COGS: A Causal Representation Learning Framework for Out-of-Distribution Generalization in Time Series

Xinxin Song[1,*], **Yuxiao Cheng**[1,*], **Tingxiong Xiao**[1], **Jinli Suo**[1,2,†]

[1]Department of Automation, Tsinghua University, Beijing 100084, China
[2]Institute for Brain and Cognitive Science, Tsinghua University, Beijing 100084, China
jlsuo@tsinghua.edu.cn

Abstract

Time series analysis is crucial in various fields such as healthcare and finance. However, environmental variations and the inherent non-stationarity of time series data often lead to out-of-distribution (OOD) scenarios, consequently causing model performance degradation. Most existing OOD generalization methods primarily focus on images or text, leaving time series analysis relatively underexplored. In this paper, we propose COGS, a novel framework that incorporates causal representation learning into the OOD generalization of time series. By imposing structural priors, our method identifies latent variables and learns a causal graph to disentangle causal variables from non-causal ones. These causal variables are then used to learn domain-invariant representations for stable prediction. Moreover, to tackle the challenge of the absence of domain labels, we further introduce a prototype-based domain discovery algorithm that infers domain labels in an unsupervised manner. The entire framework is optimized in a two-phase iterative manner, resulting in robust OOD performance. Extensive experiments on multiple real-world time series datasets demonstrate that our method achieves competitive performance compared to baseline methods.

Introduction

Time series data is prevalent across various fields, including disease diagnosis (Tomašev et al. 2019) and weather prediction (Wu et al. 2023). However, the complexity of real-world scenarios and the non-stationarity of time series (Wu et al. 2007) can lead to shifts in data distribution, which often results in a significant decline in the performance of machine learning models when applied to unseen situations, i.e., out-of-distribution (OOD) generalization problem (Wang et al. 2023; Wu et al. 2025).

In recent years, numerous studies have explored the issue of OOD generalization; however, a majority of these focus primarily on image or text data (Zhou et al. 2023). Although some studies have begun to explore OOD generalization methods for time series and have shown promising results (Ragab et al. 2022; Lu et al. 2023; Shi et al. 2024), most of them overlook the underlying causal mechanisms that govern the temporal generation process. In particular, they mainly focus on learning domain-invariant representations directly from raw time series without explicitly identifying and disentangling the underlying causal variables behind the data, which may introduce spurious correlations between non-causal variables and the target. These spurious correlations are usually unstable and change with the environment. For example, in patient monitoring, the blood oxygen level has a stable causal effect on respiratory failure, while the hospital ID may spuriously correlate with outcomes due to differences in treatment quality or protocols across hospitals. If the model uses these spurious correlations for prediction, it will have difficulty generalizing to new environments. In contrast, the relationship between causal variables and the target is stable across domains. Therefore, if we can identify and disentangle latent causal variables from non-causal variables and only use causal variables for prediction, we can achieve stable OOD generalization. In addition, the domain labels of time series data are usually missing, existing methods usually use manually specified domain labels for domain-invariant representation learning, which may be inaccurate and coarse-grained (Creager, Jacobsen, and Zemel 2021). Since non-causal variables in the data usually contain unique information about the domain, if we can utilize this domain-specific information, more accurate domain label inference can be achieved.

Inspired by recent research in temporal causal representation learning (Yao, Chen, and Zhang 2022), we propose Causal OOD Generalization for time-Series (COGS), a framework that learns causally invariant representations to achieve OOD generalization of time series without requiring domain labels. Specifically, to identify the true latent variables from observations, we analyze the underlying dynamics of time series data and impose structural causal equation constraints on the encoded latent variables. Then, we design a causal graph discovery strategy to disentangle causal and non-causal latent variables, and only use causal variables to learn invariant representations for prediction. To address the lack of domain labels, we design a prototype-based domain inference algorithm that leverages non-causal variables to infer domain labels, which in turn facilitates causal graph discovery and the learning of invariant representations. The main technical contributions of our method are summarized as follows:

[*]These authors contributed equally.
[†]Corresponding author

- We propose COGS, a causal representation learning framework for time series OOD generalization. By combining prior estimation with latent causal graph discovery, COGS identifies the underlying causal variables from observational time series, thereby enabling the learning of causally invariant representations.
- We propose a novel prototype-guided domain inference strategy that infers domain labels from non-causal latent variables, which in turn facilitate subsequent causal representation learning and graph discovery.
- Extensive experiments on multiple datasets demonstrate that our method achieves competitive performance against state-of-the-art OOD generalization methods.

Related Work

OOD Generalization. OOD generalization aims to train models that perform well on unseen domains (Wang et al. 2023). A mainstream approach is invariant representation learning, which seeks stable domain-invariant features through domain adversarial training (Ganin et al. 2016), regularization (Arjovsky et al. 2019; Krueger et al. 2021), and knowledge distillation (Lu et al. 2022). Other directions include data augmentation (Tobin et al. 2017; Zhang et al. 2018) and distributionally robust optimization (Sagawa et al. 2020). Most OOD methods rely on known domain labels, with only a few exceptions. To infer domain labels from data, EIIL (Creager, Jacobsen, and Zemel 2021) stratifies examples into discrete bins to maximize violations of the invariance principle; HRM (Liu et al. 2021) proposes a two-stage iterative optimization framework and designs a Gaussian distribution-based clustering algorithm to infer domain labels; ZIN (Lin et al. 2022) leverages auxiliary information to facilitate domain discovery. AdaRNN (Du et al. 2021) and Diversity (Lu et al. 2023) leverage distributional metrics and adversarial training for domain inference. This work also supports inferring domain labels from data.

Time-series OOD Generalization. Among all OOD generalization methods, few are tailored to time series. GILE (Qian, Pan, and Miao 2021) adopts a VAE framework and a domain classifier to separate domain-invariant and domain-specific features. CTSDG (Hu et al. 2022) incorporates prior knowledge into a variational framework to build a causal graph. CCDG (Ragab et al. 2022) uses class-conditional contrastive loss to constrain logits, while ITSR (Shi et al. 2024) employs learnable orthogonal decomposition for feature separation. AdaRNN (Du et al. 2021) performs domain splitting and temporal distribution matching in two separate stages to achieve OOD generalization. Building on this, Diversity (Lu et al. 2023) proposes an end-to-end adversarial training framework. TTSO (Jian, Yang, and Jiao 2024) further formulates a tri-level optimization problem solvable by large language models. However, most of the existing time-series OOD methods do not explicitly identify and disentangle the causal variables behind the data to eliminate spurious correlations between non-causal variables and the target. Our method, on the other hand, is grounded in the causal generative process of time series, i.e., temporal causal representation learning.

