Realization of Causal Representation Learning to Adjust Confounding Bias in Latent Space

Jia Li[1], Xiang Li[2], Xiaowei Jia[3], Michael Steinbach[1], Vipin Kumar[1]

University of Minnesota, [1] Computer Science; [2] Bioproducts and Biosystems Engineering. [3] University of Pittsburgh, Computer Science.

[1] {jiaxx213, stei0062, kumar001}@umn.edu,[2] lixx5000@umn.edu, [3] xiaowei@pitt.edu

ABSTRACT

Causal graphs are usually considered in a 2D plane, but it has rarely been noticed that within multiple relatively independent timelines, which is comparatively common in causality machine learning, the individual-level differences may lead to *Causal Representation Bias* (CRB). More importantly, such a blind spot has brought obstacles to interdisciplinary applications. Deep Learning (DL) methods overlooking CRBs confront the trouble of models' generalizability, while statistical analytics face difficulties in modeling individual-level features without a geometric global view.

In this paper, we initially discuss the *Geometric Meaning* of causal graphs regarding multi-dimensional timelines; and, accordingly, analyze the scheme of CRB and explicitly define causal model generalization and individualization from a geometric perspective. We also spearhead a novel framework, Causal Representation Learning (CRL), to construct a valid learning plane (in latent space) for causal graphs, propose a particular autoencoder architecture to realize it, and experimentally prove the feasibility.

Involved causal data includes Electronic Healthcare Records (EHR) to estimate medical effects and a hydrology dataset to forecast the environmentally influenced streamflow.

KEYWORDS

Causal Representation, Geometric Meaning of Causal Graphs, Higher-Dimensional Representation, Multi-Timelines in Causality

1 INTRODUCTION

Graphical causal learning is essential for multiple research fields, like meteorology, biology, epidemiology, social sciences, policy-making, etc. [1–3]. It aims to uncover the underlying causal relationship determining how the observational data was generated, to answer questions like "when the flood will happen", "whether the medication works well", and so on. Causality study originated from classical statistics decades ago and constructed the basis for many domain knowledge. But in recent years, technical improvements dramatically sped up the high-quality data collecting and brought a significant challenge to statistical causal analytics: learning graphical causality from data to be Bayesian networks is NP-hard. Meanwhile, Machine Learning (ML) shows the advanced capability of modeling causality from big data. Causality ML has been considered to constitute the bedrock of Artificial Intelligence (AI) [18].

Causal analytics methods are usually knowledge-driven, beginning with priorly specified models. In contrast, the data-driven ML methods are more advantageous in reducing modeling errors and leveraging prior knowledge simultaneously [13]. Unsurprisingly, considerable effort has been expended on applying ML methods to domain causality problems [8, 10]. However, it is strange that they are still far from widespread practical use in many domains,

like healthcare or bioinformatics [7, 9, 12]. A growing consensus is that ML models are more powerful, but statistics can keep models interpretable well [9, 11]. However, understanding why model effectiveness and interpretability can be paradoxical is not intuitive.

Remarkably, some recent works using Neural Networks (NNs) greatly improved efficiency by approximating the discrete causal structure constraint as a continuous optimization function [4, 34], which bypasses the NP-hard combinational searching thoroughly. Meanwhile, considering the black-box nature of Deep Learning (DL), efforts are made toward the interpretable NNs by reconstructing knowledge referring the network architecture [5, 34], but can hardly achieve a general success [6]. If a barrier exists to applying known causality, why hinder DL but not the conventional causal inferences? What makes the "gap" between the two methodologies?

In the EHR data causality learning, it has been shown that some inherent bias exists in DL models [16]. The problem can be straight-forward - Suppose achieving the full effect of medicine M_1 needs one month by expectation, but a patient P_1 only used 20 days because of being very young, and another older patient P_2 used 50 days. So for an RNN model with a 30-day time window, P_1 is counted as a 1.5 effect while P_2 is only counted as 0.6. This may not be a concern for M_1's effect estimation individually (like the typical context setting of statistical causal inference), but for subsequent ones, e.g., the conditional effect of adding M_2 given M_1 is taken, the personal-level biases will persist and snowball to form globally distorted graphical models (like the typical context in causality DL.)

We should model the variables defined by effects instead of time-steps. Notably, it's our models defining the personalized timelines rather than defining *by* some absolute timeline, e.g., time elapses $\frac{1}{3}$ faster for P_1 and $\frac{1}{3}$ slower for P_2 based on same effects.

In this paper, we will initially raise the geometric meaning of causal graphs regarding multi-dimensional (relatively independent) timelines, analyze the inherent bias (1.1), and discuss the relevance between causal models' generalizability concerned in DL and individualized features learning needed by domains (1.2). Then, a new causal learning framework will be proposed as a general solution (1.3). In the methodology section, we will introduce its realization method; and use comprehensive experiments to verify its feasibility. As the principal contributions in this paper, we:

- Initially raise the geometric meaning of causal graphs and identify the source of Causal Representation Bias (CRB);
- Propose a novel framework CRL as a resolution mechanism;
- Propose the architecture design for higher-dimensional representation extraction, a critical step to realizing CRL.

1.1 Geometric Meaning of Causal Graphs

Given a causal graph G with vertices V representing all variables, the causal relations are observable by changing V values along

timeline T. Here T can be either naturally ordered time steps (like the calendar dates or timing hours in one day), or any relatively meaningful time-lapse, e.g., the 30-day observation period is needed to model M_1's effect, no matter when is the actual starting date. To learn any causations, timeline T must exist, so for convenience, we call it *Learning Timeline*, and the plane spanned by the axes of V and T, carrying the graph G, is called *Learning Plane*. Two timelines are *Relatively Independent* if they represent different causal effects, regardless if they are in identical time steps - But they can join because of common variables.

Figure 1 shows a typical scenario of medical effects estimation with two independent timelines T_Y and T_Z joint at variable A. T_Y is the learning timeline in calendar dates, representing causally changed vitals and signs of patients, including their blood lipid LDL, blood pressure BP, and probability of developing type II diabetes, T2D. The relative timeline T_Z describes the medical effects of Statin, whose use is decided by physicians based on LDL criteria ($A \rightarrow S$) and will directly reduce LDL values ($S \rightarrow A'$) and subsequentially decrease the T2D risk ($A' \rightarrow B'$). Here we aim to predict T2D risk given the patient's LDL, BP, and Statin use in previous time steps.

The learning plane (the upper-right figure) only carries causations among $\{A, B, C\}$, but excluding S, whose timestamp t is not meaningful here, and a circle $A \rightarrow S \rightarrow A$ is formed. Along the timeline T_Z (the lower-right figure), the circle no longer exists because of two split roles of A before and after taking Statin. The target learning effects are colored in green, which we usually assume to define an *eligible* learning plane by default.

Theorem 1. *A valid causal learning plane must satisfy that the underlying causality of any samples in the data can be represented as a linear transformation of the causal graph carried by this plane.*

Corollary 1. *If a causal graph crosses multiple independent timelines but contains no confounding relations, it is in a valid learning plane.*

Theorem 1 defines the geometric basis of causal learning. In the simplest case, modeling pair-wise causation is equivalent to estimating the correlation between two variables so that deviations can be minimized and normally distributed around the average estimation, i.e., randomly linear transformed from it. By extension, modeling causal graph G means that all the pair-wise causations, AKA edges in G, must be linearly transformed in a *single* transformation for any data samples. Since two edges define a 2D plane, in Corollary 1, the graph can define a hyperplane valid for learning.

In Figure 1, our concern is the confounding over $\{A, S, B'\}$. Considering two edges only, $A \rightarrow S$ and $A \rightarrow B'$, they can span a valid plane according to Corollary 1. Since the former edge belongs to T_Z while the latter one crosses T_Z and T_Y ($\overrightarrow{AB'} = \overrightarrow{AA'} + \overrightarrow{A'B'}$), this spanned plane has determined the relative proportion of T_Z and T_Y, AKA their relative time-lapse speed, because, for any data samples, this proportion is fixed in the single linear transformation. Then, the third edge $S \rightarrow B'$ crossing T_Z and T_Y has been determined passively. So, assuming the triangle defines a 2D plane is to take the coincidence that the causal effect $S \rightarrow B'$ in data accidentally represents the required edge. This risk of bias is commonly ignored in causality machine learning. It is, therefore, necessary to propose the name Causal Representation Bias (CRB) for explicit reference.

