Disentangling Long-Short Term State Under Unknown Interventions for Online Time Series Forecasting

Ruichu Cai[1, 2], Haiqin Huang[1], Zhifan Jiang[1], Zijian Li[3*], Changze Zhou[1], Yuequn Liu[1], Yuming Liu[1], Zhifeng Hao[4]

[1]School of Computer Science, Guangdong University of Technology, China
[2]Peng Cheng Laboratory, Shenzhen, China
[3]Machine Learning Department, Mohamed bin Zayed University of Artificial Intelligence, United Arab Emirates
[4]Shantou University

Abstract

Current methods for time series forecasting struggle in the online scenario, since it is difficult to preserve long-term dependency while adapting short-term changes when data are arriving sequentially. Although some recent methods solve this problem by controlling the updates of latent states, they cannot disentangle the long/short-term states, leading to the inability to effectively adapt to nonstationary. To tackle this challenge, we propose a general framework to disentangle long/short-term states for online time series forecasting. Our idea is inspired by the observations where short-term changes can be led by unknown interventions like abrupt policies in the stock market. Based on this insight, we formalize a data generation process with unknown interventions on short-term states. Under mild assumptions, we further leverage the independence of short-term states led by unknown interventions to establish the identification theory to achieve the disentanglement of long/short-term states. Built on this theory, we develop a **L**ong **S**hort-**T**erm **D**isentanglement model (**LSTD**) to extract the long/short-term states with long/short term encoders, respectively. Furthermore, the **LSTD** model incorporates a smooth constraint to preserve the long-term dependencies and an interrupted dependency constraint to enforce the forgetting of short-term dependencies, together boosting the disentanglement of long/short-term states. Experimental results on several benchmark datasets show that our **LSTD** model outperforms existing methods for online time series forecasting, validating its efficacy in real-world applications.

Code — https://github.com/DMIRLAB-Group/LSTD

Introduction

As one of the most fundamental tasks in time series analysis (Hamilton 2020; Liu et al. 2023), time series forecasting (Zhou et al. 2021; Zeng et al. 2023; Kitaev, Kaiser, and Levskaya 2020; Liu et al. 2021; Wu et al. 2021; Zhou et al. 2021) plays a critical role in various fields such as finance (Clements, Franses, and Swanson 2004; Cao, Li, and Li 2019), and traffic (Lippi, Bertini, and Frasconi 2013). However, in the industry, since time series data often arrives sequentially and is accompanied by temporal distribution shifts (Wang et al. 2022; Li et al. 2024b), existing methods (Wu et al. 2021; Nie et al. 2022; Lopez-Paz and Ranzato 2017) that heavily rely on the mini-batch training paradigm can hardly adapt to these changing distributions, leading to suboptimal prediction results in the online scenario.

To solve this problem, several recent methodologies (Cai et al. 2023; Guo et al. 2024; Mejri, Amarnath, and Chatterjee 2024; Lin 1992) are proposed to adapt the short-term nonstationarity and long-term dependencies. FSNet (Pham et al. 2022) leverages the partial derivative to characterize the short-term information and an associative memory to preserve the long-term dependencies. To better combine long-term and short-term historical information, OneNet (Wen et al. 2024) uses a reinforcement learning-based to dynamically adjust the combination of temporal correlation and cross-variable dependency models. Recently, Zhang et al. (Zhang et al. 2024) propose the Concept Drift Detection and Adaptation framework (D^3A), which first detects the temporal distribution shift and then employs an aggressive manner to update the model. In summary, these methods aim to address online time series forecasting via two steps: 1) disentangling long/short-term states; and 2) adapting short-term states and reusing long-term states for forecasting. Please refer to Appendix A for further discussion about online time series forecasting and causal representation learning.

Although current methods achieve non-trivial contributions on how to update short-term states or how to efficiently combine the long/short-term states, they implicitly assume that the long/short-term states have been well-disentangled from nonstationary time series data. However, this assumption is hard to meet, and without disentanglement of the long/short-term states, existing methods can hardly adapt to the nonstationary environments. Figure 1 provides a finance example, where the monetary exchange rate is influenced by the short-term variables (e.g. customs duties) and the long-term variables (e.g., financial revenue). Nonstationarity occurs due to unknown customs tariff policies. As shown in Figure 1 (a), when the long-term and short-term latent variables are not disentangled, the financial revenues are entangled with the customs duties. As a result, existing methods can be hard to effectively adapt to the changes in financial revenues and may obtain an inaccurate forecasting performance even if they use a masterly strategy to update the

*corresponding author: Zijian Li (leizigin@gmail.com)

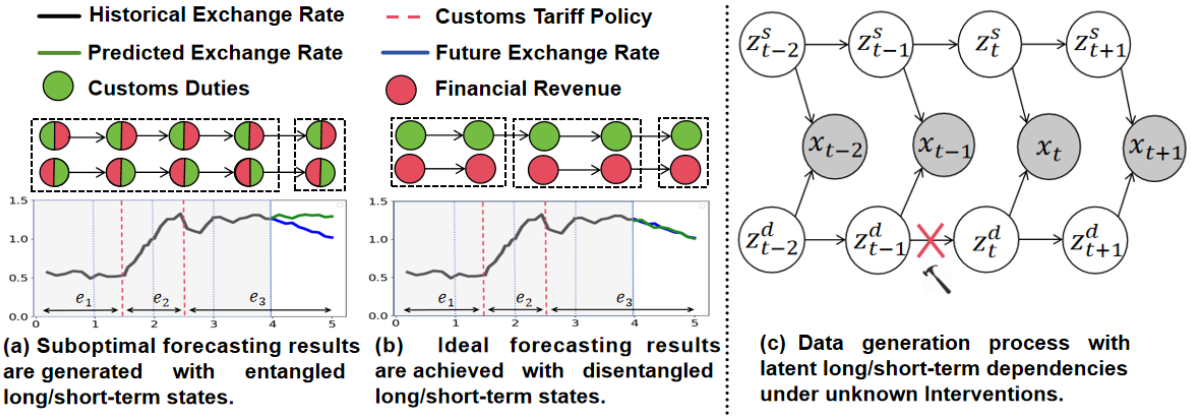

Figure 1: Illustration of sequentially arriving exchange rate data, which is influenced by short-term customs duties and long-term financial revenue. Moreover, the short-term customs duties are intervened by sudden customs tariff policies. (a) If the estimated short-term customs duties and long-term financial revenue are entangled, short-term influence from Environments (e.g., e_1, e_2, e_3) may affect the effectiveness of the models to adapt to the changing environments, leading to suboptimal forecasting performance. (b) When the long/short-term states are disentangled, the model can quickly adapt to environmental changes and hence achieve correct forecasting results. (c) Data generation process for time-series data. $\boldsymbol{z}_t^s$ and $\boldsymbol{z}_t^d$ denotes the long/short-term states. Note that the short-term states $\boldsymbol{z}_t^d$ are intervened randomly.

short-term states and preserve the long-term states.