Causal Representation Learning. Causal representation learning seeks to uncover latent variables from observed data by modeling generative processes through Structural Causal Models (SCMs) (Schölkopf et al. 2021). For static data, this often involves interventions and acyclicity constraints(Vowels, Camgoz, and Bowden 2022). In contrast, time series provide temporal dependencies and nonstationarity that facilitate identification. For example, LEAP (Yao et al. 2022) exploits nonstationary noise, while CITRIS (Lippe et al. 2022) uses auxiliary interventions. TDRL (Yao, Chen, and Zhang 2022) provides identifiability theory and a variational framework for both stationary and non-stationary settings, though still requiring domain labels. Extending this, NCTRL (Song et al. 2023) uses hidden Markov models to infer unknown domains. Building on TDRL, our method is designed for time series OOD generalization problem and further employs a prototype contrastive clustering algorithm to infer unknown domain labels.

Background

OOD Generalization for Time Series

We first denote a univariate or multivariate time series $\mathbf{x}$ of length T as $\mathbf{x} = (x_1, x_2, \ldots, x_T) \in \mathbb{R}^{D\times T}$, where D is the number of variables. Let $\mathcal{X} \subset \mathbb{R}^{D\times T}$ and $\mathcal{Y} \subset \mathbb{N}$ denote the input and label spaces, respectively. The standard time series classification (TSC) task aims to learn a mapping function $f : \mathcal{X} \to \mathcal{Y}$ from a training dataset $\mathcal{D}_{\text{train}} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, such that f generalizes well to a test set drawn from the same underlying distribution as the training data.

In contrast, the OOD generalization setting assumes that the training data are collected from M different domains, which can be denoted as $\mathcal{D}_{\text{train}} = \{\mathcal{D}^1, \ldots, \mathcal{D}^M\}$. Each domain $\mathcal{D}^i = \{(\mathbf{x}_j^i, y_j^i)\}_{j=1}^{n_i}$ consists of n_i independent and identically distributed samples drawn from the domain-specific distribution $P_i(\mathbf{X}, Y)$. The goal of OOD generalization is to learn a model f from $\mathcal{D}_{\text{train}}$ that generalizes well to an unseen test domain whose distribution $P_{\text{test}}(\mathbf{X}, Y)$ is different from any of the training distributions, i.e., $P_{\text{test}} \neq P_i$ for all $i \in \{1, \ldots, M\}$. Formally, the objective becomes

$$\min_f \mathbb{E}_{(\mathbf{x},y)\sim P_{\text{test}}(\mathbf{X},Y)}[\mathcal{L}(f(\mathbf{x}), y)], \tag{1}$$

where $\mathbb{E}$ is the expectation and $\mathcal{L}$ is the loss function.

Temporal Causal Generative Process

For a more in-depth analysis of the time series OOD generalization problem, we first consider the generative process of time series across multiple domains. Specifically, we assume that the observed time series $\mathbf{x}_t \in \mathbb{R}^D$ at time t is generated from the latent variable $\mathbf{z}_t \in \mathbb{R}^N$ through an invertible and potentially non-linear mixing function $\mathbf{g}$:

$$\mathbf{x}_t = \mathbf{g}(\mathbf{z}_t). \tag{2}$$

The dynamics among latent variables in each domain can be modeled using a temporal SCM, in which each latent variable $z_{t,i}$ is generated from its parent variables and independent exogenous noise. For nonstationary cases, this generation mechanism may vary across domains, resulting in distribution shifts. To represent the non-stationary generation process, we incorporate the domain-specific low-dimensional

change factor $\boldsymbol{\theta}_d$ into the generation process, which can be formally expressed as

$$z_{t,i} = h_i\left(\text{pa}(\mathbf{z}_{t,i}), \boldsymbol{\theta}_d, \epsilon_{t,i}\right), \tag{3}$$

where $\text{pa}(\mathbf{z}_{t,i})$ denotes the time-delayed causal parents of $z_{t,i}$, i.e., $\text{pa}(\mathbf{z}_{t,i})$ is any subset of $\{z_{\tau,j} \mid \tau \in \{t-L, t-L+1, \ldots, t-1\},\ j \in \{1, 2, \ldots, N\}\}$ with L denoting the maximum time delay of the causal effects. $\boldsymbol{\theta}_d$ has a constant value in each domain but varies across domains. $d \in \mathcal{E}_{train} = \{1, \ldots, M\}$ represents the domain index. $\epsilon_{t,i} \sim p_i(\epsilon)$ is the exogenous noise term. We assume that the noise terms are mutually independent and that there are no instantaneous causal relationships among latent variables.

To achieve robust OOD generalization for time series, it is crucial to identify the underlying latent variables that generate the observed data, from which we can extract causal representations that remain invariant across domains. Next, we will present the core assumptions that support the identification of latent variables from observed time series data.

Identifiability of Latent Variables

Recent studies have utilized temporal structures and nonstationarities to achieve the identifiability of latent variables in time series analysis. Yao, Chen, and Zhang (2022) proved that given certain assumptions, the latent process and variables are identifiable. In the following, we provide identifiability assumptions that allow recovery of latent variables across domains, adapted from Theorem 2 in TDRL (Yao, Chen, and Zhang 2022).

Theorem 1 ***(Simplified)*** *Consider the data generation process described in Eqs.* (2) *and* (3)*. Let* $\eta_{tk}(d) \triangleq \log p(z_{t,k} \mid \{\mathbf{z}_{t-\tau}\}_{\tau=1}^{L}, d)$ *denote the condition distribution in domain* d*. Suppose that there exists an invertible function* $\hat{\mathbf{g}}^{-1}$ *that maps* $\mathbf{x}_t$ *to* $\hat{\mathbf{z}}_t$*, i.e.,* $\hat{\mathbf{z}}_t = \hat{\mathbf{g}}^{-1}(\mathbf{x}_t)$*. If the nonstationary domains* d *induce sufficiently complex and diverse influences on the latent transition dynamics, and such diversity can be characterized by the linear independence of vectors composed of higher-order and mixed partial derivatives of* $\eta_{tk}(d)$ *with respect to the latent variables across domains. Under this condition, the estimated latent variables* $\hat{\mathbf{z}}_t$ *must be an invertible, component-wise transformation of a permuted version of the true latent variables* $\mathbf{z}_t$*. In other words, the true latent processes can be uniquely identified.*

A more rigorous mathematical formulation of Theorem 1 and the corresponding proof can be found in Supplementary Section A.2. Intuitively, this theorem states that if the data is generated from a relatively large number of distinct domains, then the latent variables can be uniquely recovered.