1.2 Individualization and Generalization

Interestingly, CRB comes from the heterogeneous *individual-level* differences in cohort data, e.g., one cannot infer that P_1 is always 0.6 times faster than P_2 for any medical effects just based on M_1. In health informatics, assuming no CRB commonly means all samples are from the same patient who runs a consistent causal system in the body, thus, whose underlying causal graph G can be modeled in one-time learning, i.e., the valid learning plane carrying G exists. However, it is still possible that individual-level features are not heterogenous but linearly differ in some other domains' questions.

Corollary 2. *The common causal knowledge G of a cohort cannot be none-CRB modeled in one-time learning unless all <u>individualized</u> causality can be represented as linear transformations of G.*

For EHR data, obtaining enough samples to build a model for one patient is almost impossible. Thus learning individual-level features is very challenging for statistical analytics in the domain. On the other side, machine learning can model a large amount of heterogeneous data simultaneously, but due to involved CRBs, such established models can hardly be generalizable.

Recently, Scholkopf et al. [17] identified the causal model generalization problem and first raised the term *Causal Representation*. They point out that "most work in causality starts from the premise that the causal variables are given" (i.e., the valid learning plane is given = no CRB) and describes three critical challenges we are facing on the way to realizing Artificial Intelligence (AI): 1) The robustness of modeling is still weak; 2) We lack reusable mechanisms to generalize learned causal knowledge; 3) No general effect way to extract the most common causality from heterogeneous data.

A typical generalization problem is described in our experiments (Section 4), from the hydrology area: Since the environmental conditions cause streamflow changes complying with the identical physical regulations on the earth, how can we build structural
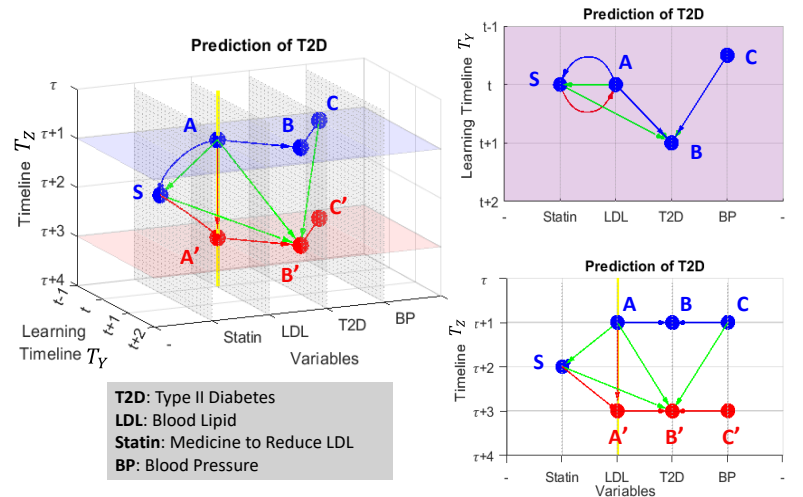

Figure 1: A 3-dimensional causal graph within two timelines. The learning target is to predict the risk of developing T2D given values of BP and LDL, which can be causally impacted by using Statin. ($A \rightarrow S$ and $S \rightarrow A'$ are intensionally curved to distinguish overlapped edges.)

models to describe its underlying complex causality, which can be generalized to any watershed on the earth?

The connection between the causal model's *individualization* and *generalization* can be easily understood by analogy: if we consider a watershed the same as an individual patient, the only difference between the hydrology and EHR data is that any individual in the former can produce plenty samples as much as needed to build models, while the latter one cannot. Thus, the EHR data has *one more dimension* than the hydrology data from one watershed, since EHR contains numerous patients daily visiting the hospital.

Corollary 3. *Two different data have isostructural underlying causality G_1 and G_2, then their causal models can only be* <u>*generalized to each other via the shared learning plane of G_1 and G_2.*</u>

For a cohort dataset with n records from m individuals whose underlying common causality is represented as graph G, causal individualization means finding $\{G_j\}$ compared to G, where $j = 1, \ldots, m$; and causal generalization means finding G generalizable to $\{G_j\}$, or finding G_j by given G_i, where $i \neq j \in 1, \ldots, m$.

Corollaries 1-3 emphasize that for a general causality G involving multiple independent timelines, the valid learning plane may not naturally be given but must be intentionally constructed, via multiple times of learning. The complexity of confoundings across timelines determines the minimum number of learnings. E.g., In Figure 1, we need at least two learning processes to model the green-colored relations: $A \rightarrow S$, $A \rightarrow B'$, and $C \rightarrow B'$ can be in one, and $S \rightarrow B'$ has to be in another one as a pre-process or post-process to adjust the CRB. We can explicitly adjust the B'' value [16], but it is more generally feasible to realize it in the latent space.
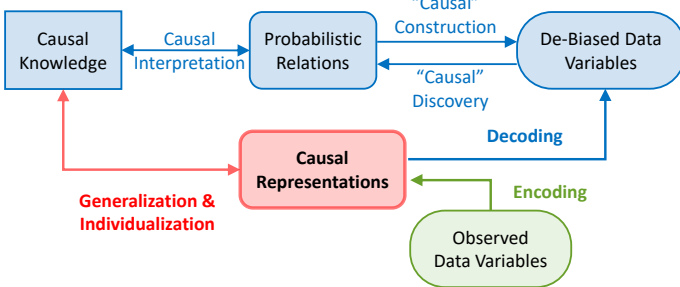

Figure 2: The proposed framework Causal Representation Learning (CRL). Red = the causal representation feature space; Green = the original data space; Blue = the reconstructed data space with de-biased variables. Here "causal" means it is correlation indeed; causal construction means modeling known causations, like causal effects estimation between specified variables.

1.3 Causal Representation Learning (CRL)

As shown in Figure 2, the proposed CRL method uses DL-based representation extractions to construct a latent feature space in which all data variables are individually represented as different feature vectors. Such latent space plays the role of "variables axis". Because the feature vectors' values can be modified as needed and kept representative simultaneously, potential CRBs can be easily adjusted; in other words, autoencoders allow us to modify the variables' values from time-steps-defined to causal-effects-defined.

Under the conventional framework, causal learning is restricted in observational data space based on unmodifiable variable values. Further, considering the black-box nature of neural networks, it is reasonable for many DL works to assume data variables adequately represent underlying relations, to avoid further demand for interpretation. And the advantage of DL in handling numerous variables on large-scale data can make CRBs easier to be overlooked.

The term "CRL" was initially raised by Scholkopf et al. [17], but they mainly focus on specific questions in computer vision instead of on universality. In our proposed framework, the most conservative strategy is to model pair-wise causations one by one in the latent space until going through all edges of G. In such a process, the feature vectors representing data variables can have values changing but without loss of representativeness. Reconstruction via autoencoders (encoding + decoding) is to realize the nonlinear transformations between the latent learning plane and the original data space; but differently, the reconstructed variables are friendly to one-time learning without concerns about potential CRBs.

Previously, there may not be a strong demand for CRL, because leveraging prior knowledge to calibrate models case-by-case can commonly satisfy the requirement. But nowadays, with the information explosion, statistical models are approximately reaching their capability ceiling, and machine learning has to face larger-scale questions much more than before.

Theorem 2. *Suppose data matrix X is augmented by column vectors of all variables in G, then the feature space adequately representing G must have a dimensionality larger than $rank(X)$.*

Theorem 2 is derived from the principle that the autoencoder learns the subspace spanned by the top principal components of X [20–22], which is more well-known as PCA. It implies that, unlike the regular autoencoder for lower-dimensional representation, here we need to design the one to realize higher-dimensional from scratch, as introduced in Section 3.

We have a further hypothesis that if reducing the feature space's dimensionality to be lower than $rank(X)$, the latent causal relations can be more causally (but less adequately) representative than G in the original data space. It is inspired by the principle that autoencoders can produce aligned latent spaces by stretching along the top singular vectors of X [23], which has been widely used in NLP for identifying word analogies [24, 25], and in bioinformatics to embedding gene expression[26, 27]. But this hypothesis needs further corroboration; once verified, it is promising that CRL can also help to improve causal inference efficiency.