Based on the aforementioned example, we observe that nonstationarity is brought by the unknown interventions on short-term states. Moreover, to address the online forecasting task, it is intuitive to find that we should disentangle the long/short-term states from the time series with unknown interventions as shown in Figure 1 (b). Under this intuition, we first consider that the sequentially arriving data follow a data generation process in Figure 1 (c), where the latent short-term states are influenced by unknown interventions. Under mild assumptions, we establish disentanglement results on long/short-term states by leveraging the independence of intervened short-term states. To bridge the gap between theory and practice, we further develop a **L**ong **S**hort-**T**erm **D**isentanglement model (**LSTD**) to solve the online time series forecasting problem. Specifically, the proposed **LSTD** model includes a minimal update constraint to preserve the long-term dependencies and an interrupted dependency constraint to enforce the forgetting of short-term dependencies, which facilitates the disentanglement of long-term and short-term latent states. Empirical results on several real-world benchmark datasets show that the proposed **LSTD** method outperforms existing state-of-the-art methods for online time series forecasting, highlighting its effectiveness in real-world applications.

Data Generation Process for Time Series Data

To show how to disentangle the long-term and short-term latent states in the online time series forecasting scenario, we first introduce the data generation process of time series data as shown in Figure 1 (c). Mathematically, we let $\mathbf{x} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_t, \cdots\}$ be time series data with discrete time steps, in which each observation $\mathbf{x}_t$ is generated from latent variables $\mathbf{z}_t$ through an invertible and nonlinear mixing function g as formalized in Equation (1).

$$\mathbf{x}_t = g(\mathbf{z}_t). \tag{1}$$

At each time step t, $\mathbf{z}_t \in \mathbb{R}^n$ are divided into the long-term latent states $\mathbf{z}_t^s \in \mathbb{R}^{n_s}$ and short-term latent states $\mathbf{z}_t^d \in \mathbb{R}^{n_d}$, and $n = n_s + n_d$. Moreover, the i-th component of $\mathbf{z}_t^s$ is generated by some components of historical long-term latent states $\mathbf{z}_{t-\tau}^s$ with the time lag of τ via a nonparametric function as shown in Equation (2).

$$\begin{aligned} z_{t,i}^s = f_i^s(\{z_{t-\tau,k}^s | z_{t-\tau,k}^s \in \mathbf{Pa}(z_{t,i}^s)\}, \varepsilon_{t,i}^s), \\ \text{with} \quad \varepsilon_{t,i}^s \sim p_{\varepsilon_{t,i}^s}, \end{aligned} \tag{2}$$

where $\mathbf{Pa}(z_{t,i}^s)$ denotes the set of latent variables that directly cause $z_{t,i}^s$ and $\varepsilon_{t,i}^s$ denotes the temporally and spatially independent noise extracted from a distribution $p_{\varepsilon_{t,i}^s}$.

Moreover, according to the observation in the example in Figure 1, we assume that the nonstationarity in time series data is led by the interventions on the short-term latent variables (e.g., the truncation between $\mathbf{z}_{t-1}^d$ and $\mathbf{z}_t^d$ in Figure 1 (c)). It is noted that when the interventions occur is unknown. To illustrate the randomness of the interventions, we let I be an indicator to decide if an intervention occurs and I comes from a Bernoulli distribution $\mathbf{B}(I, \theta)$ with the probability of θ. When $I = 0$, it indicates no intervention, whereas when $I = 1$, it signifies intervention. When intervention occurs, the data is generated solely by noise. Formally, the generation process of the short-term latent variables is shown as follows:

$$\begin{aligned} z_{t,j}^d = \begin{cases} f_j^d(\{z_{t-\tau,k}^d | z_{t-\tau,k}^d \in \mathbf{Pa}(z_{t,j}^d)\}, \varepsilon_{t,j}^d) & \text{, if I = 0} \\ f_j^d(\varepsilon_{t,j}^d) & \text{, if I = 1} \end{cases} \\ \text{where} \quad \varepsilon_{t,j}^d \sim p_{\varepsilon_{t,j}^d} \quad \text{and} \quad I \sim \mathbf{B}(I, \theta), \end{aligned} \tag{3}$$

where $\mathbf{Pa}(z_{t,j}^d)$ denotes the set of latent variables that directly cause $z_{t,j}^d$ and $\varepsilon_{t,j}^d$ denotes the temporally and spatially independent noise extracted from a distribution $p_{\varepsilon_{t,j}^d}$.

The data generation process as shown in Equation (1)-(3) can be well interpreted by the aforementioned financial example. First, the exchange rate can be considered as the observation time series data. Sequentially, the financial revenue and the customs duties denote the long-term and short-term latent variables, respectively. Finally, $I = 1$ denotes that the customs tariff policy intervenes with customs duties and leads to temporal distribution changes. In the context of online time series forecasting, where the time series data arrive sequentially, we first predict the value of $\mathbf{x}_{L+1:H}$ given $\mathbf{x}_{1:L}$ at t-th time step. Then at $t+1$-th time step, we have access to the true value of $\mathbf{x}_{L+1:H}$ to update the model and then use $\mathbf{x}_{2:L+1}$ to predict the value of $\mathbf{x}_{L+2:H+1}$.

Disentanglement of Long-Term and Short-Term States

To disentangle the long-term latent variables $\mathbf{z}_t^s$ and the short-term latent variables $\mathbf{z}_t^d$, we propose the block-wise identification theory in Theory 1. Mathematically, the block-wise identification means that for the ground-truth $\mathbf{z}_t^*$, there exists $\hat{\mathbf{z}}_t^*$ and an invertible function $h_z^* : \mathbb{R}^{n^*} \to \mathbb{R}^{n^*}$, such that $\hat{\mathbf{z}}_t^* = h_z^*(\mathbf{z}_t^*)$. And $*$ can be d or s.