Once the latent variables have been successfully recovered, we can further identify and extract the causal variables, i.e., the latent variables that have a direct causal relationship with the target variable Y. These causal variables are invariant across domains and can thus serve as robust features for OOD generalization. Further details will be discussed in the next section.

Method

In this section, we introduce COGS, a novel framework that autonomously discovers and utilizes stable causal variables from time series for robust OOD prediction, even in the absence of domain labels. The core idea is to iteratively identify latent variables and decompose them into two parts: causal parts, which exhibit invariant causal effects across domains and are causally related to the target Y, and non-causal parts, which capture domain-specific variations causally unrelated to the target. To achieve this target, we specifically build three key modules: (1) A *latent variable identification module* that identifies latent variables from observations across diverse domains; (2) A *latent causal graph discovery module* that discovers causal relationships from latent variables for robust OOD prediction; and (3) A *prototype-guided domain discovery module* that leverages non-causal variables to infer domain labels via prototypical contrastive learning. The overall framework of our method is illustrated in Figure 1. Through iterative optimization across the three components, the model finally achieves stable OOD prediction.

Latent Variable Identification Module

The previous section has shown that, under certain assumptions, the latent variables underlying the data-generating process are identifiable from the observed temporal data. To facilitate latent variable identification, we adopt a VAE-based framework (Song et al. 2023). Moreover, a prior network is designed to impose the structural prior constraints on the learned latent variables.

Sequential VAE To satisfy the constraint in Eq. (2) that assumes a nonlinear and invertible generative process, we adopt a variational framework with a sequential encoder-decoder architecture. The encoder $\text{enc}(\cdot)$ infers the posterior distribution $q(\hat{\mathbf{z}}_{1:T} \mid \mathbf{x}_{1:T})$, from which latent variables are sampled using the reparameterization trick. The decoder $\text{dec}(\cdot)$ reconstructs the input as $\hat{\mathbf{x}}_{1:T} = \text{dec}(\hat{\mathbf{z}}_{1:T})$. This facilitates the extraction of temporally structured latent variables suitable for causal modeling.

Prior Network Assuming that the latent process follows the SCM described in Eq. (3) and the latent variables are conditionally independent given their parents. Let $\{\hat{\mathbf{z}}_{t-\tau}\}$ denote the lagged latent variables up to maximum time lag L. To estimate the prior $p(\hat{\mathbf{z}}_{1:T} \mid d)$ of latent variables in each domain, we learn a set of inverse transition functions $\{m_i\}_{i=1}^{N}$, which receive the lagged latent variables $\{\hat{\mathbf{z}}_{t-\tau}\}$, current state $\hat{z}_{t,i}$ and domain change factor $\boldsymbol{\theta}_d$ as input and output the corresponding noise terms $\hat{\epsilon}_{t,i}$, i.e., $\hat{\epsilon}_{t,i} = m_i\left(\{\hat{\mathbf{z}}_{t-\tau}\}, \boldsymbol{\theta}_d, \hat{z}_{t,i}\right)$. We can therefore deduce that:

$$p\left(\hat{\mathbf{z}}_t | \{\hat{\mathbf{z}}_{t-\tau}\}, d\right) = p\left(\hat{\epsilon}_t | \{\hat{\mathbf{z}}_{t-\tau}\}, d\right) \cdot |\mathbf{J}|, \tag{4}$$

where $\mathbf{J} = \begin{bmatrix} I_{N\times L} & 0 \\ * & diag(\frac{\partial m_i(\cdot)}{\partial \hat{z}_{t,i}}) \end{bmatrix}$ is a low-triangular Jacobian matrix. Since the noise terms are mutually independent, the transition prior at each time point can be expressed as

$$\log p(\hat{\mathbf{z}}_t | \{\hat{\mathbf{z}}_{t-\tau}\}, d) = \sum_{i=1}^{N} \left(\log p(\hat{\epsilon}_{t,i} | d) + \log \left| \frac{\partial m_i(\cdot)}{\partial \hat{z}_{t,i}} \right| \right). \tag{5}$$