2 RELATED WORK

2.1 Existing Causal Models

Conventionally, learning causality is to establish Bayesian networks (BN) on the observed data variables by encoding the conditional independencies between variables as directed acyclic graphs (DAGs). The traditional statistical methods include constraint-based ones and score-based ones [13]. The constraint-based causal discovery, including PC and fast causal inference (FCI), typically detects causation by exploiting conditional independences among data. PC is based on a no-hidden-confounder assumption (commonly called the causal *sufficiency* assumption), while FCI is not. However, there

exists a significant deficiency: they can necessarily but not sufficiently identify causations. Because many possible causal structures can satisfy the same conditional independence, where the set of these structures is called a Markov equivalence class [14]. The score-based methods usually perform heuristics, to search through the space of all possible DAGs using a pre-defined score function, e.g., greedy equivalence search (GES) directs its search over the space of equivalence classes under the no-hidden-confounder assumption.

From the view of computer science, the directed graphical causal models, often presented as Functional Causal Models (FCMs) or Structural Equation Models (SEMs) [14, 15], are commonly used machine learning methods to learn causality based on causally interpreted DAGs. They assume the value of Y to be a deterministic function of its direct causes X with noise term ϵ, $Y = f(X, \epsilon; \theta)$ where θ indicates the parameters set involved in f. They essentially aim to decompose a causal DAG into a product of more superficial vectors, where the separation is by conditional independence. It has been shown that they can distinguish different DAGs from an equivalence class and are good at leveraging prior knowledge [14].
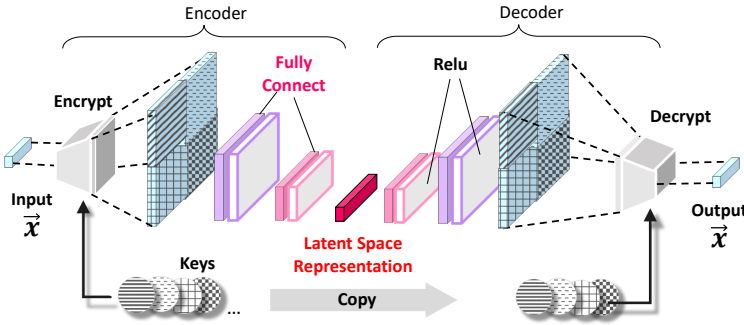

Figure 3: Proposed architecture of the higher-dimensional representation extraction autoencoder.

2.2 Deep Learning in Causality

In recent years, Deep Learning (DL) has been applied to realize highly efficient graphical causal models. These applications established an elegant mathematical connection between classical combinatorial searching algorithms and the continuous optimization techniques of machine learning [6]. The key novelty is to turn the discrete possible-DAGs-space searching problem into a smooth function optimization question. Precisely, these methods quantify the "acyclic-ness" of the graph's adjacency matrix as a continuous constraint. For instance, Lachapelle et al. described how to convert the DAG learning into a pure neural network problem in 2019 [5]; and Zheng et al. transformed it into a generalized nonparametric problem that requires no modeling assumptions in 2020 [4].

Given these achievements, people also made great efforts toward the interpretable DL causal models. One way is to determine conditional independence by finding zero derivatives of the graphical causal functions [5]. Another way is to learn it from the weights of the neural networks [4]: among all Markov causal paths that lead the direction from x to y, if at least one of them has weight 0, then y does not depend on x. Meanwhile, the adjacency matrix is optimized with the DAG constraint by maximizing the log-likelihood

using the augmented Lagrangian method. Such a weights-detecting methodology is called neural architecture search (NAS) [6].

However, this strategy has not achieved general success [6]. Identifying how well the knowledge has been utilized in optimization without meaningful insight into networking is quite challenging. The geometric explanation of causal graphs proposed in this paper may help find the reason - The representation biases could have been inherently determined when DAGs are considered in a 2D plane, which assumes the overall one-time learning is valid by default. People barely notice that 2D graphs are used for awareness models only, based on our intuitive understanding of events along timelines. But for AI, "causation" is not an actual conception; like illustrated in Figure 2, it is just a correlation whose causal meaning is defined by human-specified time-lapse.

3 METHODOLOGY OF CAUSAL REPRESENTATION

From the probabilistic perspective, causal representation learning (CRL) is to "simulate" the dependent relations among data variables inside the latent space. The set of causal variables corresponds to a group of latent feature vectors, and the distances between them allow us to model their correlations in this space. For data variable x and its corresponding latent vector h, their joint distribution can be decomposed as $P(x, h) = P(x|h)P(h)$, where $P(h)$ is a *prior* and $P(x|h)$ is a *likelihood*. Thus a general latent model can be inherently interpreted as a latent cause model [19] directed from h to x, denoted as $h \rightarrow x$. This is the theoretical basis of realizing CRL.

Consequently, we can reasonably decomposite the causation from x to y (denoted as $x \Rightarrow y$) as a causal chain $x \rightarrow h \rightarrow v \rightarrow y$ where v is the corresponding latent vector of y, and $x \rightarrow h$ relies on a *posterior* $P(h|x)$. Such that the dependency between x and y is represented as the *posterior* distribution $P(v|h)$. Suppose we have the parametrized model $v = f(h)$, then tuple (v, h, f) can completely represent the causation $x \Rightarrow y$ in latent space. To be convenient, we call f as the *Latent Causal Effect*.

The ultimate goal of CRL is to build up a *Directed Graphical Model* in the latent space by "stacking" these posterior distributions, which are flexible to be generalized or individualized. For example, given causation $x \Rightarrow y \Rightarrow z$, we are interested in the indirect causal effect of x on z via y, decomposed as $x \rightarrow h \rightarrow v \rightarrow t \rightarrow z$, where t is the latent vector of z, and we want to know $P(t|h) = P(t|v)\big|_{v=f(h)}$. Since f is known, we only need to learn the latent causal effect g, such that $t = g(v)\big|_{v=f(h)}$ by stacking the unknown *posterior* $P(t|v)$ onto the known *posterior* $P(v|h)$.

In the following, we will start from the proposed autoencoder architecture to realize the required higher-dimensional representation extraction (3.1) along with the initial design of its critical layers (3.2); then, introduce how to model the latent causal effect and stack them to complete a graphical structure (3.3); and lastly demonstrate a causal discovery algorithm in latent space (3.4).

3.1 Autoencoder Architecture

The regular autoencoder intends to deal with large-volume objects (like images with numerous pixels) and, accordingly, extract their lower-dimensional features, which is more like a distillation process. However, our appeal is to individually convert the causal

variables into a higher dimensional latent space and then model their dependencies (i.e., causal effects). Considering the graphical complexity, this latent space should be able to provide sufficient freedom for all possibly needed causal modeling. Corollary 2 gives a lower boundary for the dimensionality, while we temporally shelve further discussion about its upper boundary. In our experimental dataset, lengths of the 10 variables are between $1 \sim 5$ and added up to be 32 in total (refer to Table 2); their average node degree will be around $2 \sim 3$. Initially, we approximate that 64-dimensional latent space is about to satisfy the requirement, but it turns out to be highly redundant. It is probably because variable attributes are strongly related, and we gradually reduced the dimension to 16.
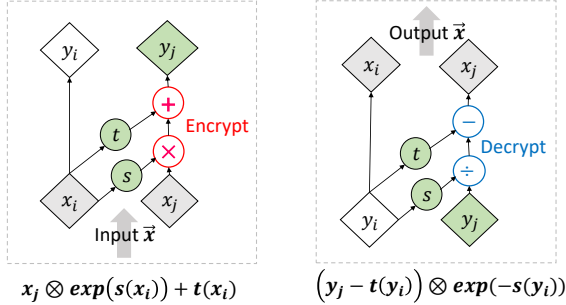

$$x_j \otimes exp(s(x_i)) + t(x_i) \qquad \left(y_j - t(y_i)\right) \otimes exp(-s(y_i))$$

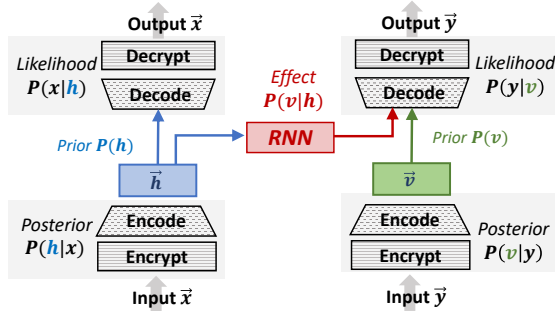**Figure 4: Encrypt and Decrypt with Key θ = (s, t).**

Figure 5: Architecture of learning the latent causal effect $h \to v$ as a RNN model.