Theorem 1. ***(Subspace Identification of the long-term and short-term Latent Variables)*** *Suppose that the observed data from long/short-term is generated following the data generation process in Figure1 (c), and we further make the following assumptions:*

- *A1 (**Smooth, Positive and Conditional independent Density:**) (Yao, Chen, and Zhang 2022; Yao et al. 2021) The probability density function of latent variables is smooth and positive, i.e., $p(\mathbf{z}_{t-\tau+1:t}|\mathbf{z}_{t-\tau}) > 0$ over $\boldsymbol{Z}_{t-\tau}$ and $\boldsymbol{Z}_{t-\tau+1:t}$. Conditioned on $\mathbf{z}_{t-\tau}$ each z_i is independent of any other z_j for $i, j \in 1, ..., n, i \neq j$, i.e, $\log p(\mathbf{z}_{t-\tau+1:t}|\mathbf{z}_{t-\tau}) = \sum_{k=1}^{n_s} \log p(z_{t-\tau+1:t,k}|\mathbf{z}_{t-\tau})$*
- *A2 (**non-singular Jacobian**): (Kong et al. 2023b) Each generating function g has non-singular Jacobian matrices almost anywhere and g is invertible.*
- *A3 (**Linear Independence**:) (Yao, Chen, and Zhang 2022) For any $\mathbf{z}^d \in \boldsymbol{Z}_{t-\tau+1:t}^d \subseteq R^{n_d}, \bar{v}_{t-\tau,1}, ..., \bar{v}_{t-\tau,n_d}$ as n_d vector functions in $z_{t-\tau,1}^d, ..., z_{t-\tau.l}^d, ..., z_{t-\tau,n_d}^d$ are linear independent, where $\bar{v}_{t-\tau,l}$ are formalized as follows:*

$$\bar{\boldsymbol{v}}_{t-\tau,l} = \frac{\partial^2 \log p(\mathbf{z}_{t-\tau+1:t}^d|\mathbf{z}_{t-\tau}^d)}{\partial z_{t-\tau+1:t,k}^d \partial z_{t-\tau,l}^d} \tag{4}$$

Suppose that we learn $(\hat{g}, \hat{f}_i^s, \hat{f}_i^d)$ to achieve Equation (1)-(3) with the minimal number of transition edge among short term latent variables $\mathbf{z}_1^d, \cdots, \mathbf{z}_t^d, \cdots$, then the long-term and short-term latent variables are block-wise identifiable.

Proof Sketch: The proof can be found in Appendix **B**. First, we construct an invertible transformation h_z between the ground-truth latent variables and estimated ones. Sequentially, we prove that the ground truth of long-term latent variables is not the function of short-term latent variables by leveraging the pairing time series from different influences. Sequentially, we leverage sufficient variability of historical information to show that the short-term latent variables are not the function of the estimated long-term latent variables. Moreover, by leveraging the invertibility of transformation h_z, we can obtain the Jacobian of h_z as shown in Equation (5), where $B = 0$ and $C = 0$, since the ground truth long-term latent variables are not the functions of short-term latent variables and the short-term latent variables are not the function of the estimated long-term latent variables.

$$\mathbf{J}_{h_z} = \left[\begin{array}{c|c} \mathbf{A} := \frac{\partial \mathbf{z}_t^s}{\partial \hat{\mathbf{z}}_t^s} & \mathbf{B} := \frac{\partial \mathbf{z}_t^s}{\partial \hat{\mathbf{z}}_t^d} = 0 \\ \hline \mathbf{C} := \frac{\partial \mathbf{z}_t^d}{\partial \hat{\mathbf{z}}_t^s} = 0 & \mathbf{D} := \frac{\partial \mathbf{z}_t^d}{\partial \hat{\mathbf{z}}_t^d} \end{array}\right] \tag{5}$$

Discussion of the Identification Results: We would like to highlight that the theoretical results provide sufficient conditions for the identification of our model. That implies: 1) our model can be correctly identified when all the assumptions hold. 2) at the same time, even if some of the above assumptions do not hold, our method may still learn the correct model. From an application perspective, these assumptions rigorously defined a subset of applicable scenarios of our model. Thus, we provide detailed explanations of the assumptions, how they relate to real-world scenarios, and in which scenarios they are satisfied.

Smooth, Positive and Conditional independent Density. This assumption is common in the existing identification results (Yao, Chen, and Zhang 2022; Yao et al. 2021; Yao, Chen, and Zhang 2022; Yao et al. 2021). In real-world scenarios, smooth and positive density implies continuous changes in historical information, such as temperature variations in weather data. To achieve this, we should sample as much data as possible to learn the transition probabilities more accurately. Moreover, The conditional independent assumption is also common in identifying temporal latent processes (Li et al. 2024a). Intuitively, it means there are no immediate relations among latent variables. To satisfy this assumption, we can sample data at high frequency to avoid instantaneous dependencies caused by subsampling.

Non-singular Jacobian of g. This assumption is also common in (Kong et al. 2023b; Li et al. 2024b,c; Xie et al. 2023; Kong et al. 2023a). Mathematically, it denotes that the Jacobian from the latent variables to the observed variables is full rank. In real-world scenarios, it means that there is at least one observation for each latent variable. To meet this assumption, we can ignore such independent latent variables since they have no influence on the observations.

Linear Independence. The second assumption is also common in (Yao, Chen, and Zhang 2022), meaning that the influence from each latent source to observation is independent. The linear independence assumption is standard in the identification of nonlinear ICA (Allman, Matias, and Rhodes 2009; Hyvarinen and Morioka 2016; Yan et al. 2024; Huang et al. 2023; Hälvä and Hyvarinen 2020; Lippe et al. 2022). It implies that linear independence is necessary for a unique

solution to a system of equations. Though this assumption is untestable, we may investigate whether it is satisfied through the prior knowledge of the applications.

Data generation process with unknown interventions. In real-world time series data, there are many unknown interventions that lead to nonstationarity like the financial example in Figure 1. Therefore, this assumption is reasonable. Besides, we need to impose discontinuities in the short-term components to break the symmetry between the long and short terms in the causal graph. This ensures that the long and short terms are identifiable. Through the identifiable theory, we can explain whether the module learns long-term or short-term components, thereby theoretically guaranteeing the disentanglement of long and short terms. In practice, we may investigate the nonstationarity of the data to test whether this assumption is valid.

Long Short-Term Disentanglement Model

Model Overview

In this section, we introduce the implementation of the long/short-term disentanglement model as shown in Figure 2. Specifically, it uses a variational sequential autoencoder as a backbone architecture and further employs long-term and short-term prior architectures with smooth constraint and sparse dependency constraint for long-term and short-term latent variable disentanglement.