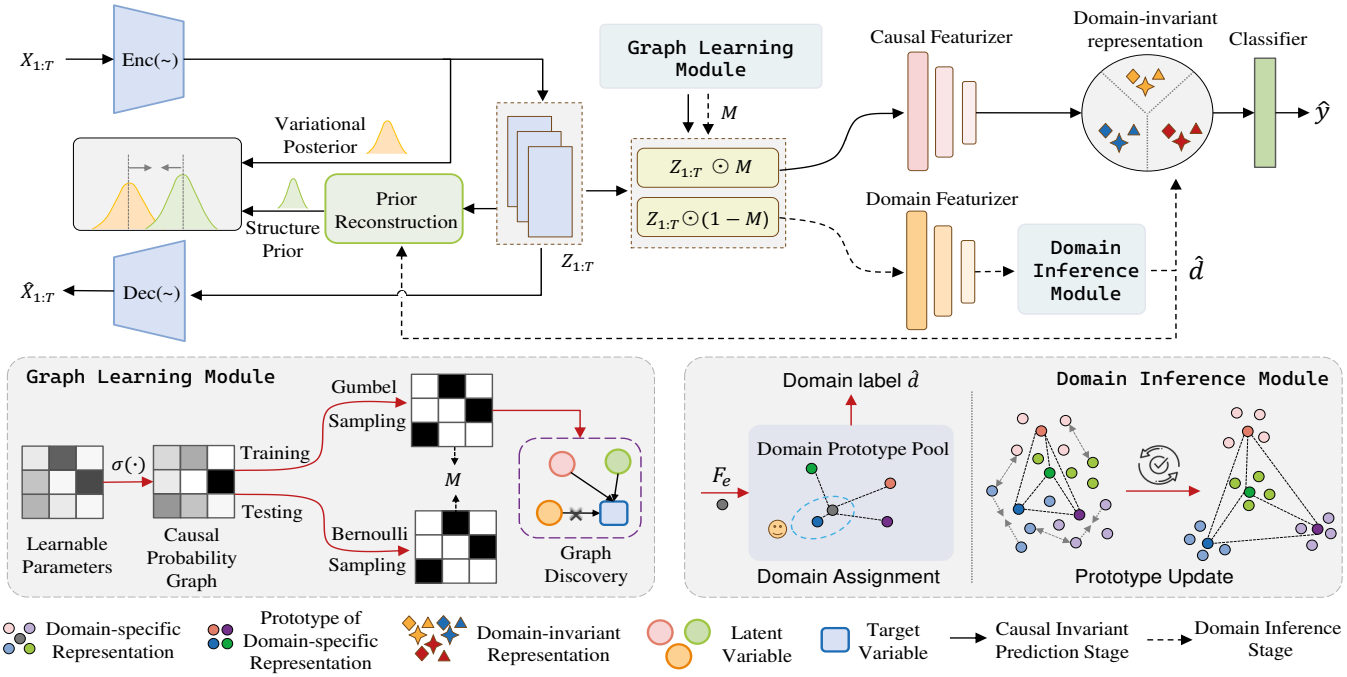


Figure 1: Overview of our method. The framework is optimized by alternating two stages. The ***Causal Invariant Prediction Stage*** recovers and disentangles latent variables from the observed time series, and then uses the discovered causal variables to learn invariant representations. The ***Domain Inference Stage*** uses domain-specific variables to infer domain labels through a prototype clustering algorithm.

Finally, we can get the prior of latent variables:

$$\log p(\hat{\mathbf{z}}_{1:T} \mid d) = \log p(\hat{\mathbf{z}}_{1:L} \mid d) + \sum_{t=L+1}^{T} \log p(\hat{\mathbf{z}}_t | \{\hat{\mathbf{z}}_{t-\tau}\}, d). \tag{6}$$

Detailed derivations of the above process can be found in supplementary Section A.3. For the implementation of the network, we use simple MLPs to learn $\{m_i(\cdot)\}_{i=1,\cdots,N}$ and use an embedding layer to encode domain label d into $\boldsymbol{\theta}_d$. We assume that the noise term $p(\hat{\epsilon}_{t,i} \mid d)$ and init prior $p(\hat{\mathbf{z}}_{1:L} \mid d)$ follow Gaussian distributions.

We compute the mean-square error loss between $\mathbf{x}_{1:T}$ and $\hat{\mathbf{x}}_{1:T}$ to reconstruct the time series observations. To ensure the recovery of true latent variables, we compute the Kullback-Leibler (KL) Divergence between the posterior and the estimated prior via a sampling approach similar to Song et al. (2023). The loss can be written as

$$\mathcal{L}_{\text{TCRL}} = \mathbb{E}_{\hat{\mathbf{z}}_t} \sum_{t=1}^{T} -\log p(\hat{\mathbf{x}}_t \mid \hat{\mathbf{z}}_t) + \alpha \cdot D_{KL}\left(q(\hat{\mathbf{z}}_{1:T} \mid \mathbf{x}_{1:T}) \,||\, p(\hat{\mathbf{z}}_{1:T} \mid d)\right), \tag{7}$$

where the first term represents the MSE loss and the second term represents the KL loss, α is the hyperparameter used to balance these two losses.

Latent Causal Graph Discovery Module

Even after identifying the latent variables from observed temporal data, not all of them are causally related to the target label Y. Some latent variables encode domain-specific information that is spuriously correlated with Y and may shift across domains, thereby impairing OOD generalization. This highlights the importance of disentangling causal variables that are invariant across domains from non-causal or domain-specific variables in the latent space. Formally, we give the following definition:

Definition 1 (Causal vs. Non-causal Latent Variables) *Given a target variable Y, the set of latent variables $\mathbf{Z} = \{Z^1, Z^2, \ldots, Z^N\} \in \mathbb{R}^{N \times T}$ can be partitioned into two disjoint subsets:*

- ***Causal Latent Variables*** *($\mathbf{Z}^c$): Variables having direct causal influence on the target Y, with the causal relationship assumed to be invariant across all domains $d \in \{1, \ldots, M\}$ and stable under distribution shifts.*
- ***Non-causal Latent Variables*** *($\mathbf{Z}^s$): Variables that are spuriously correlated with Y, with the correlation typically capturing domain-specific information and varying across domains.*

To model the causal relationships between latent variables and the target Y, we implement the causal graph as *causal probability graph* $\mathcal{G}$, where each element $\mathcal{G}_i$ denotes the probability that latent variable Z^i is the cause of Y, i.e., $\mathcal{G}_i = P(Z^i \to Y)$, for $i = 1, \ldots, N$. Note that this causal probability graph is *temporally aggregated*, that is, if a latent variable Z^i is determined to be causally related to Y, then all its values across the time window will be used to predict Y.

Causal Graph Learning For graph discovery, we initialize a learnable parameter set Θ, which is transformed via a sigmoid function to obtain the causal probability graph $\mathcal{G} = \{p_i\}_{i=1}^N$, where $p_i \in [0,1]$ represents the probability that latent variable $\mathbf{Z}^i$ is causally related to the target Y.

To select potential causal variables for prediction, we sample a binary mask $\widetilde{\mathbf{M}} = \{s_i\}_{i=1}^N$ from $\mathcal{G}$, where each $s_i \sim \text{Bernoulli}(p_i)$, and $s_i = 1$ indicates that $Z^i \in \text{pa}(Y; \mathbf{G})$. During training, since the discrete sampling is non-differentiable, we employ the Gumbel-Softmax trick (Jang, Gu, and Poole 2017) to obtain a differentiable approximation, thereby enabling end-to-end training, i.e.,

$$s_i = \frac{\exp\left(\left(\log(p_i) + g_i\right)/\tau\right)}{\sum_{i=1}^N \exp\left(\left(\log(p_i) + g_i\right)/\tau\right)}, \tag{8}$$

where g_i is the Gumbel noise and τ is the temperature parameter.

Causally Invariant Prediction Given the sampled binary mask $\widetilde{\mathbf{M}} \in \{0,1\}^N$ that indicates the selection of causal variables among N latent variables, we broadcast it along the temporal dimension to obtain the time-expanded binary mask $\mathbf{M} \in \{0,1\}^{N \times T}$. Then we define the causal latent variables as $\mathbf{Z}^c = \mathbf{Z} \odot \mathbf{M}$, where $\odot$ denotes the Hadamard product. This operation ensures the consistency of variable selection: once a variable is identified as a causal variable, its dynamic information over the entire time window will be fully preserved for subsequent prediction. A causal-invariant representation $\mathbf{F}_c$ is then extracted and will be used to predict the target Y. This can be expressed as

$$\mathbf{F}_c = \text{feat}\left(\mathbf{Z}^c\right), \quad \hat{Y} = \text{cls}\left(\mathbf{F}_c\right), \tag{9}$$

where $\text{feat}(\cdot)$ denotes the feature extraction function and $\text{cls}(\cdot)$ denotes the classifier.

