M(x(t); θ)$, and detected anomaly at node $k$, we propose the following edge-anomaly attribution pipeline: For each $n \in \mathbb{P}(k)$, set $W_{nk} = 0$, re-normalize the $k$th column of $W$ to produce the “intervened” causal graph. Then construct an estimated

2We are implicitly assuming that $\{ \eta_k^t \}$ are gaussian and identically distributed for a fixed $k$, however, these conditions could be relaxed by considering a mixture of gaussians (as in mixture density networks) or allowing the standard deviation/mixture terms to be time dependent, respectively.
score, $\tilde{S}_n$, using each of the modified models and choose the broken edge as $(\hat{n}, k)$, $\hat{n} = \arg\min_n \tilde{S}_n(x_{\hat{n}})$, based on the assumption that the minimum scoring model corresponds to the broken edge anomaly. Pseudocode for this procedure is provided in Algorithm 1 below.

```plaintext
Algorithm 1 Identify Broken Edge Anomaly
1: Input: Model $M$, anomalous node $k$, $x$, $W$
2: Output: brokenEdge
3: min_score $\leftarrow \infty$
4: $W_0 \leftarrow W$
5: for $n$ in $P_A(k)$ do
6: $W[n, k] \leftarrow 0$
7: $W[n, k] \leftarrow W[n, k]/W[\cdot, k]$
8: score $\leftarrow \tilde{S}_n(x_{\hat{n}})$
9: if score $< \text{minScore}$ then
10: brokenEdge $\leftarrow (\hat{n}, k)$
11: $W \leftarrow W_0$
12: return brokenEdge
```

Experiments and Results

In this section, we evaluate our system’s anomaly detection and localization performance using realistic and synthetic data respectively. In order to test the impact of the causal graph fine-tuning, in addition to the fixed-$W$ baseline version of our model, we consider both models with unconstrained and DAG-constrained fine-tuning of $W$ as introduced in the Methods section.

Anomaly Detection

Datasets In order to evaluate the efficacy of our model for anomaly detection, we utilize two common anomaly detection benchmark datasets, referred to as SWaT (Mathur and Tippenhauer, 2016) and WADI (Ahmed, Palleti, and Mathur, 2017). Both datasets are constructed from sensor readings of small-scale water treatment plant systems.

The SWaT contains time series data for 51 discrete and continuous sensors. This data was collected over 11 days. During the last four days, 36 attack scenarios were executed, which targeted various attack points, including physical sensors, actuators, and network communication infrastructure. The WADI testbed is an extension of the SWaT testbed with the inclusion of additional components such as analyzers, booster pumps, and chemical dosing systems (Fährmann et al., 2022). WADI is significantly larger than SWaT, consisting of 126 discrete and continuous variables collected over 14 days under normal conditions and for an additional 2 days under various attack scenarios.

Following (Fährmann et al., 2022), we perform the following preprocessing. Since (Goh et al., 2017) noted that the systems only reach stabilization after at most 6 hours, these initial samples are removed from both datasets. Since both datasets are very large, we subsample the time intervals to 1 sample per second, which is expected to capture all relevant variation present in the data. In (Kravchik and Shabtai, 2021), the Kolmogorov-Smirnov test (Karon, 1968) was used to show that there is a subset of variables whose distributions do not match across the training and validation sets. This is an issue in the context of anomaly detection, since a required assumption for learning the nominal data distribution is that it is stationary. We filter out the identified troublesome variables (19 from WADI and 15 from SWaT) and any constant variables as they cannot be appropriately modeled. Finally, as noted in (Kravchik and Shabtai, 2021), the value of the sensor $2B_{ALT,002,PV}$ in WADI becomes one thousand times larger in the middle of the test data without triggering an anomaly flag, so we additionally remove this variable from our data. For both datasets, we take a 90-10 split of train/validation data.

In order to learn the causal graphs for datasets, we utilize the stable PC algorithm (Colombo, Maathuis, and others, 2014) with a partial correlation based conditional independence test as outlined in (Runge et al., 2019). For our method, we model the root nodes by adding self-edges as discussed earlier.

Evaluation Metric Kim et al. (2022) demonstrates that the commonly used point-adjusted anomaly metric is effectively useless since when it is used for performance comparison, untrained anomaly models can outperform trained state-of-the-art detectors. Instead they propose using either F1 or “point-adjust@k” metrics. Here, we follow their advice and report only the unadjusted F1 scores. Since all methods require a threshold to be set in order to classify anomalies, we pick the threshold to be the one which produces the maximal F1 score for each method. In order to provide a thorough analysis of the behavior of each model, we additionally report the Precision-Recall curves for each dataset.