Variational Sequential Autoencoder

To model the time series data, we follow the data generation process in Figure 1 (c) and derive the evidence lower bound (ELBO) as shown in Equation (6).

$$\begin{aligned} ELBO = & \underbrace{\mathbb{E}_{q(\mathbf{z}^s_{1:H}|\mathbf{x}_{1:H})}\mathbb{E}_{q(\mathbf{z}^d_{1:H}|\mathbf{x}_{1:H})} \ln p(\mathbf{x}_{1:H}|\mathbf{z}^s_{1:H}, \mathbf{z}^d_{1:H})}_{L_R + L_P} \\ & \underbrace{- D_{KL}(q(\mathbf{z}^s_{1:H}|\mathbf{x}_{1:H})||p(\mathbf{z}^s_{1:H}))}_{L^s_K} \\ & \underbrace{-D_{KL}(q(\mathbf{z}^d_{1:H}|\mathbf{x}_{1:H})||p(\mathbf{z}^d_{1:H}))}_{L^d_K} \end{aligned} \tag{6}$$

where L_R and L_P denote the reconstructed and prediction loss, respectively:

$$\begin{aligned} L_R &= \frac{1}{L}\sum_{i=1}^{L}(\hat{\mathbf{z}}_i - \mathbf{z}_i)^2 \\ L_P &= \frac{1}{H-L}\sum_{i=L+1}^{H}(\hat{\mathbf{z}}_i - \mathbf{z}_i)^2 \end{aligned} \tag{7}$$

D_{KL} denotes the KL divergence. Specifically, $q(\mathbf{z}^s_{1:H}|\mathbf{x}_{1:H})$, $q(\mathbf{z}^d_{1:H}|\mathbf{x}_{1:H})$, which includes the encoder and the latent transition module in Figure 2, is used to approximate the prior distribution. $p(\mathbf{x}_{1:H}|\mathbf{z}^s_{1:H}, \mathbf{z}^d_{1:H})$ is used to reconstruct the historical observations and forecast the future values. The aforementioned two distributions can be formalized as follows:

$$\hat{\mathbf{z}}^s_{1:H}, \hat{\mathbf{z}}^d_{1:H} = \phi(\mathbf{x}_{1:H}), \quad \hat{\mathbf{x}}_{1:H} = \psi(\mathbf{z}_{1:H}), \tag{8}$$

Where $\mathbf{z}_{1:H}$ denotes the combination of $\hat{\mathbf{z}}^s_{1:H}$ and $\hat{\mathbf{z}}^d_{1:H}$. For the implementation of ϕ, we follow the backbone of FSNet (Pham et al. 2022). For the implementation of ψ, we employ an MLP (Multilayer Perceptron). Please refer to Appendix C for more implementation details of the **LSTD** model.

Long-Term and Short-Term Prior Networks

To model the prior distribution of the long-term latent variables, we propose the long-term prior networks. Similar to the existing methods for causal representation learning (Yao et al. 2021; Yao, Chen, and Zhang 2022), we let $\{r^s_i\}$ be a set of learned inverse transition functions that take the estimated long-term latent variables and output the noise term, i.e., $\hat{\epsilon}^s_{t,i} = r^s_i(\hat{z}^s_{t,i}, \hat{\mathbf{z}}^s_{t-1})$ [1] and each r^s_i is modeled with MLPs. Then we devise a transformation $\kappa^s := \{\hat{\mathbf{z}}^s_{t-1}, \hat{\mathbf{z}}^s_t\} \rightarrow \{\hat{\mathbf{z}}^s_{t-1}, \hat{\epsilon}^s_t\}$, and its Jacobian is $\mathbf{J}_{\kappa^s} = \begin{pmatrix} \mathbb{I} & 0 \\ M & \text{diag}\left(\frac{\partial r^s_i}{\partial \hat{z}^s_{t,i}}\right) \end{pmatrix}$, where M denotes a matrix. By applying the change of variables formula, we have the following equation:

$$\log p(\hat{\mathbf{z}}^s_{t-1}, \hat{\mathbf{z}}^s_t) = \log p(\hat{\mathbf{z}}^s_{t-1}, \hat{\epsilon}^s_t) + \log|\det(\mathbf{J}_{\kappa^s})|. \tag{9}$$

Since we assume that the noise term in Equation (9) is independent with $\mathbf{z}^s_{t-1}$, we can enforce the independence of the estimated noise $\hat{\epsilon}^s_t$ and further have:

$$\log p(\hat{\mathbf{z}}^s_t|\hat{\mathbf{z}}^s_{t-1}) = \log p(\hat{\epsilon}^s_t) + \sum_{i=1}^{n_s}\log|\frac{\partial r^s_i}{\partial \hat{z}^s_{t,i}}|. \tag{10}$$

Therefore, the long-term prior can be estimated as follows:

$$\log p(\hat{\mathbf{z}}^s_{1:t}) = \log p(\hat{\mathbf{z}}^s_1) + \sum_{\tau=2}^{t}\left(\sum_{i=1}^{n_s}\log p(\hat{\epsilon}^s_{\tau,i}) + \sum_{i=1}^{n_s}\log|\frac{\partial r^s_i}{\partial \hat{z}^s_{\tau,i}}|\right), \tag{11}$$

where $p(\hat{\epsilon}^s_i)$ follow Gaussian distributions. Similarly, we can further estimate the short-term prior as follows:

$$\log p(\hat{\mathbf{z}}^d_{1:t}) = \log p(\hat{\mathbf{z}}^d_1) + \sum_{\tau=2}^{t}\left(\sum_{i=1}^{n_d}\log p(\hat{\epsilon}^d_{\tau,i}) + \sum_{i=1}^{n_d}\log|\frac{\partial r^d_i}{\partial \hat{z}^d_{\tau,i}}|\right), \tag{12}$$

Smooth Constraint for Long-Term Disentanglement

To preserve the long-term dependencies in the long-term latent variables, we propose the smooth constraint. Since the causal relationships of the long-term dependencies are stable, the association of the long-term dependencies is also stable. Based on this insight, we consider the attention weights as associations and extract the association with the help of the self-attention mechanism. Specifically, we first split the $\mathbf{z}^s_{1:H}$ into two equal-size segmentation $\mathbf{z}^s_{1:H/2}$ and $\mathbf{z}^s_{H/2:H}$. And then the association of $\mathbf{z}^s_{1:H/2}$ and $\mathbf{z}^s_{H/2:H}$ can be formalized as follows:

$$\begin{aligned} A_{\mathbf{z}^s_h} &= \text{Softmax}(\frac{\mathbf{z}^s_{1:H/2}{\mathbf{z}^s_{1:H/2}}^\top}{\sqrt{n_s}}), \\ A_{\mathbf{z}^s_e} &= \text{Softmax}(\frac{\mathbf{z}^s_{H/2:H}{\mathbf{z}^s_{H/2:H}}^\top}{\sqrt{n_s}}), \end{aligned} \tag{13}$$

[1] We use the superscript symbol to denote estimated variables.