To enhance the invariance of $\mathbf{F}_c$ across different domains, we utilize a loss function that penalizes the variance of empirical risks across training domains while constraining the causal structure. The optimization objective is given by

$$\mathcal{L}_{\text{CIP}} = \sum_{d \in \mathcal{E}_{\text{train}}} \mathcal{L}_{\text{CE}}^{(d)} + \beta \cdot \text{Var}_{d \in \mathcal{E}_{\text{train}}}\left[\mathcal{L}_{\text{CE}}^{(d)}\right] + \lambda \cdot \|\mathcal{G}\|_1, \tag{10}$$

where $\mathcal{L}_{\text{CE}}^{(d)}$ denotes the cross-entropy loss on domain d. $\|\mathcal{G}\|_1$ is the L_1 norm favoring attention on causal variables while suppressing the influence of non-causal ones. β and λ are hyperparameters for balancing losses.

Prototype-Guided Domain Discovery Module

Identifying latent variables and learning invariant representations both require access to domain labels. However, such labels are often unavailable in practice. In this section, we introduce an unsupervised domain discovery method based on domain prototype clustering.

To capture domain-specific information, we utilize the non-causal latent variables $\mathbf{Z}^s$ extracted via $\mathbf{Z}^s = \mathbf{Z} \odot (1 - \mathbf{M})$, and obtain the domain feature $\mathbf{F}_e = \text{envFeat}(\mathbf{Z}^s)$ for each sample, where $\text{envFeat}(\cdot)$ is the domain feature extractor. Then we establish a domain prototype pool $\mathcal{V} = \{\mathbf{v}_i\}_{i=1}^K$, where each $\mathbf{v}_i \in \mathbb{R}^d$ characterizes the domain-specific pattern of prototype i. We employ an EM-style strategy to update the prototypes and infer domain labels.

Prototype Optimizing. We freeze all network parameters except the domain feature extractor and the causal graph, and we optimize domain feature extractor by minimizing the following prototype loss and contrastive loss:

$$\mathcal{L}_{\text{proto}} = \mathbb{E}\left[-\log \frac{\exp(\text{sim}(\mathbf{F}_e, \mathbf{v}_{\hat{d}})/\phi)}{\sum_{k=1}^K \exp(\text{sim}(\mathbf{F}_e, \mathbf{v}_k)/\phi)}\right], \tag{11}$$

$$\mathcal{L}_{\text{con}} = \mathbb{E}\left[-\log \frac{\sum_{j \in \mathcal{P}(i)} \exp(\text{sim}(\mathbf{F}_{ei}, \mathbf{F}_{ej})/\phi)}{\sum_{k \in \mathcal{A}(i)} \exp(\text{sim}(\mathbf{F}_{ei}, \mathbf{F}_{ek})/\phi)}\right], \tag{12}$$

where ϕ is the temperature parameter. $\mathcal{P}(i)$ represents the set of positive samples, which belong to the same domain, and $\mathcal{A}(i)$ represents the set of other samples in the batch. In addition, to minimize the mutual information between domain feature $\mathbf{F}_e$ and causal representation $\mathbf{F}_c$, we minimize the following correlation loss within the batch:

$$\mathcal{L}_{\text{ind}} = \left\| \frac{1}{B-1} (\mathbf{F}_c - \bar{\mathbf{F}}_c)^\top (\mathbf{F}_e - \bar{\mathbf{F}}_e) \right\|_F^2, \tag{13}$$

where $\bar{\mathbf{F}}_c = \mathbb{E}\left[\mathbf{F}_c\right], \bar{\mathbf{F}}_e = \mathbb{E}\left[\mathbf{F}_e\right]$. The overall optimization objective is:

$$\mathcal{L}_{\text{EI}} = \mathcal{L}_{\text{proto}} + \eta \cdot \mathcal{L}_{\text{con}} + \gamma \cdot \mathcal{L}_{\text{ind}} + \lambda \cdot \|\mathcal{G}\|_1. \tag{14}$$

Domain Assignment. We compute domain features $\{\mathbf{F}_e\}$ for all samples and perform K-means clustering to obtain current cluster centers $\{\mathbf{c}_1, \ldots, \mathbf{c}_K\}$. Then the prototypes are updated using an exponential moving average:

$$\mathbf{v}_k^t = m \cdot \mathbf{v}_k^{t-1} + (1-m) \cdot \mathbf{c}_k, \quad \forall k = 1, \ldots, K, \tag{15}$$

where $m \in [0,1)$ is a momentum parameter and v_k^t is the domain prototype of the t-th epoch.

Each sample is then assigned to a domain $\hat{d}$ based on similarity to the prototypes:

$$\hat{d} = \arg \max_{k \in \{1,\ldots,K\}} \text{sim}(\mathbf{F}_e, \mathbf{v}_k^t), \tag{16}$$

where $\text{sim}(\cdot,\cdot)$ can be cosine similarity or Euclidean distance.

Overall Architecture

Training We adopt a two-stage alternating optimization training process. In each training epoch, the process is divided into causal invariant prediction stage and domain inference stage. In the causal invariant prediction stage, we perform class prediction and the optimization objective is $\mathcal{L}_{\text{TCRL}} + \mathcal{L}_{\text{CIP}}$, as described in Eqs. (7) and (10). In the domain inference stage, we optimize $\mathcal{L}_{\text{EI}}$ described in Eq. (14) and assign domain labels to all samples. We prove that under certain assumptions, the learnable causal mask will converge to the true causal graph that can accurately distinguish causal variables from non-causal variables. Details can be found in Supplementary Section A.1.

Inference At inference time, we only use the discovered causal graph, encoder, feature extractor, and classifier to obtain predictions. Specifically, we use $\text{enc}(\cdot)$ to get the latent variable $\mathbf{Z}$ from $\mathbf{X}$, and then get the final prediction through $\hat{Y} = \text{cls}\left(\text{feat}\left(\mathbf{Z} \odot \mathbf{M}\right)\right)$. Therefore, no additional computational overhead is introduced during testing.