For better encoding + decoding, input variables need a length extension, where the extended one should sufficiently encode the associative information among its values. More importantly, this process has to be *Invertible* to achieve high accuracy for the output vector, unlike the regular autoencoders without concern of reconstructing each pixel. We design a pair of layers to realize a pair of symmetric processes, *Encrypt* and *Decrypt*, respectively located at the Input and Output of the autoencoder, as shown in Figure 3.

The Encrypt works as a "feature amplifier" for extracting higher-order features from the input vector, while Decrypt exactly reverses this process. How much the feature order should reach is on practical demand, such as a *double-wise* amplifying to extract the 2-dimensional associations, or even higher like *triple-wise* to encode 3-dimensional features further. In this work, we only demonstrate the double-wise extraction. To be specific, all possible combinations of two values from this vector will be "encrypted" using a set of constants named *Key*, generated by Encoder. Then the *Key* will be

copied by the Decoder to "decrypt" its output back to a pair of such values. For a pair, one *Key* means one-time amplifying, so with multiple *Keys*, we can get a group of vectors differently amplified from the same two values; then concatenate them to form a much longer new vector to be a double-wise feature of the input pair of values. Because for a variable with length n, its double-wise features can be arranged as a $(n-1) * (n-1)$ sized matrix, we use a square to represent it in Figure 3 (not meaning a 2-dimensional feature). The displayed occasion includes four *Keys*, which means extracting four times amplified features. As a metaphor, in this architecture, the role of *Keys* is similar to the filters in a regular autoencoder.

3.2 Variable Encrypt and Decrypt

Suppose the input variable denoted as $\overrightarrow{X} = (x_1, x_2, \ldots, x_n)$ has length n. The Encrypt is defined as a function $f(x_j; x_i, \theta)$, a transformation of x_j using x_i as weights and use the *Key* θ as parameters, where $i \neq j$ and $i, j \in \{1, \ldots, n\}$.

Let $\theta = (s, t)$, s.t. $s(x) = w_s x$ and $t(x) = w_t x$, given w_s and w_t. We define that $f(x_j) = x_j \otimes exp(s(x_i)) + t(x_i)$, where $\otimes$ denotes the element-wise multiplication, and let $y_j = f(x_j)$ to be the corresponding output for x_j. Accordingly, the inversed function f^{-1} in Decrypt is going to be $(y_j - t(y_i)) \otimes exp(-s(y_i))$. Figure 4 displays the two symmetric processes. Because calculation of f^{-1} does not involve inversed function s^{-1} or t^{-1}, the two basic transformation functions s and t can be defined very flexible to be non-linear.

We assemble all f functions as a set $\mathcal{F}(X; \Theta)$, where Θ is the set of all θ. Then the two layers of Encrypt and Decrypt can be denoted as $Y = \mathcal{F}(X; \Theta)$, and $X = \mathcal{F}^{-1}(Y; \Theta)$ respectively. Source codes are provided as a complete demo[1]. It is worth highlighting that the work of Dinh, L. et al. inspires the design proposed above [35]. As a possible extension, one can perform higher-order associations using the same principle, e.g., a triple-wise feature extraction.

3.3 Latent Causal Effects Stacking

Figure 5 displays the architecture of learning the latent causal effect $h \to v$ for representing the causation $x \Rightarrow y$, which is to construct the causal chain $x \to h \to v \to y$ using *RNN* model. The feature vectors h and v have been established as latent vectors corresponding to x and y, respectively, which have the same dimensionality ($h, v \in \mathcal{R}^L$), but independent to each other. In this latent causal effect learning process, h and v will keep updated during optimizations to get "closer" to each other. It can be seen that both h and v are shifting in $\mathcal{R}^L$ space toward finding the proper locations minimizing their distance and, at the same time, letting the *RNN* sufficiently model their dependence $P(v|h)$, which represents the estimated causal effect $h \to v$ in $\mathcal{R}^L$.

There are three optimization tasks with three different objective functions, sequentially executed in each optimization iteration:

(1) Update *posterior* $P(h|x)$ and *likelihood* $P(x|h)$ to minimize the reconstruction error of x.
(2) Update *posterior* $P(h|x)$ and latent causal effect *posterior* $P(v|h)$ to minimize the reconstruction error of y.
(3) Update *posterior* $P(v|y)$ and *likelihood* $P(y|v)$ to minimize the reconstruction error of y

---

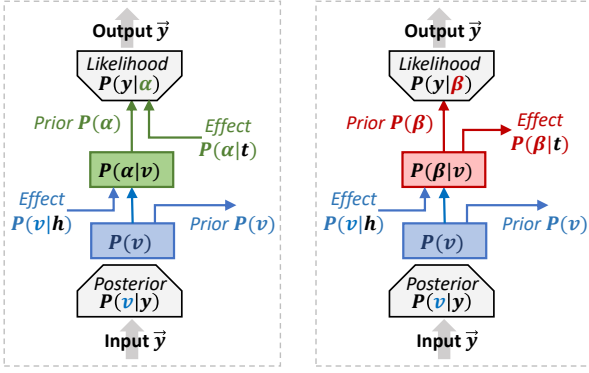[1]https://github.com/kflijia/bijective_crossing_functions.git

Figure 6: Example of stacking two pair-wise causations to represent the complete causal chain over x, y and z.

Figure 7: Example of Causal Discovery in the latent representation feature space.

"Stacking" two causations in the latent representation feature space means that given one is well established, we aim to build up the second one starting from either end of the first, to ultimately establish a 3-nodes causal chain, which can eventually shape as a colliding, a confounding, or a chain according to needs. This is essentially equal to stack two *posteriors*, e.g., stack $P(t|v)$ onto $P(v|h)$ to form the chain $h \rightarrow v \rightarrow t$. There are four possible occasions by combining two binary options: is the *prior* at the bottom a *cause* or a *result*? And how will the *prior* at the top be?

Suppose data variable y is the connecting point where the stacking will happen, and we have x and z at each side of y to be the candidates for connection, represented as vectors h and t. As shown at Figure 6 left side, the *prior* $P(v)$ at the bottom can be either a *cause* to the other one (output toward the right) or the *result* caused by h (input as causal effect $P(v|h)$). While for the top layer, as shown at Figure 6, there are two possible roles of t: to be the new *cause* that inputs the effect $P(\alpha|t)$ (green colored at left), or the new *result* that accepts the effect $P(\beta|t)$ (red colored at right). In the optimization process of the top layer, t always represents the fixed *prior* to support updating the *posterior*, either $P(\alpha|t)$ or $P(\beta|t)$.

For each occasion, multiple inputs and outputs exist to be the potential entrance and exit, respectively, where the selection would be on demand during causal constructions or discoveries. For example, let $\mapsto$ be the notation that connects entrance and exit, then, $P(v|h) \mapsto y$ means output $P(y|x)$, $P(\alpha|t) \mapsto y$ means output $P(y|z)$; $y \mapsto P(\beta|t)$ means output $P(z|y)$ with the ground truth y input, while $P(v|h) \mapsto P(\beta|t)$ means output $P(z|y)\big|_{y=f(x)}$ where f is the estimated causal effect $x \Rightarrow y$.

From a geometric perspective, the desired representation feature space (as a valid learning plane) is further specified every time we complete a stacking in $\mathcal{R}^L$. Suppose L is large enough to provide a sufficient degree of freedom throughout the causal graph. In that case, we can finally obtain a group of feature vectors in $\mathcal{R}^L$, which have been optimized to distribute as tight as possible.

3.4 Discovery Algorithm in Latent Space

Here we adopt the heuristics strategy to realize causal discovery in latent space among the well-established feature vectors representing variables; and use KLD (KL-Divergence) as the metric to evaluate the similarity between two feature vectors in the latent space $\mathcal{R}^L$, as shown in Algorithm 1. Another consideration is
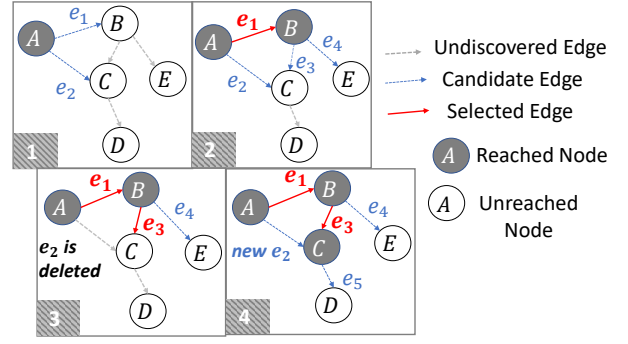
adopting the conventional MSE evaluation. However, it has been demonstrated that the MSE-based optimization subject to acyclicity constraint has some inherent restrictions, such that the model may be misled and eventually dominated by the data variances instead of the underlying causal relations [32, 33].