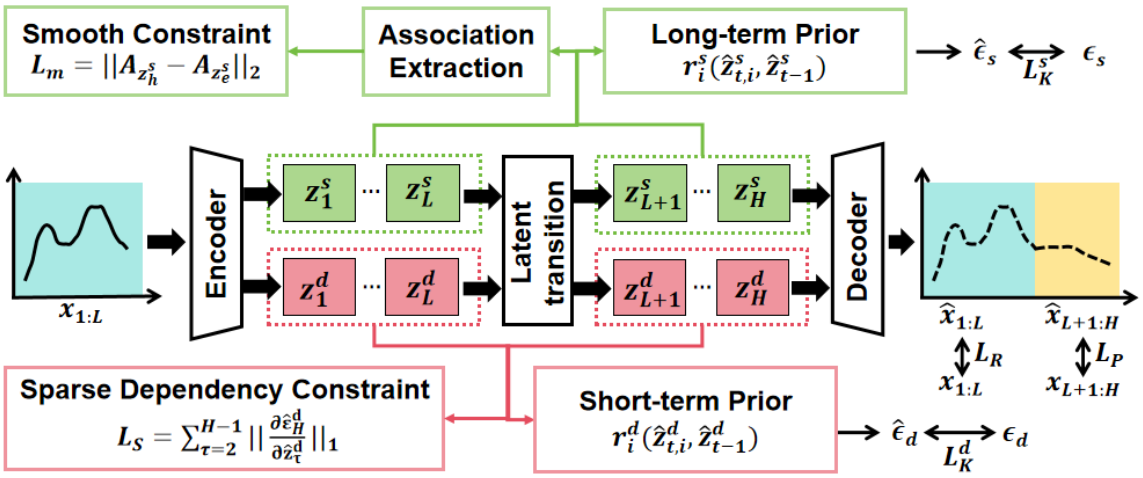

Figure 2: The framework of the proposed **LSTD** model. The long/short-term latent variables $\mathbf{z}_{1:L}^d$ and $\mathbf{z}_{1:L}^s$ are extracted from the encoder. And the latent transition module is used to estimated the $\mathbf{z}_{L+1:H}^d$ and the $\mathbf{z}_{L+1:H}^s$ from $\mathbf{z}_{1:L}^d$ and $\mathbf{z}_{1:L}^s$, respectively. The long-term and short-term prior networks are used to estimate the prior distributions.

in which $A_{\mathbf{z}_h^s}$ and $A_{\mathbf{z}_e^s}$ denote the association matrices of the start half and the end half segments. Hence, we can restrict the long-term dependencies by restricting the similarity of these two matrices as shown in Equation (14)

$$\mathcal{L}_m = ||A_{\mathbf{z}_h^s} - A_{\mathbf{z}_e^s}||_2, \tag{14}$$

where $||\cdot||_2$ denotes the L2 norm of matrices.

Interrupted Dependency Constraint for Short-Term Disentanglement

Since the nonstationarity is assumed to be led by the interventions to the short-term latent variables, given $\mathbf{z}_{1:H}^d$, if intervention occurs at τ-th time step, and $2 < \tau < H-1$, then $\frac{\partial \varepsilon_{H,i}^d}{\partial z_{\tau-1,j}^d} = 0$, where $i, j \in \{1, \cdots, n_d\}$. Based on this intuition, we aim to enforce the interruption of the estimated short-term dependencies to meet the unknown interventions. To achieve this, we propose the interrupted dependency constraint for the short-term variables. Specifically, given the estimated short-term variables $\mathbf{z}_{1:H}^d$, we have:

$$\mathcal{L}_s = \sum_{(i,j)\in\{1,\cdots,n_d\}} \sum_{\tau\in\{2,\cdots,H-1\}} ||\frac{\partial \hat{\varepsilon}_{H,i}^d}{\partial \hat{z}_{\tau-1,j}^d}||_1, \tag{15}$$

where $||\cdot||_1$ denote the L1 norm.

By using the aforementioned interrupted dependency constraint, the intervention on the short-term latent variables can be automatically detected, which finally enforces the disentanglement of the short-term latent variables.

Model Summary

By combining the aforementioned variational sequentially autoencoder with the restriction of smooth constraint and interrupted dependency constraint, we can finally formalize the total loss of the proposed **LSTD** model as follows:

$$\mathcal{L} = L_R + L_P + \beta L_K + \alpha L_m + \gamma L_s, \tag{16}$$

where $L_K = L_K^s + L_K^d$. And α, β, γ are hyper-parameters.

Experiment

Datasets

To evaluate the performance of our method, we consider the following datasets. **ETT** is an electricity transformer temperature dataset collected from two separate counties in China, which contains two separate datasets {ETTh2, ETTm1} for one hour level and minutes level, respectively. **Exchange** is the daily exchange rate dataset from eight foreign countries including Australia, British, Canada, Switzerland, China, Japan, New Zealand, and Singapore ranging from 1990 to 2016. **Weather** [2] is recorded at the Weather Station at the Max Planck Institute for Biogeochemistry in Jena, Germany. **ECL** [3] is an electricity-consuming load dataset with the electricity consumption (kWh) collected from 321 clients. **Traffic** [4] is a dataset of traffic speeds collected from the California Transportation Agencies (CalTrans) Performance Measurement System (PeMS). For each dataset, we follow the standard preprocessing and setting in OneNet (Wen et al. 2024).

Baselines

We consider nine state-of-the-art as follows: OneNet (Wen et al. 2024) which considered the temporal and feature relationships and used reinforcement learning to update their relationships in real-time. At the same time, we compared with a very excellent backbone model FSNet (Pham et al. 2022) which considered gradient updates to optimize fast new as well as retained information and be used in OneNet. Besides, we also compared the OneNet model with TCN as its backbone named OnetNet-TCN, and the regular usage of TCN named Online-TCN (Zinkevich 2003) for online learning. The Experience Replay (ER) (Chaudhry et al.