Methods	USC-HAD					PAMAP					WESAD				
	0	1	2	3	AVG	0	1	2	3	AVG	0	1	2	3	AVG
ERM	71.92	62.90	66.48	62.36	65.92	51.27	48.93	56.85	37.42	48.62	47.64	50.69	45.45	45.87	47.41
IRM	70.66	63.64	68.68	62.91	66.47	52.30	58.27	62.35	40.12	53.26	47.15	50.91	39.71	43.66	45.36
VREx	72.38	63.22	67.52	59.46	65.65	50.40	54.29	64.97	38.45	52.03	44.85	57.94	46.70	44.44	48.48
GroupDRO	72.17	59.37	69.40	60.30	65.31	51.17	52.54	60.67	32.59	49.24	**61.93**	39.22	58.38	30.11	47.41
IB-ERM	72.88	61.22	71.12	60.18	66.35	51.51	52.81	63.29	36.19	50.95	51.36	42.68	55.66	40.93	47.66
EIIL	73.11	62.18	64.52	61.07	65.22	54.18	52.83	57.32	31.54	48.97	50.61	54.58	56.19	43.81	51.30
DIFEX	70.75	65.13	67.74	61.00	66.16	59.27	54.29	64.26	38.74	54.14	47.32	**60.06**	53.46	50.65	52.87
AdaRNN	74.29	67.76	72.21	68.03	70.57	57.55	61.57	**70.06**	36.98	56.54	51.32	40.47	57.93	45.05	48.59
GILE	75.21	65.69	68.45	**70.03**	69.85	60.05	55.98	56.29	36.54	52.22	48.35	51.42	49.38	43.71	48.22
CCDG	71.73	63.57	66.12	55.61	64.26	51.04	32.45	47.44	38.94	42.47	53.74	40.44	52.36	50.02	49.14
Diversity	72.92	66.80	68.50	67.74	68.99	59.69	52.54	59.40	32.12	50.94	56.34	49.85	48.46	50.42	51.27
ITSR	73.45	64.08	67.08	60.68	66.32	57.76	56.88	51.90	40.71	51.81	42.48	55.74	48.99	41.75	47.24
TTSO	68.33	66.99	68.56	60.99	66.22	56.75	63.71	64.71	41.17	56.59	53.72	59.02	60.89	52.67	56.58
Ours	**75.99**	**68.78**	**73.97**	65.98	**71.18**	**67.25**	**64.10**	69.17	**42.18**	**60.68**	61.67	59.82	**61.56**	**53.12**	**59.04**

Table 1: Classification accuracy (%) on USC-HAD, PAMAP, and WESAD datasets. We highlight the best and the second best in bold and with underlining, respectively.

Methods	DSADS				
	0	1	2	3	AVG
ERM	77.50	70.48	85.83	78.73	78.14
IRM	71.54	65.57	86.89	80.66	76.17
VREx	74.47	69.25	85.22	75.75	76.17
GroupDRO	82.37	71.75	87.81	74.47	79.10
IB-ERM	78.82	75.26	87.06	75.79	79.23
EIIL	78.73	73.95	87.15	74.34	78.54
DIFEX	80.96	78.86	87.68	82.68	82.55
AdaRNN	**86.58**	74.69	83.60	77.32	80.55
GILE	77.11	75.66	88.95	78.33	80.01
CCDG	81.23	74.25	85.31	73.46	78.56
Diversity	82.63	77.72	82.54	81.67	81.14
ITSR	76.01	66.54	78.68	77.19	74.61
TTSO	81.71	71.05	79.08	80.13	77.99
Ours	83.38	**79.08**	**89.87**	**83.29**	**83.91**

Table 2: Classification accuracy (%) on the DSADS datasets. We highlight the best and the second-best in bold and with underlining, respectively.

Experiments

In this section, we conduct comprehensive experiments on multiple real-world datasets to evaluate the effectiveness of COGS. The source code and supplementary materials can be found at https://github.com/simon-sxx/COGS.

Experimental Settings

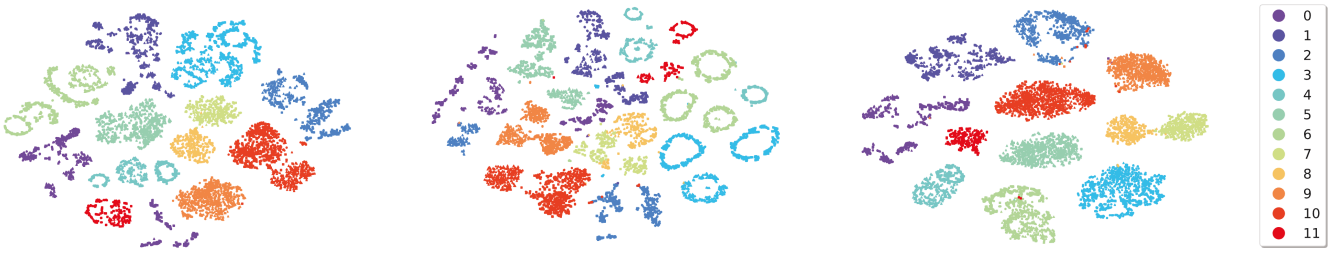
Datasets We conducted experiments on four representative real-world datasets, including USC-HAD (Zhang and Sawchuk 2012), DSADS (Barshan and Yüksek 2014), PAMAP (Reiss 2012), and WESAD (Schmidt et al. 2018). These datasets were partitioned into different domains to implement the OOD generalization setup, and we use the standard leave-one-domain-out setting (Gulrajani and Lopez-Paz 2021) for evaluation. Detailed information regarding the datasets and data pre-processing is provided in Section B.1 of the Supplementary.

Baselines To validate the effectiveness of COGS, we performed a comprehensive comparison with existing OOD generalization methods. We first compare general OOD generalization methods, including IRM (Arjovsky et al. 2019), VREx (Krueger et al. 2021), GroupDRO (Sagawa et al. 2020), IB-ERM (Ahuja et al. 2021), EIIL (Creager, Jacobsen, and Zemel 2021) and DIFEX (Lu et al. 2022); in addition, we also compare OOD generalization methods specifically designed for time series, including GILE (Qian, Pan, and Miao 2021), AdaRNN (Du et al. 2021), CCDG (Ragab et al. 2022), Diversity (Lu et al. 2023), ITSR (Shi et al. 2024) and TTSO (Jian, Yang, and Jiao 2024). Since TTSO has no public code, we reproduced it based on the original paper.

Implementation Details All the experiments are implemented with PyTorch (Paszke et al. 2019) on an NVIDIA RTX 3090 24GB GPU. For the implementation of $\text{enc}(\cdot)$ and $\text{dec}(\cdot)$, we use simple MLPs following Yao et al. (2022). As for $\text{feat}(\cdot)$ and $\text{envFeat}(\cdot)$, we use Convolutional Neural Networks (CNNs) similar to Lu et al. (Lu et al. 2023). For fairness, all methods (except GILE and AdaRNN) are implemented using the same encoder $\text{enc}(\cdot)$, feature extractor $\text{feat}(\cdot)$, and classifier $\text{cls}(\cdot)$. Detailed architecture and hyperparameters can be found in Supplementary Section B.2.