Baseline methods

Isolation Forest (IF) (Liu, Ting, and Zhou, 2008): This method works by attempting to separate each data point using decision trees. The number of branching points above a given data point provides a measure of how anomalous that data point is – fewer splits indicate the data is more separated from the overall dataset and is therefore more anomalous.

Graph Deviation Network (GDN) (Deng and Hooi, 2021): This method utilizes a learned node-embedding scheme to disambiguate sensors of different characteristics, which is leveraged through an attention-based network. This method is a strong baseline for our model, as it similarly learns a graph structure and was shown in Kim et al. (2022) to be the only method to reliably outperform simple baselines under appropriate evaluation metrics. We keep all default parameters utilized in the original paper for each of the respective datasets, which we obtained from the GDN github comments by the authors.

Causal Conditional Variational Autoencoder (C-CVAE) (Yang, Zhang, and Hoi, 2022): The authors of this Arxiv pre-print introduce a general causality-based anomaly detection framework in which they assume an input causal graph, train a local causal mechanism model for non-root nodes and apply separate model to model the marginal distributions of each root-node. Among the methods tested, they show that the optimal choices of the models for the non-root/root nodes are independent conditional variational
apply these models and train their pipeline using the hyperparameters described in the pre-print. Note, however, that we use our constructed causal graph to allow for a more direct comparison with our method.

For our architecture, all networks are feed-forward with ReLU activations and we take $\tau = 10$. $M_{\text{in}}$ has hidden layer dimensionality $[32, 16, 8]$ and $[64, 32, 16]$ for SWaT and WADI respectively. We take the hidden layer dimensionality of $M_{\text{pred}}$ to be $[10, 20, 10]$ for every $k$. In order to train our method, we utilize Adam optimization and train for 500 epochs, with learning rate 0.01, patience of 50, and 100 epochs of burn-in. For each of the $W$-tuning variants, we apply this training process, by modifying the learning rate to be 0.001 and initializing with the baseline model. For the DAG-constrained models, we set $\gamma = 1$.

Results We train each method 5 times with different random seeds in order to obtain standard-deviation estimates for each metric. The results from this experiment are shown in Table 1. We note that Causanom performs at least as well as the other baselines and, as expected, both Causanom tuning variants have improved performance across all metrics as compared to the base Causanom model, however we find no decisive optimal tuning strategy. The Precision-Recall curves for both SWaT and WADI are shown in Figure 2. Note, we truncate the domain of Figure 2 (a) to the beginning of deviation in the curves. From this figure we see that all Causanom variants dominate the baseline methods and GDN, C-CVAE, and IF are fairly well distinguished in descending order of dominance. Figure 2 (b) shows a less clear dominance order, however the Causanom variants show clear dominance for low-recall, high-precision values and all methods have similar performance in the high-recall setting.

<table>
<thead>
<tr>
<th></th>
<th>SWaT</th>
<th>WADI</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(F1-best)</td>
<td>(F1-best)</td>
</tr>
<tr>
<td>IF</td>
<td>72(0.6)</td>
<td>26.4(1.6)</td>
</tr>
<tr>
<td>GDN</td>
<td>77.5(1.8)</td>
<td>24.5(1.8)</td>
</tr>
<tr>
<td>C-CVAE</td>
<td>72.1(1)</td>
<td>25.2(0.1)</td>
</tr>
<tr>
<td>Causanom</td>
<td>78.8(1)</td>
<td>32.3(1.7)</td>
</tr>
<tr>
<td>Causanom+tuning</td>
<td>79.5(0.9)</td>
<td>35.2(3.2)</td>
</tr>
<tr>
<td>Causanom+DAGtuning</td>
<td>78.6(0.6)</td>
<td>35.3(2.7)</td>
</tr>
</tbody>
</table>

Table 1: Precision (P), Recall (R), and F1-score results from our anomaly detection experiment on the SWaT and WADI datasets. Parenthetical values indicate 1-standard deviation for each metric.