An illustrative example of the latent space causal discovery is given in Figure 7. In the four consecutive steps, edge e_1 and edge e_3 are selected one after another, since the selected e_1 make node B reachable, and B is the start node of e_3. At step 3, the learned causal effect of e_2, from A to C, is deleted from the candidate edges and need to be recalculated, because edge e_3 has reached C and changed the causal conditions on C.

Algorithm 1: Latent Causal Discovery with KLD Metric

Result: ordered edges set $\mathbf{E} = \{e_1, \ldots, e_n\}$
$\mathbf{E} = \{\}$;
$N_R = \{n_0 \mid n_0 \in N, Parent(n_0) = \varnothing\}$;
while $N_R \subset N$ **do**
   $\Delta = \{\}$;
   for $n \in N$ **do**
      for $p \in Parent(n)$ **do**
         if $n \notin N_R$ and $p \in N_R$ **then**
            $e = (p, n)$;
            $\beta = \{\}$;
            for $r \in N_R$ **do**
               if $r \in Parent(n)$ and $r \neq p$ **then**
                  $\beta = \beta \cup r$
               end
            end
            $\delta_e = K(\beta \cup p, n) - K(\beta, n)$;
            $\Delta = \Delta \cup \delta_e$;
         end
      end
   end
   $\sigma = argmin_e(\delta_e \mid \delta_e \in \Delta)$;
   $\mathbf{E} = \mathbf{E} \cup \sigma$;
   $N_R = N_R \cup n_\sigma$;
end

$G = (N, E)$	graph G consists of nodes set N and edges set E
N_R	the set of reachable nodes
$\mathbf{E}$	edges in order of being discovered
$K(\beta, n)$	KLD metric between the causes set β and effect node n
$\Delta = \{\delta_e\}$	the set of all KLD Gain δ_e for each candidate edge e
n, p, r	notation of node
e, σ	notation of edge

4 FEASIBILITY EXPERIMENTS

The feasibility experiments are designed to verify that the proposed autoencoder architecture can effectively perform higher-dimensional representation and subsequently, can successively realize the latent space causal effect learning. Eventually, we aim to establish a complete graphical causal structure in the latent space, based on which the general causal discovery and construction processes can be accurately performed.

The adopted experimental data is a professional synthetic hydrology dataset, with the learning task of forecasting streamflow by given observed environmental conditions (e.g., temperature and precipitation). The purpose of performing causal representation learning (CRL) on this hydrology data is to realize causal model generalization among different watersheds, such that the causal model established on the dataset X_1 collected from watershed No.1, can be calibrated to fit the dataset X_2 from watershed No.2. This application is motivated by sharing common knowledge across different data qualities. For example, X_1 is qualified enough to establish a complete graphical causal model, but X_2 does not. It has been known that their underlying hydrological schemes are very similar (e.g., geographical positions are closed), but because of various unmeasurable conditions (e.g., different economic developments and land use), directly applying X_1's model on X_2 performs poorly. Moreover, the existing causal models are based on physical modules with limited parameters. Thus these models' degree of freedom is far from being able to do calibrations on low-qualification data.

Because of space limitation, in this paper, we can only finish the CRL modeling process without the experimental performance of causal model generalization. In the subsequent work, we plan to implement both individualization and generalization of CRL models. Besides, due to empirical restrictions, we are out of access to EHR data for this work; for proof of the existence of causal representation bias (CRB) in EHR, please refer to another work [16].

4.1 Hydrology Dataset

SWAT (Soil & Water Assessment Tool) is a professional hydrology data simulation system, which is commonly used to generate synthetic datasets or calibrate realistic data samples based on physical modules. Our experiments are based on the synthetic data generated by SWAT system, for simulating the Root River Headwater watershed in Southeast Minnesota. From this dataset, we selected successive 60 virtual years with a daily update frequency. The performances are mainly evaluated as the accuracies of unsupervised data reconstruction via the proposed CRL model.

The hydrology system entails water fluxes and states across the earth's space, such as snowpack melting, evapotranspiration, and soil moisture. Its underlying causality is highly complex, exhibits nonlinearities, and contains a lot of unobservable dependencies. In recent years, machine learning on causal inferences naturally attracted hydrologists' attention [28]; DL-based methods are widely used to extract representations from time series efficiently. As a typical application, RNN models contribute state-of-the-art techniques for streamflow prediction [29–31]. Here, the word "streamflow" refers to a critical statement variable in hydrologic processes, observed by numerous monitors in the water body, and influenced by comprehensive physical factors over a period of months.

Table 1: Explanations of the nodes in Figure 8

ID	Variable Name	Explanation
A	Environmental set I	Wind Speed + Humidity + Temperature
B	Environmental set II	Temperature + Solar Radiation + Precipitation
C	Evapotranspiration	Evaporation and transpiration
D	Snowpack	The winter frozen water in the ice form
E	Soil Water	Soil moisture in vadose zone
F	Aquifer	Groundwater storage
G	Surface Runoff	Flowing water over the land surface
H	Lateral	Vadose zone flow
I	Baseflow	groundwater discharge
J	Streamflow	Sensors recorded outputs
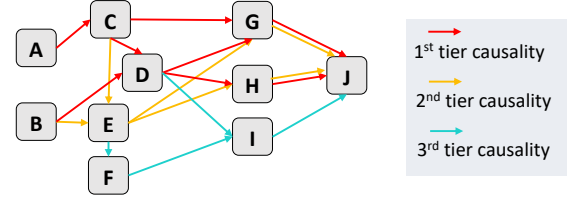
Figure 8: The underlying causal graph from current hydrology knowledge. The reducing expert-expected causal strengths define the tiers of causal routines.

Figure 8 displays the ground truth graphical causal relationship used by SWAT to generate synthetic data based on current hydrology knowledge. Explanations about the nodes are listed in Table 1. The causal strengths of these routines are determined by their present contributions to output streamflow, remarked by different colors. The hydrological interpretation is that the surface runoff routine (1st tier causality) cause streamflow peaks more quickly than the lateral flow routine (2nd tier causality), which is more critical than the baseflow dynamics (3rd tier causality).

4.2 Autoencoder Reconstruction Test

The proposed higher-dimensional representation extraction autoencoder is used to reconstruct each node of the causal graph. Accordingly, 10 feature vectors have been extracted in the latent space to represent the nodes from A to J, respectively. The major challenge comes from the low dimensionality of the original data variables. As shown in Table 2, the lengths of these variables maximally reach 5, while the prediction target, node J, is just a one-column variable. We first expand each variable to have a new length of 12 by randomly repeating their columns, then augmented with 12-dimensional dummy variables of months to form 24-dimensional input variables. In the double-wise feature amplifications, every column has been extracted 23 times and augmented with itself. In this way, we obtain a 576-dimensional output through the *Enrypt* process. In our experimental context, the representation feature space is set to be 16-dimensional.

Table 2 shows each variable's statistics characteristics and the reconstruction performance (comparing the output and input of the autoencoder) evaluated by RMSE, where less RMSE indicates higher accuracy. We provide two RMSE evaluations for each variable in the scaled (i.e., normalized) and original values, respectively. The characteristic columns are calculated as scaled values.

Table 2: Performance of Autoencoder Reconstruction.

Variable	Length	Mean	Std	Min	Max	Non-Zero Rate%	RMSE on Scaled	RMSE on Original	BCE of Mask
A	5	1.8513	1.5496	-3.3557	7.6809	87.54	0.093	0.871	0.095
B	4	0.7687	1.1353	-3.3557	5.9710	64.52	0.076	0.678	1.132
C	2	1.0342	1.0025	0.0	6.2145	94.42	0.037	0.089	0.428
D	3	0.0458	0.2005	0.0	5.2434	11.40	0.015	0.679	0.445
E	2	3.1449	1.0000	0.0285	5.0916	100	0.058	3.343	0.643
F	4	0.3922	0.8962	0.0	8.6122	59.08	0.326	7.178	2.045
G	4	0.7180	1.1064	0.0	8.2551	47.87	0.045	0.81	1.327
H	4	0.7344	1.0193	0.0	7.6350	49.93	0.045	0.009	1.345
I	3	0.1432	0.6137	0.0	8.3880	21.66	0.035	0.009	1.672
J	1	0.0410	0.2000	0.0	7.8903	21.75	0.007	0.098	1.088

Table 3: Brief Summary of the Latent Causal Discovery Results.