Experiment Results

Comparison with Baselines. The results for the USC-HAD, PAMAP, and WESAD datasets are shown in Table 1 and results for DSADS are shown in Table 2, where each subcolumn presents the results from leaving one domain out for testing, along with the corresponding average. Comprehensive experiments show that COGS consistently outperforms the baselines both in most domains and on average. On the PAMAP dataset, our method achieves an average performance improvement of 12.06% over ERM and outperforms the second-best baseline by 4.09% on average, which



(a) ERM (b) Diversity (c) Ours

Figure 2: t-SNE visualizations of the learned representations on PAMAP dataset, showing different classes with distinct colors and various domains by unique shapes.

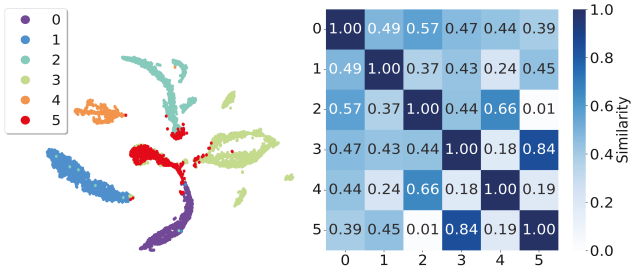
shows the effectiveness and superiority of COGS in improving the OOD generalization performance.

Ablation Study To demonstrate the effectiveness of each module in COGS, we conduct the following ablation experiments: (1) Using predefined domain labels, identifying and using all latent variables for prediction; (2) Using predefined domain labels, identifying and disentangling latent causal variables and non-causal variables, and only use causal variables for prediction; (3) Using non-causal variables to perform prototype learning and infer domain labels, that is, the complete COGS framework. The ablation study results are shown in Table 3. From the results of the ablation study, we can see that COGS outperforms all ablated versions in OOD classification accuracy, i.e., each module in COGS has a positive effect on the OOD generalization performance. In addition, after adding the domain inference module, the out-of-distribution accuracy on each dataset has been greatly improved, which shows the importance of accurate domain labels to improving OOD generalization performance.

$\mathcal{L}_{\text{TCRL}}$	$\mathcal{L}_{\text{CIP}}$	$\mathcal{L}_{\text{EI}}$	PAMAP	USC-HAD	WESAD
✗	✗	✗	48.62	65.92	47.41
✓	✗	✗	53.75	66.69	50.37
✓	✓	✗	55.88	67.41	51.96
✓	✓	✓	**60.08**	**71.18**	**58.09**

Table 3: Ablation studies for components in COGS, comparing the average classification accuracy (%) across domains.

Visualization To show that our method can extract causal invariant representations, we use t-SNE method (Van der Maaten and Hinton 2008) to visualize the learned representations, and the results are shown in Figure 2. It can be seen that our method can extract more compact intra-class representations while effectively separating clusters of different classes. To demonstrate that our method can uncover diverse domains, we also visualized domain-specific representations and calculated the cosine similarity matrix between different domain prototypes, as shown in Figure 3. It can be seen that domain-specific representations of different domains are well separated, and there are also obvious differences between different domains. Moreover, we provide $\mathcal{H}$-divergence among domains to further illustrate the advantages of our domain inference strategy, which can be found in Supplementary Section D.1.



(a) t-SNE Visualization of domain-specific features (b) Cosine similarity between domain prototypes

Figure 3: Visualizations of the domain inference stage.

Additional Experiments To analyze the impact of key parameters on OOD generalization performance, we conducted sensitivity analysis experiments on domain number K and the λ parameter that controls the degree of causal sparsity. Furthermore, we conduct statistical significance tests on the results and present additional visualizations to illustrate convergence, among other aspects. These results can be found in Supplementary Section D.2 and Section D.3.

Conclusion

In this paper, we propose COGS, a framework that integrates temporal causal representation learning into the time series OOD generalization problem. COGS effectively identifies and disentangles latent causal and non-causal variables, enabling stable prediction based only on causal variables. The domain inference strategy based on non-causal variables further promotes the identification of latent variables and the learning of domain-invariant representations. Extensive experiments on real-world datasets show that COGS achieves competitive performance compared to the state-of-the-art baselines. This work demonstrates the potential of understanding and solving OOD problems from the perspective of causal learning. A promising direction for future work is to extend the current framework to time series forecasting or anomaly detection problems in OOD scenarios.

Acknowledgments

This work is jointly supported by the National Key R&D Program of China (Grant No. 2024YFF0505703) and Beijing Municipal Natural Science Foundation (Grant No. L257009).

References

Ahuja, K.; Caballero, E.; Zhang, D.; Gagnon-Audet, J.-C.; Bengio, Y.; Mitliagkas, I.; and Rish, I. 2021. Invariance Principle Meets Information Bottleneck for Out-of-Distribution Generalization. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 3438–3450. Curran Associates, Inc.

Arjovsky, M.; Bottou, L.; Gulrajani, I.; and Lopez-Paz, D. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Barshan, B.; and Yüksek, M. C. 2014. Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units. *The Computer Journal*, 57(11): 1649–1667.

Creager, E.; Jacobsen, J.-H.; and Zemel, R. 2021. Environment Inference for Invariant Learning. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 2189–2200. PMLR.

Du, Y.; Wang, J.; Feng, W.; Pan, S.; Qin, T.; Xu, R.; and Wang, C. 2021. AdaRNN: Adaptive Learning and Forecasting of Time Series. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, 402–411. New York, NY, USA: Association for Computing Machinery. ISBN 9781450384469.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; March, M.; and Lempitsky, V. 2016. Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research*, 17(59): 1–35.

Gulrajani, I.; and Lopez-Paz, D. 2021. In Search of Lost Domain Generalization. In *International Conference on Learning Representations*.

Hu, Y.; Jia, X.; Tomizuka, M.; and Zhan, W. 2022. Causal-based Time Series Domain Generalization for Vehicle Intention Prediction. In *2022 International Conference on Robotics and Automation (ICRA)*, 7806–7813.

Jang, E.; Gu, S.; and Poole, B. 2017. Categorical Reparameterization with Gumbel-Softmax. In *International Conference on Learning Representations*.