[2] https://www.bgc-jena.mpg.de/wetter/

[3] https://archive.ics.uci.edu/dataset/321/electricityloaddiagrams20112014

[4] https://pems.dot.ca.gov/

Models	Len	LSTD	OneNet	FSNet	OneNet-T	DER++	ER	MIR	TFCL	Online-T	Informer
ETTh2	1	**0.377**	0.380	0.466	0.411	0.508	0.508	0.486	0.557	0.502	7.571
	24	0.543	**0.532**	0.687	0.772	0.828	0.808	0.812	0.846	0.830	4.629
	48	0.616	**0.609**	0.846	0.806	1.157	1.136	1.103	1.208	1.183	5.692
ETTm1	1	**0.081**	0.082	0.085	0.082	0.083	0.086	0.085	0.087	0.214	0.456
	24	0.102	**0.098**	0.115	0.212	0.196	0.202	0.192	0.211	0.258	0.478
	48	0.115	**0.108**	0.127	0.223	0.208	0.220	0.210	0.236	0.283	0.388
WTH	1	**0.153**	0.156	0.162	0.171	0.174	0.180	0.179	0.177	0.206	0.426
	24	**0.136**	0.175	0.188	0.293	0.287	0.293	0.291	0.301	0.308	0.380
	48	**0.157**	0.200	0.223	0.310	0.294	0.297	0.297	0.323	0.302	0.367
ECL	1	**2.112**	2.351	3.143	2.470	2.657	2.579	2.575	2.732	3.309	3.813
	24	**1.422**	2.074	6.051	4.713	8.996	9.327	9.265	12.094	11.339	9.185
	48	**1.411**	2.201	7.034	4.567	9.009	9.685	9.411	12.110	11.534	11.183
Traffic	1	**0.231**	0.241	0.312	0.236	0.271	0.284	0.298	0.306	0.334	0.234
	24	**0.398**	0.438	0.426	0.425	0.476	0.461	0.451	0.441	0.481	0.451
	48	**0.426**	0.473	0.445	0.451	0.486	0.510	0.502	0.438	0.503	0.496
Exchange	1	**0.013**	0.017	0.094	0.031	0.106	0.097	0.095	0.106	0.113	0.102
	24	**0.039**	0.047	0.113	0.060	0.111	0.162	0.104	0.098	0.116	0.107
	48	**0.043**	0.062	0.156	0.065	0.183	0.181	0.101	0.101	0.168	0.116

Table 1: Mean Square Error (MSE) results on the different datasets. TCN is abbreviated as T

2019) stored the previous data in a buffer and interleaved with newer samples during learning. Meanwhile, ER has many advanced variants: TFCL (Aljundi, Kelchtermans, and Tuytelaars 2019) used a task-boundary detection mechanism and a knowledge consolidation strategy; MIR (Aljundi et al. 2019) selected samples that cause the most forgetting; and DER++ (Buzzega et al. 2020) incorporated a knowledge distillation strategy. Additionally, we have incorporated a long-term time-series forecasting model, Informer (Zhou et al. 2021), to investigate the performance of conventional forecasting models in online time-series forecasting problems.

Quantitative Results and Discussion

Experiment results on each dataset are shown in Table 1 and Table 2. Since some methods report the best results on the original paper, we also show the best results on the aforementioned tables. Please refer to Appendix D for the experiment results with mean and variance over three random seeds. Our **LSTD** model significantly outperforms all other baselines on most online forecasting tasks. Specifically, our method outperforms the most competitive baselines by a clear margin of 44% on the Exchange, which verifies the example in the introduction. Moreover, our method also greatly reduced prediction errors in the WTH and ECL datasets. However, our method achieves the second-best but still comparable results in the ETT dataset, this might be because there are a few unknown interventions in the ETT datasets. How to address other types of nonstationarity will be an interesting future direction. In addition, we conduct performance analysis experiments and visualization in Appendix D. Compared with other models, we can find that the proposed **LSTD** has the best model performance and relatively good model efficiency.

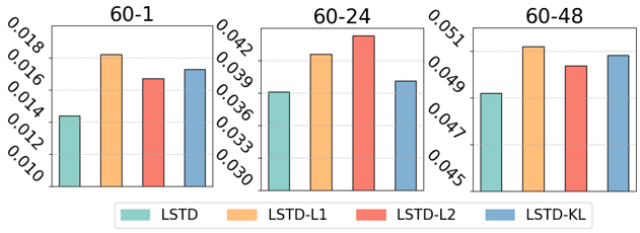

Figure 3: Ablation study on the Exchange datasets. We explore the impact of different loss terms

Qualitative Results and Discussion

We further conduct visualization results in the WTH and Exchange dataset in Figure 4. Remarkably, our method detects interventions well and achieves better visualization results than that of OneNet and FSNet, which do not explicitly disentangle the short-term and long-term variables. This is because the long/short term variables of these methods might be entangled, hindering the rapid adaptation to the changing environment of the data streams, and finally resulting in suboptimal predictions. In the meanwhile, our method disentangles the long/short term variables by sparsity dependency constraint, and can efficiently adapt to the new environment. At the same time, the smooth constraint further maintains the long-term variables behind the time series data. Therefore, the prediction curve of our method can well align with the ground truth even if the prediction length is long.

Ablation Study

We further devise three model variants. a) **LSTD**-L1: we remove the interrupted dependency constraint for short-term disentanglement. b) **LSTD**-L2: we remove the smooth constraint for long-term disentanglement. c) **LSTD**-KL: we remove the long/short-term prior and the corresponding