Edge	A⇒C	B⇒D	C⇒D	C⇒G	D⇒G	G⇒J	D⇒H	H⇒J	B⇒E	E⇒G	E⇒H	C⇒E	E⇒F	F⇒I	I⇒J	D⇒I
KLD	7.63	8.51	10.14	11.60	27.87	5.29	25.19	15.93	37.07	39.13	39.88	46.58	53.68	45.64	17.41	75.57
Gain	7.63	8.51	1.135	11.60	2.454	5.29	25.19	0.209	37.07	-5.91	-3.29	2.677	53.68	45.64	0.028	3.384

It is worth mentioning that the hydrology dataset contains a large number of meaningful zero values. For example, the variable D, named Snowpack, only has non-zero values in winter, whose water body will present as Soil Water (variable E) in the other seasons. However, the zeros do have meanings since they represent the variables vanishing. Therefore, we perform double reconstructions simultaneously in the autoencoder: one for the variable's continuous values, and the other to be the non-zero indicator, evaluated by BCE and named Mask in the table. The RMSE performances in Table 2 are obtained by multiplying these two results. The percentage of non-zero values is also provided for each node. These shallow RMSE values indicate the success of the reconstruction processes, which are in the range of [0.01, 0.09], except node F, the Aquifer variable. It has been known that the aquifer system modeling is still premature in the present hydrology area. So this is reasonable to infer that in the synthetic data, Aquifer is closer to random noise than other variables.

4.3 Latent Causal Effects Learning Test

The latent causal effect learning is evaluated on pair-wise causations. And by stacking them, we successfully recover the ground truth graphical causal structure displayed in Fig 8. To compare the changing of the reconstructing performance by each stacking, we construct the Table 4 for each variable that has ever been the *result* node of pair-wise causations. For example, node G has three possible causes C, D, and E, so we can build four causal effect models in total: the individual pair-wise causations $C \Rightarrow G$, $D \Rightarrow G$, $E \Rightarrow G$, and the stacked causal effects $CDE \Rightarrow G$. For convenience, we call them "*pair-effect*" and "*stacking-effect*" respectively in the following. In Table 4 we also list the initial reconstruction performance for each variable, as the comparison baseline, in the column named "Variable Reconstruction (initial performance)".

There are three different optimization tasks in the latent causal effect learning process (refer to Section 3.3), among which both the *second* and *third* one are about the *result* node. Therefore, in Table 4, we give out the performances of these two roles for each node: the role in the *third* optimization is shown as the columns named "Variable Reconstruction (as result node)"; and the role of the *second*

one is in the columns named "Latent Causal Effect Reconstruction", which represent the causal effect learning performances of RNN models, along with the KLD metrics to display the learned causality strength (the lower is stronger).

The KLD metric differences among variables are easy to be observed. Variable J has the smallest KLD values, which means that G (Surface Runoff), H (Lateral), and I (Baseflow) significantly cause J (Streamflow). While the weaker causal relations present as higher values of KLD. For example, variable I can hardly be predictable given D and F. For *result* nodes, D, E, J, the *stacking-effect* causal strength is in the intermediate range of *pair-effect* strength, which implies that the internal associative relationship inside their cause nodes is ambiguous. For G and H, the *stacking-effect* causal strength KLD is lower than all the *pair-effect* ones. It is reasonable to infer that the *stacking-effect* model has captured additional causal information from the associative relationships among their causes nodes. Besides, the KLD metric also tells which cause node contributes the most to the latent causal effect. For example, $C \Rightarrow G$ strength is closer to $CDE \Rightarrow G$ than the other causes nodes, which indicates C to be the strongest source in this causal effect.

To better present the reconstruction performances of the stacked causal structure, we illustrate their results as data simulations in Figure 9, which includes three different nodes, J, G, and I in the same synthetic year. For each node, the initial variable reconstruction performance is plotted as the blue line, and all the latent causal effects we have learned are displayed in different colors. Besides RMSE, here we also use another metric, NSE (Nash–Sutcliffe model efficiency coefficient) to be the hydrology-meaningful accuracy evaluation, where NSE = 1 defines the best prediction. The three initial variable reconstructions (blue lines) in Figure 9 almost reach the best performance, as they overlapped with the ground truth observations (black dots). The red lines indicate simulations from *stacking-effect*, whose performances are highly consistent with our analysis above: Node J has the best prediction; node I can hardly be predictable; and node G is predicted best by CDE causes combination, among which C provides the strongest causality.
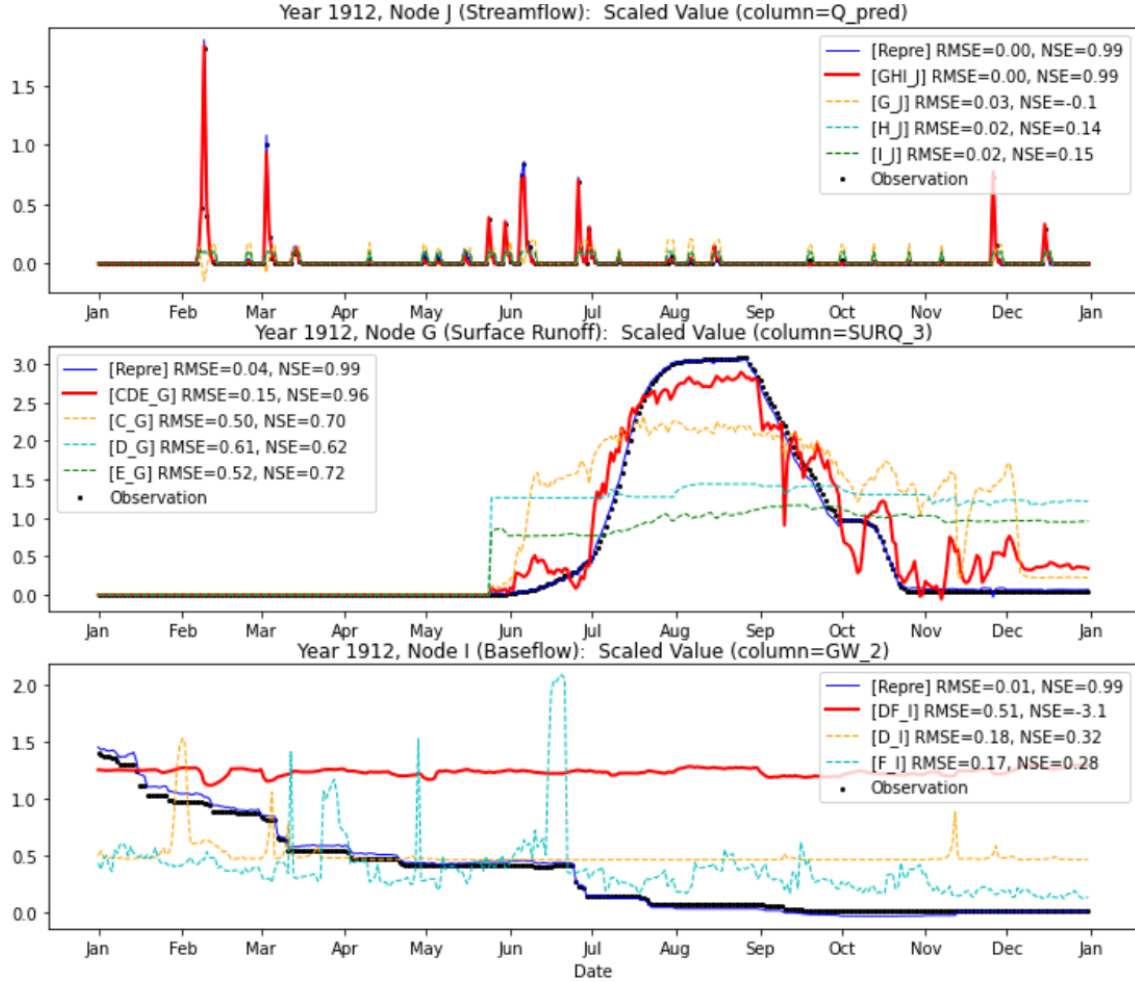
Figure 9: Performance of Latent Causal Effect Reconstruction Examples. (evaluated as data simulation accuracy)

4.4 Causal Structure Discovery Test

The latent causal discovery is based on the KLD causality strength evaluation. 3 displays the edges' discovered order and their corresponding KLD and KLD-Gains. Here the table cells are colored according to the different tiers of causality, consistent with the ground truth shown in Figure 8. Since the discovery algorithm successfully distinguished the 3 tiers, the proposed KLD metric has been proven to work effectively.