Jian, C.; Yang, K.; and Jiao, Y. 2024. Tri-Level Navigator: LLM-Empowered Tri-Level Learning for Time Series OOD Generalization. In Globerson, A.; Mackey, L.; Belgrave, D.; Fan, A.; Paquet, U.; Tomczak, J.; and Zhang, C., eds., *Advances in Neural Information Processing Systems*, volume 37, 110613–110642. Curran Associates, Inc.

Krueger, D.; Caballero, E.; Jacobsen, J.-H.; Zhang, A.; Binas, J.; Zhang, D.; Priol, R. L.; and Courville, A. 2021. Out-of-Distribution Generalization via Risk Extrapolation (REx). In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 5815–5826. PMLR.

Lin, Y.; Zhu, S.; Tan, L.; and Cui, P. 2022. ZIN: When and How to Learn Invariance Without Environment Partition? In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 24529–24542. Curran Associates, Inc.

Lippe, P.; Magliacane, S.; Löwe, S.; Asano, Y. M.; Cohen, T.; and Gavves, S. 2022. CITRIS: Causal Identifiability from Temporal Intervened Sequences. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvari, C.; Niu, G.; and Sabato, S., eds., *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 13557–13603. PMLR.

Liu, J.; Hu, Z.; Cui, P.; Li, B.; and Shen, Z. 2021. Heterogeneous Risk Minimization. In Meila, M.; and Zhang, T., eds., *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, 6804–6814. PMLR.

Lu, W.; Wang, J.; Li, H.; Chen, Y.; and Xie, X. 2022. Domain-invariant Feature Exploration for Domain Generalization. *Transactions on Machine Learning Research*.

Lu, W.; Wang, J.; Sun, X.; Chen, Y.; and Xie, X. 2023. Out-of-distribution Representation Learning for Time Series Classification. In *The Eleventh International Conference on Learning Representations*.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; and Chintala, S. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Qian, H.; Pan, S. J.; and Miao, C. 2021. Latent Independent Excitation for Generalizable Sensor-based Cross-Person Activity Recognition. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(13): 11921–11929.

Ragab, M.; Chen, Z.; Zhang, W.; Eldele, E.; Wu, M.; Kwoh, C.-K.; and Li, X. 2022. Conditional Contrastive Domain Generalization for Fault Diagnosis. *IEEE Transactions on Instrumentation and Measurement*, 71: 1–12.

Reiss, A. 2012. PAMAP2 Physical Activity Monitoring. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5NW2H.

Sagawa, S.; Koh, P. W.; Hashimoto, T. B.; and Liang, P. 2020. Distributionally Robust Neural Networks. In *International Conference on Learning Representations*.

Schmidt, P.; Reiss, A.; Duerichen, R.; Marberger, C.; and Van Laerhoven, K. 2018. Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, ICMI '18, 400–408. New York,

NY, USA: Association for Computing Machinery. ISBN 9781450356923.

Schölkopf, B.; Locatello, F.; Bauer, S.; Ke, N. R.; Kalchbrenner, N.; Goyal, A.; and Bengio, Y. 2021. Toward Causal Representation Learning. *Proceedings of the IEEE*, 109(5): 612–634.

Shi, R.; Huang, H.; Yin, K.; Zhou, W.; and Jin, H. 2024. Orthogonality Matters: Invariant Time Series Representation for Out-of-distribution Classification. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, 2674–2685. New York, NY, USA: Association for Computing Machinery. ISBN 9798400704901.

Song, X.; Yao, W.; Fan, Y.; Dong, X.; Chen, G.; Niebles, J. C.; Xing, E.; and Zhang, K. 2023. Temporally Disentangled Representation Learning under Unknown Nonstationarity. In Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; and Levine, S., eds., *Advances in Neural Information Processing Systems*, volume 36, 8092–8113. Curran Associates, Inc.

Tobin, J.; Fong, R.; Ray, A.; Schneider, J.; Zaremba, W.; and Abbeel, P. 2017. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 23–30.

Tomašev, N.; Glorot, X.; Rae, J. W.; Zielinski, M.; Askham, H.; Saraiva, A.; Mottram, A.; Meyer, C.; Ravuri, S.; Protsyuk, I.; et al. 2019. A clinically applicable approach to continuous prediction of future acute kidney injury. *Nature*, 572(7767): 116–119.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).

Vowels, M. J.; Camgoz, N. C.; and Bowden, R. 2022. D'ya Like DAGs? A Survey on Structure Learning and Causal Discovery. *ACM Comput. Surv.*, 55(4).

Wang, J.; Lan, C.; Liu, C.; Ouyang, Y.; Qin, T.; Lu, W.; Chen, Y.; Zeng, W.; and Yu, P. S. 2023. Generalizing to Unseen Domains: A Survey on Domain Generalization. *IEEE Transactions on Knowledge and Data Engineering*, 35(8): 8052–8072.

Wu, H.; Zhou, H.; Long, M.; and Wang, J. 2023. Interpretable weather forecasting for worldwide stations with a unified deep model. *Nature Machine Intelligence*, 5(6): 602–611.

Wu, X.; Teng, F.; Li, X.; Zhang, J.; Li, T.; and Duan, Q. 2025. Out-of-Distribution Generalization in Time Series: A Survey. *arXiv preprint arXiv:2503.13868*.

Wu, Z.; Huang, N. E.; Long, S. R.; and Peng, C.-K. 2007. On the trend, detrending, and variability of nonlinear and nonstationary time series. *Proceedings of the National Academy of Sciences*, 104(38): 14889–14894.

Yao, W.; Chen, G.; and Zhang, K. 2022. Temporally Disentangled Representation Learning. In Koyejo, S.; Mohamed, S.; Agarwal, A.; Belgrave, D.; Cho, K.; and Oh, A., eds., *Advances in Neural Information Processing Systems*, volume 35, 26492–26503. Curran Associates, Inc.

Yao, W.; Sun, Y.; Ho, A.; Sun, C.; and Zhang, K. 2022. Learning Temporally Causal Latent Processes from General Temporal Data. In *International Conference on Learning Representations*.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2018. mixup: Beyond Empirical Risk Minimization. In *International Conference on Learning Representations*.

Zhang, M.; and Sawchuk, A. A. 2012. USC-HAD: a daily activity dataset for ubiquitous activity recognition using wearable sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, 1036–1043. New York, NY, USA: Association for Computing Machinery. ISBN 9781450312240.

Zhou, K.; Liu, Z.; Qiao, Y.; Xiang, T.; and Loy, C. C. 2023. Domain Generalization: A Survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4): 4396–4415.