Models	Len	LSTD	OneNet	FSNet	OneNet-T	DER++	ER	MIR	TFCL	Online-T	Informer
ETTh2	1	**0.347**	0.348	0.368	0.374	0.375	0.376	0.410	0.472	0.436	0.850
	24	0.411	**0.407**	0.467	0.511	0.540	0.543	0.541	0.548	0.547	0.668
	48	**0.423**	0.436	0.515	0.543	0.577	0.571	0.565	0.592	0.589	0.752
ETTm1	1	**0.187**	**0.187**	0.191	0.191	0.192	0.197	0.197	0.198	0.085	0.512
	24	**0.217**	0.225	0.249	0.319	0.326	0.333	0.325	0.341	0.381	0.525
	48	0.249	**0.238**	0.263	0.371	0.340	0.351	0.342	0.363	0.403	0.460
WTH	1	**0.200**	0.201	0.216	0.221	0.235	0.244	0.244	0.240	0.276	0.458
	24	**0.223**	0.225	0.276	0.345	0.351	0.356	0.355	0.363	0.367	0.417
	48	**0.242**	0.279	0.301	0.356	0.359	0.363	0.361	0.382	0.362	0.419
ECL	1	**0.226**	0.254	0.472	0.411	0.421	0.506	0.504	0.524	0.635	0.549
	24	**0.292**	0.333	0.997	0.513	1.035	1.057	1.066	1.256	1.196	1.198
	48	**0.294**	0.348	1.061	0.534	1.048	1.074	1.079	1.303	1.235	1.164
Traffic	1	**0.225**	0.240	0.278	0.236	0.251	0.256	0.284	0.297	0.284	0.258
	24	**0.316**	0.346	0.365	0.346	0.409	0.417	0.443	0.493	0.385	0.365
	48	**0.332**	0.371	0.378	0.355	0.386	0.294	0.397	0.531	0.380	0.394
Exchange	1	**0.070**	0.085	0.174	0.117	0.173	0.124	0.118	0.153	0.169	0.115
	24	**0.132**	0.148	0.206	0.166	0.227	0.210	0.204	0.227	0.213	0.196
	48	**0.142**	0.170	0.254	0.173	0.243	0.241	0.209	0.183	0.258	0.217

Table 2: Mean Absolute Error (MAE) results on the different datasets. TCN is abbreviated as T

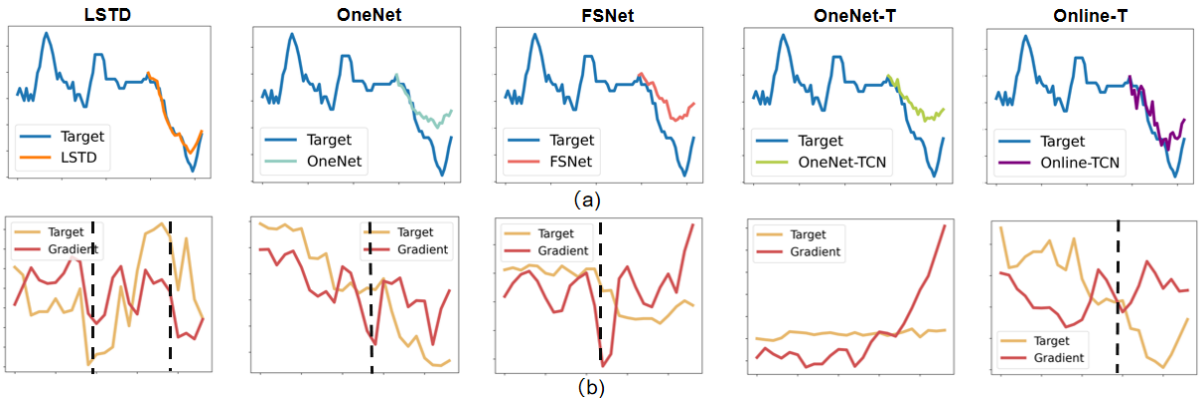

Figure 4: The figure (a) represents the visualization of the proposed LSTD and other baselines. The blue lines denote the ground-truth time series data and the lines with other colors denote the predicted results of different methods. The figure (b) shows the visualization of the LSTD method for detecting interventions. The yellow lines represent the real-time series data, and the red lines represent the gradient. Black dotted lines denote intervention occurs. *(Best view in color)*

Kullback-Leibler divergence term. Experiment results on the Exchange dataset are shown in Figure 3. We find that 1) the performance of **LSTD**-L1 drops without an accurate forgetting of the information, implying that the accurate forgetting benefits the quickly adapting to changes in the data domain and improves the disentanglement and forecasting performance. 2) the performance of **LSTD**-L2 drops without retention of the information, implying that the retention benefits the preserving of the long-term effects and improves the forecasting performance. 3) Both long-term and short-term priors play an important role in forecasting, implying that these priors can capture temporal information.

Summary

This paper presents a long/short-term state disentanglement model to address the challenges of online time-series forecasting in the presence of nonstationarity led by unknown interventions. Unlike existing methods, this model can theoretically identify both long-term and short-term latent variables, enhancing its relevance to real-world data. Technologically, the LSTD model employs the smooth constraint and sparse dependency constraint to enforce the disentanglement of long/short-term variables. In summary, this paper offers valuable insights into enhancing online time-series forecasting via causal representation learning.

Acknowledgments

This research was supported in part by National Science and Technology Major Project (2021ZD0111501), National Science Fund for Excellent Young Scholars (62122022) and Natural Science Foundation of China (U24A20233, 62206064, 62206061).

References

Aljundi, R.; Belilovsky, E.; Tuytelaars, T.; Charlin, L.; Caccia, M.; Lin, M.; and Page-Caccia, L. 2019. Online continual learning with maximal interfered retrieval. *Conference and Workshop on Neural Information Processing Systems*, 32.

Aljundi, R.; Kelchtermans, K.; and Tuytelaars, T. 2019. Task-free continual learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 11254–11263.

Allman, E. S.; Matias, C.; and Rhodes, J. A. 2009. Identifiability of parameters in latent structure models with many observed variables.

Buzzega, P.; Boschini, M.; Porrello, A.; Abati, D.; and Calderara, S. 2020. Dark experience for general continual learning: a strong, simple baseline. *Conference and Workshop on Neural Information Processing Systems*, 33: 15920–15930.

Cai, Z.; Jiang, R.; Yang, X.; Wang, Z.; Guo, D.; Kobayashi, H.; Song, X.; and Shibasaki, R. 2023. Memda: forecasting urban time series with memory-based drift adaptation. *Conference on Information and Knowledge Management, 2023*.

Cao, J.; Li, Z.; and Li, J. 2019. Financial time series forecasting model based on CEEMDAN and LSTM. *Physica A: Statistical mechanics and its applications*, 519: 127–139.

Chaudhry, A.; Rohrbach, M.; Elhoseiny, M.; Ajanthan, T.; Dokania, P. K.; Torr, P. H.; and Ranzato, M. 2019. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*.

Clements, M. P.; Franses, P. H.; and Swanson, N. R. 2004. Forecasting economic and financial time-series with nonlinear models. *Int. J. Forecast*, 20(2): 169–183.