Due to limited space, the results in Table 3 is only a brief version, whose complete version is provided in Appendix A Table 1, which includes all results during the causal discovery, organized by the round of detection. As a baseline comparison, we also performed the conventional FGES (fast greedy equivalence search) causal discovery method in 10 cross-validation. Their results are given in Appendix A Table 2, with apparently worse performance than the proposed CRL method.

5 CONCLUSION

Recently, the application ChatGPT has become very popular, and we are interested in how well it can replace humans in doing intellectual work. On the contrary, we barely hear AI can replace doctors in making prescription decisions, or cracking the genetic code automatically; although they have much fewer computational complexities than learning tens of languages, computing pixels of movies, or evaluating billions of possibilities in the Go game. So, what sealed AI's capability when learning structural data?

Understanding the timeline is a fundamental difference between humans and AI, distinguishing causality from correlation. Thus, we need to eliminate the causally formed heterogeneity (i.e., the individualized time-lapse differences) before feeding the structural data to AI. We hope the proposed CRL can lead to the next trend.

REFERENCES

[1] Wood, Christopher J., and Robert W. Spekkens. *The lesson of causal discovery algorithms for quantum correlations: Causal explanations of Bell-inequality violations require fine-tuning.* New Journal of Physics 17.3 (2015): 033002.
[2] Vuković, Matej, et al. *Causal Discovery in Manufacturing: A Structured Literature Review.* Journal of Manufacturing and Materials Processing 6.1 (2022): 10.
[3] Ombadi, Mohammed, et al. *Evaluation of methods for causal discovery in hydrometeorological systems.* Water Resources Research 56.7 (2020): e2020WR027251.
[4] Zheng, Xun, et al. *Learning sparse nonparametric dags.* International Conference on Artificial Intelligence and Statistics. PMLR, 2020.
[5] Lachapelle, Sébastien, et al. *Gradient-based neural dag learning.* arXiv preprint arXiv:1906.02226 (2019).

Table 4: The latent causal effect learning performances.

| Variable | Variable Reconstruction (initial performance) | | | Cause Node | Variable Reconstruction (as result node) | | | Latent Causal Effect Reconstruction | | | |
	RMSE on Scaled Values	RMSE on Original Values	BCE Mask		RMSE on Scaled Values	RMSE on Original Values	BCE Mask	RMSE on Scaled Values	RMSE on Original Values	BCE Mask	KLD (in latent space)
C	0.037	0.089	0.428	A	0.0295	0.0616	0.4278	0.1747	0.3334	0.4278	7.6353
D	0.015	0.679	0.445	BC	0.0350	1.0179	0.1355	0.0509	1.7059	0.1285	9.6502
				B	0.0341	1.0361	0.1693	0.0516	1.7737	0.1925	8.5147
				C	0.0331	0.9818	0.3404	0.0512	1.7265	0.3667	10.149
E	0.058	3.343	0.643	BC	0.4612	26.605	0.6427	0.7827	45.149	0.6427	39.750
				B	0.6428	37.076	0.6427	0.8209	47.353	0.6427	37.072
				C	0.5212	30.065	1.2854	0.7939	45.791	1.2854	46.587
F	0.326	7.178	2.045	E	0.4334	8.3807	3.0895	0.4509	5.9553	3.0895	53.680
G	0.045	0.81	1.327	CDE	0.0538	0.9598	0.0878	0.1719	3.5736	0.1340	8.1360
				C	0.1057	1.4219	0.1078	0.2996	4.6278	0.1362	11.601
				D	0.1773	3.6083	0.1842	0.4112	8.0841	0.2228	27.879
				E	0.1949	4.7124	0.1482	0.5564	10.852	0.1877	39.133
H	0.045	0.009	1.345	DE	0.0889	0.0099	2.5980	0.3564	0.0096	2.5980	21.905
				D	0.0878	0.0104	0.0911	0.4301	0.0095	0.0911	25.198
				E	0.1162	0.0105	0.1482	0.5168	0.0097	3.8514	39.886
I	0.035	0.009	1.672	DF	0.0600	0.0103	3.4493	0.1158	0.0099	3.4493	49.033
				D	0.1212	0.0108	3.0048	0.2073	0.0108	3.0048	75.577
				F	0.0540	0.0102	3.4493	0.0948	0.0098	3.4493	45.648
J	0.007	0.098	1.088	GHI	0.0052	0.0742	0.2593	0.0090	0.1269	0.2937	5.5300
				G	0.0077	0.1085	0.4009	0.0099	0.1390	0.4375	5.2924
				H	0.0159	0.2239	0.4584	0.0393	0.5520	0.4938	15.930
				I	0.0308	0.4328	0.3818	0.0397	0.5564	0.3954	17.410

[6] Luo, Yunan, Jian Peng, and Jianzhu Ma. *When causal inference meets deep learning.*" Nature Machine Intelligence 2.8 (2020): 426-427.

[7] Qayyum, Adnan, et al. *Secure and robust machine learning for healthcare: A survey.* IEEE Reviews in Biomedical Engineering 14 (2020): 156-180.

[8] Ahmad, Muhammad Aurangzeb, Carly Eckert, and Ankur Teredesai. *Interpretable machine learning in healthcare.* Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics. 2018.

[9] Chen, Irene Y., et al. *Ethical machine learning in healthcare.* Annual review of biomedical data science 4 (2021): 123-144.

[10] Sanchez, Pedro, et al. *Causal machine learning for healthcare and precision medicine.* Royal Society Open Science 9.8 (2022): 220638.

[11] Crown, William H. *Real-world evidence, causal inference, and machine learning.* Value in Health 22.5 (2019): 587-592.

[12] Lecca, Paola. *Machine learning for causal inference in biological networks: Perspectives of this challenge.* Frontiers in Bioinformatics 1 (2021): 746712.

[13] Sobel, Michael E. *An introduction to causal inference.* Sociological Methods & Research 24.3 (1996): 353-379.

[14] Glymour, Clark, Kun Zhang, and Peter Spirtes. *Review of causal discovery methods based on graphical models.* Frontiers in genetics 10 (2019): 524.

[15] Elwert, Felix. *Graphical causal models.* Handbook of causal analysis for social research (2013): 245-273.

[16] Li, Jia, et al. *Teaching deep learning causal effects improves predictive performance.* arXiv preprint arXiv:2011.05466 (2020).

[17] Schölkopf, Bernhard, et al. *Toward causal representation learning.* Proceedings of the IEEE 109.5 (2021): 612-634.

[18] Li, Yunzhu, et al. *Causal discovery in physical systems from videos.* Advances in Neural Information Processing Systems 33 (2020): 9180-9192.

[19] Bengio, Yoshua, et al. *Unsupervised feature learning and deep learning: A review and new perspectives.* CoRR, abs/1206.5538 1.2665 (2012): 2012.

[20] Baldi, Pierre, and Kurt Hornik. *Neural networks and principal component analysis: Learning from examples without local minima.* Neural networks 2.1 (1989): 53-58.

[21] Plaut, Elad. *From principal subspaces to principal components with linear autoencoders.* arXiv preprint arXiv:1804.10253 (2018).

[22] Wang, Yasi, Hongxun Yao, and Sicheng Zhao. *Auto-encoder based dimensionality reduction.* Neurocomputing 184 (2016): 232-242.

[23] Jain, Saachi, Adityanarayanan Radhakrishnan, and Caroline Uhler. A mechanism for producing aligned latent spaces with autoencoders. arXiv preprint arXiv:2106.15456 (2021).

[24] Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. *Glove: Global vectors for word representation.* Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014.

[25] Rong, Xin. *word2vec parameter learning explained.* arXiv preprint arXiv:1411.2738 (2014).

[26] Belyaeva, Anastasiya, et al. "Causal network models of SARS-CoV-2 expression and aging to identify candidates for drug repurposing." Nature communications 12.1 (2021): 1024.

[27] Lotfollahi, Mohammad, F. Alexander Wolf, and Fabian J. Theis. *scGen predicts single-cell perturbation responses.* Nature methods 16.8 (2019): 715-721.