Anomaly Attribution

Synthetic Dataset In order to test the edge-anomaly attribution scheme, we require access to ground-truth failure attributions. Since these are not available for any publicly available datasets, we construct a synthetic dataset and induce failures by perturbing the causal graph. Following the scheme outlined in (Runge, 2020), we construct a random time-windowed graph with 10 nodes along 10 timesteps. To ensure similar edge sparsity across graphs, we restrict the average in-degree of the nodes to be 0.2 and proceed to construct the graph by iteratively sampling from all possible edges until the requisite number of edges are obtained. We adopt this edge sampling scheme so that we never add a cycle inducing edge to the summary graph (i.e., when collapsed over temporal lags, the causal graph is still a DAG). In order to generate the time series data once the graph has been constructed, we take the edge relationships to be $x_k(t) = \left(\sum_{(n,\tau)\in PA(k)} a_{kn}^\tau x_n(t-\tau)\right) + \eta_k^\tau$, for each $k \in N_c$. Here, $a_{kn}^\tau$ are uniformly chosen on $\pm[1, 5]$ and L1-normalized across $(n, \tau)$, and $\eta_k^\tau \sim N(0, 0.5)$ are independently sampled in $\bar{k}$ and $t$. In order to drive the system, we take the root-nodes to be sinusoids of varying frequencies. After generating the nominal data with the method outlined above, we can induce a broken edge anomaly between nodes $k$ and $(k', \tau) \in PA(k)$ by setting $a_{kn}^\tau = 0$.

In our experiments, we aim to determine the efficacy of our broken edge identification method. Our method can only identify a broken edge if it is modeled by $W$, therefore, we test the effect that fine tuning $W$ has on our results in settings where the estimated causal graph is imperfect. We construct the corrupted graph edges, $\tilde{E}_{cd}$, with edit distances
A histogram of the accuracy of the broken edge anomaly were modified by $E$ results over strong baselines via graph fine tuning. We also a system. We showed how this flexibility can lead to superior tion method which can flexibly model the causal structure of In this paper we introduced a causality-based anomaly detec-

tion of the true causal graph. As an explanation, we observe

that for cases in which the graph corruptions do not preserve

equality, most smooth paths which transform the corrupted graph into the true graph require a DAG violation. Therefore, the DAG-constraint may restrict the loss to a local optimum in such cases. More work is required to determine how to optimally enforce such a constraint in a continuous setting.

Conclusion

In this paper we introduced a causality-based anomaly detec-
tion method which can flexibly model the causal structure of a system. We showed how this flexibility can lead to superior results over strong baselines via graph fine tuning. We also introduced a method of edge attribution by utilizing this flex-
ibility to produce counterfactual estimates for broken edge anomalies.

To the authors’ knowledge, this is the first method to invest-
gate the problem of edge-based anomaly attribution. As a result, there are many directions for investigation in this area. For example, here we assume that it is already known that an edge anomaly has occurred, however, future work could focus on how to determine the type of anomaly directly from the data stream. Additionally, both multiple-edge and more functionally complex edge failures should be considered to increase versatility of the system for various applications. This problem poses a significant challenge as the search space for such edge failures is infinite. Additional work could be done to improve the causal structure estimation presented here. For example, imposing a hard DAG constraint could be considered or ancestor constraints could be incorporated into the loss term.

References


Kou, Y.; Lu, C.-T.; Sirwongwatana, S.; and Huang, Y.-P. 2004. Survey of fraud detection techniques. In IEEE Inter-

$i = 0, 1, 5, 10$ from the ground truth graph $G$, by randomly adding or removing edges for a selection of non-root nodes $N \subseteq N_c$ in $G$, while ensuring that $E_i \subseteq \hat{E}_{10}$ for all $i$. In order to target the corrupted edges when inducing the failures, we only allow failures to occur on the nodes whose in-edges were modified by $\hat{E}_{10}$.

**Results** We run 10 different simulations to construct the nominal data. For each experiment, break a random edge of each of the nodes in $N$ across 30 runs. For each failure, we utilize the above methodology to predict the broken edge. A histogram of the accuracy of the broken edge anomaly detection is shown in Figure 3 along with standard deviation error bars for the mean of each model. Existing methods (Li et al., 2022; Yang, Zhang, and Hoi, 2022) aim to identify the root cause of an anomaly by considering the anomaly score of the parents of an anomalous node. We show the output of these methods as a baseline (red bar) in the figure.

As can be seen in the figure, our method significantly outperforms the baseline methods for all edit distances. Additionally, we see improvements in the scores of the tuned graphs for the more corrupted experiments as expected. Note however that the DAG constraint does not improve performance over the method of tuning without constraints, which is unexpected since the DAG-constraint should facilitate modeling of the true causal graph. As an explanation, we observe that for cases in which the graph corruptions do not preserve the topological ordering, most smooth paths which transform the corrupted graph into the true graph require a DAG violation. Therefore, the DAG-constraint may restrict the loss to a local optimum in such cases. More work is required to determine how to optimally enforce such a constraint in a continuous setting.

Figure 3: Broken edge anomaly identification accuracy for the Causanom baseline method, Causanom unconstrained/DAG-constrained fine-tuned variants, and the baseline methods.