Guo, P.; Jin, P.; Li, Z.; Bai, L.; and Zhang, Y. 2024. Online Test-Time Adaptation of Spatial-Temporal Traffic Flow Forecasting. *arXiv preprint arXiv:2401.04148*.

Hälvä, H.; and Hyvarinen, A. 2020. Hidden markov nonlinear ica: Unsupervised learning from nonstationary time series. In *Conference on UAI*, 939–948. Proceedings of Machine Learning Research.

Hamilton, J. D. 2020. *Time series analysis*. Princeton university press.

Huang, Z.; Wang, H.; Zhao, J.; and Zheng, N. 2023. Latent processes identification from multi-view time series. *International Joint Conference on Artificial Intelligence-23*.

Hyvarinen, A.; and Morioka, H. 2016. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. *Conference and Workshop on Neural Information Processing Systems*, 29.

Kitaev, N.; Kaiser, Ł.; and Levskaya, A. 2020. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*.

Kong, L.; Huang, B.; Xie, F.; Xing, E.; Chi, Y.; and Zhang, K. 2023a. Identification of nonlinear latent hierarchical models. *Conference and Workshop on Neural Information Processing Systems*, 36: 2010–2032.

Kong, L.; Ma, M. Q.; Chen, G.; Xing, E. P.; Chi, Y.; Morency, L.-P.; and Zhang, K. 2023b. Understanding masked autoencoders via hierarchical latent variable models. In *IEEE Conference on Computer Vision and Pattern Recognition*, 7918–7928.

Li, Z.; Cai, R.; Chen, G.; Sun, B.; Hao, Z.; and Zhang, K. 2024a. Subspace identification for multi-source domain adaptation. *Conference and Workshop on Neural Information Processing Systems*, 36.

Li, Z.; Cai, R.; Yang, Z.; Huang, H.; Chen, G.; Shen, Y.; Chen, Z.; Song, X.; Hao, Z.; and Zhang, K. 2024b. When and How: Learning Identifiable Latent States for Nonstationary Time Series Forecasting. *arXiv preprint arXiv:2402.12767*.

Li, Z.; Shen, Y.; Zheng, K.; Cai, R.; Song, X.; Gong, M.; Zhu, Z.; Chen, G.; and Zhang, K. 2024c. On the identification of temporally causal representation with instantaneous dependence. *arXiv preprint arXiv:2405.15325*.

Lin, L.-J. 1992. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine learning*, 8: 293–321.

Lippe, P.; Magliacane, S.; Löwe, S.; Asano, Y. M.; Cohen, T.; and Gavves, S. 2022. Citris: Causal identifiability from temporal intervened sequences. In *International Conference on Machine Learning*, 13557–13603. Proceedings of Machine Learning Research.

Lippi, M.; Bertini, M.; and Frasconi, P. 2013. Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning. *IEEE trans Intell Transp Syst*, 14(2): 871–882.

Liu, S.; Yu, H.; Liao, C.; Li, J.; Lin, W.; Liu, A. X.; and Dustdar, S. 2021. Pyraformer: Low-complexity pyramidal attention for long-range time series modeling and forecasting. In *International conference on learning representations*.

Liu, Y.; Hu, T.; Zhang, H.; Wu, H.; Wang, S.; Ma, L.; and Long, M. 2023. itransformer: Inverted transformers are effective for time series forecasting. *International Conference on Learning Representations 2024*.

Lopez-Paz, D.; and Ranzato, M. 2017. Gradient episodic memory for continual learning. *Conference and Workshop on Neural Information Processing Systems*, 30.

Mejri, M.; Amarnath, C.; and Chatterjee, A. 2024. A Novel Hyperdimensional Computing Framework for Online Time Series Forecasting on the Edge. *arXiv preprint arXiv:2402.01999*.

Nie, Y.; Nguyen, N. H.; Sinthong, P.; and Kalagnanam, J. 2022. A time series is worth 64 words: Long-term forecasting with transformers. *International Conference on Learning Representations 2023*.

Pham, Q.; Liu, C.; Sahoo, D.; and Hoi, S. C. 2022. Learning fast and slow for online time series forecasting. *International Conference on Learning Representations 2023*.

Wang, R.; Dong, Y.; Arik, S. Ö.; and Yu, R. 2022. Koopman neural forecaster for time series with temporal distribution shifts. *International Conference on Learning Representations 2023*.

Wen, Q.; Chen, W.; Sun, L.; Zhang, Z.; Wang, L.; Jin, R.; Tan, T.; et al. 2024. Onenet: Enhancing time series forecasting models under concept drift by online ensembling. *Conference and Workshop on Neural Information Processing Systems*, 36.

Wu, H.; Xu, J.; Wang, J.; and Long, M. 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Conference and Workshop on Neural Information Processing Systems*, 34: 22419–22430.

Xie, S.; Kong, L.; Gong, M.; and Zhang, K. 2023. Multi-domain image generation and translation with identifiability guarantees. In *The Eleventh International Conference on Learning Representations*.

Yan, H.; Kong, L.; Gui, L.; Chi, Y.; Xing, E.; He, Y.; and Zhang, K. 2024. Counterfactual generation with identifiability guarantees. *Conference and Workshop on Neural Information Processing Systems*, 36.

Yao, W.; Chen, G.; and Zhang, K. 2022. Temporally disentangled representation learning. *Conference and Workshop on Neural Information Processing Systems*, 35: 26492–26503.

Yao, W.; Sun, Y.; Ho, A.; Sun, C.; and Zhang, K. 2021. Learning temporally causal latent processes from general temporal data. *International Conference on Learning Representations 2022*.

Zeng, A.; Chen, M.; Zhang, L.; and Xu, Q. 2023. Are transformers effective for time series forecasting? In *Association for the Advancement of Artificial Intelligence*, volume 37, 11121–11128.

Zhang, Y.; Chen, W.; Zhu, Z.; Qin, D.; Sun, L.; Wang, X.; Wen, Q.; Zhang, Z.; Wang, L.; and Jin, R. 2024. Addressing Concept Shift in Online Time Series Forecasting: Detect-then-Adapt. *arXiv preprint arXiv:2403.14949*.

Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; and Zhang, W. 2021. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Association for the Advancement of Artificial Intelligence*, volume 35, 11106–11115.

Zinkevich, M. 2003. Online convex programming and generalized infinitesimal gradient ascent. In *International Conference on Machine Learning*, 928–936.