[28] Goodwell, Allison E., et al. *Debates—Does information theory provide a new paradigm for Earth science? Causality, interaction, and feedback.* Water Resources Research 56.2 (2020): e2019WR024940.

[29] Kratzert, Frederik, et al. *Rainfall–runoff modeling using long short-term memory (LSTM) networks.* Hydrology and Earth System Sciences 22.11 (2018): 6005-6022.

[30] Kratzert, Frederik, et al. *Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets.* Hydrology and Earth System Sciences 23.12 (2019): 5089-5110.

[31] Li, Xiaojie, et al. *Sequence-to-Sequence Learning for Prediction of Soil Temperature and Moisture.* IEEE Geoscience and Remote Sensing Letters 19 (2022): 1-5.

[32] Reisach, Alexander G., Christof Seiler, and Sebastian Weichwald. *Beware of the simulated dag! varsortability in additive noise models.* arXiv preprint arXiv:2102.13647 (2021).

[33] Kaiser, Marcus, and Maksim Sipos. *Unsuitability of NOTEARS for causal graph discovery.* arXiv preprint arXiv:2104.05441 (2021).

[34] Zheng, Xun, et al. *Dags with no tears: Continuous optimization for structure learning.* Advances in Neural Information Processing Systems 31 (2018).

[35] Dinh, Laurent, Jascha Sohl-Dickstein, and Samy Bengio. *Density estimation using real nvp.* arXiv preprint arXiv:1605.08803 (2016).

Appendix A   COMPLETE EXPERIMENTAL RESULTS OF CAUSAL DISCOVERY

Table 1: The Complete Results of Heuristic Causal Discovery in latent space. Each row stands for a round of detection, with '#' identifying the round number, and all candidate edges are listed with their KLD gains as below. 1) Green cells: the newly detected edges. 2) Red cells: the selected edge. 3) Blue cells: the trimmed edges accordingly.

#	Candidate edges (KLD gains)
#1	A→C **7.6354**(R) · A→D 19.7407 · A→E 60.1876 · A→F 119.7730 · B→C **8.4753**(B) · B→D 8.5147 · B→E 65.9335 · B→F 132.7717 · C→F 111.2978 · C→G 11.6012 · C→H 39.2361 · C→I 95.1564 · D→F 123.3203 · D→G 27.8798 · D→H 25.1988 · D→I 75.5775(G)
#2	A→D **19.7407** · A→E 60.1876 · A→F 119.7730 · B→D 8.5147 · B→E 65.9335 · B→F 132.7717 · C→D 10.1490(G) · C→E 46.5876(G) · C→F 111.2978 · C→H 39.2361 · C→I 95.1564 · D→E 63.7348 · D→F 123.3203 · D→G 27.8798 · D→I 75.5775
#3	A→D **9.7357**(B) · A→E 60.1876 · A→F 119.7730 · B→E 65.9335 · B→F 132.7717 · C→D **1.1355**(R) · C→E 46.5876 · C→F 111.2978(G) · C→H 39.2361 · C→I 95.1564 · D→E 63.7348(G) · D→F 123.3203(G) · D→G 27.8798 · D→H 25.1988 · D→I 75.5775
#4	A→E 60.1876 · A→F 119.7730 · B→E 65.9335 · B→F 132.7717 · C→E 46.5876 · C→G 63.7348 · B→E **11.6012**(R) · C→H 39.2361 · C→I 95.1564 · D→E 63.7348 · D→F 123.3203 · D→G 27.8798 · D→H 25.1988 · D→I 75.5775
#5	A→E 60.1876 · A→F 119.7730 · B→E 65.9335 · B→F 132.7717 · C→E 46.5876 · C→H 39.2361 · C→I 95.1564 · D→E 63.7348 · D→F 123.3203 · D→G **2.4540**(R) · D→H 25.1988 · D→I 75.5775 · G→J 5.2924(G)
#6	A→E 60.1876 · A→F 119.7730 · B→E 65.9335 · B→F 132.7717 · C→E 46.5876 · C→H 39.2361 · C→I 95.1564 · D→E 63.7348 · D→F 63.7348 · D→H 25.1988 · D→I 75.5775 · G→J **5.2924**(R)
#7	A→E 60.1876 · A→F 119.7730 · B→E 65.9335 · B→F 132.7717 · C→E 46.5876 · C→H **39.2361**(B) · C→I 95.1564 · D→E 63.7348 · D→F 63.7348 · D→H **25.1988**(R) · D→I 75.5775
#8	A→E 60.1876 · A→F 119.7730 · B→E 65.9335 · B→F 132.7717 · C→E 46.5876 · C→F 95.1564 · C→I 95.1564 · D→E 63.7348 · D→F 123.3203 · D→I 75.5775 · H→J **0.2092**(R)
#9	A→E **60.1876**(B) · A→F 119.7730 · B→E 65.9335 · B→F 132.7717 · C→E **46.5876**(R) · C→I 95.1564 · D→E 63.7348 · D→F 123.3203 · D→I 75.5775
#10	A→F 119.7730 · B→E **-6.8372**(R) · B→F 123.3203 · C→F 111.2978 · C→I 95.1564 · D→E **17.0407**(B) · D→F 123.3203 · E→F 75.5775 · E→G 53.6806(G) · E→H -5.9191 · E→I 110.2558
#11	A→F 119.7730 · B→F 132.7717 · C→F 111.2978 · C→I 95.1564 · D→F 123.3203 · D→I 75.5775 · E→F 53.6806 · E→G **-5.9191**(G) · E→H **-3.2931**(R) · E→I 110.2558
#12	A→F 119.7730 · B→F 132.7717 · C→F 111.2978 · C→I 95.1564 · D→F 123.3203 · D→I 75.5775 · E→F 53.6806 · E→H **-3.2931**(B) · E→I 110.2558
#13	A→F **119.7730**(R) · B→F **132.7717**(B) · C→F **123.3203**(B) · D→F **123.3203** · E→F **53.6806**(B) · E→I 110.2558
#14	C→I 95.1564 · D→I 75.5775 · E→I **110.2558**(B) · F→I **45.6490**(R)
#15	C→I 95.1564 · D→I 75.5775 · E→I 110.2558 · I→J **0.0284**(R)
#16	C→I **15.0222**(R,B) · D→I **3.3845**(R)

Table 2: Average performance of 10-Fold FGES (Fast Greedy Equivalence Search) causal discovery, with the prior knowledge that each node can only cause the other nodes with the same or greater depth with it. An edge means connecting two attributes from two different nodes, respectively. Thus, the number of possible edges between two nodes is the multiplication of the numbers of their attributes, i.e., the lengths of their data vectors.
(All experiments are performed with 6 different Independent-Test kernels, including chi-square-test, d-sep-test, prob-test, disc-bic-test, fisher-z-test, mvplr-test. But their results turn out to be identical.)

Cause Node	A	B		C			D			E			F	G	H	I
True Causation	A→C	B→D	B→E	C→D	C→E	C→G	D→G	D→H	D→I	E→F	E→G	E→H	F→I	G→J	H→J	I→J
Number of Edges	16	24	16	6	4	8	12	12	9	8	8	8	12	4	4	3
Probability of Missing	0.038889	0.125	0.125	0.062	0.06875	0.039286	0.069048	0.2	0.142857	0.3	0.003571	0.2	0.142857	0.0	0.072727	0.030303

Wrong Causation	C→F	D→E	D→F	F→G	G→H	G→I	H→I
Times of Wrongly Discovered	5.6	1.2	0.8	5.0	8.2	3.0	2.8

Table 3: Brief Results of the Heuristic Causal Discovery in latent space, identical with Table 3 in the paper body, for better comparison to the traditional FGES methods results on this page.
The edges are arranged in detected order (from left to right) and their measured causal strengths in each step are shown below correspondingly.
Causal strength is measured by KLD values (less is stronger). Each round of detection is pursuing the least KLD gain globally.
All evaluations are in 4-Fold validation average values. Different colors represent the ground truth causality strength tiers (referred to the Figure 10 in the paper body).

Causation	A→C	B→D	C→D	C→G	D→G	G→J	D→H	H→J	C→E	B→E	E→G	E→H	E→F	F→I	I→J	D→I
KLD	7.63	8.51	10.14	11.60	27.87	5.29	25.19	15.93	46.58	65.93	39.13	39.88	53.68	45.64	17.41	75.57
Gain	7.63	8.51	1.135	11.60	2.454	5.29	25.19	0.209	46.58	-6.84	-5.91	-3.29	53.68	45.64	0.028	3.384