# Causal Sensitivity Identification using Generative Learning

**Soma Bandyopadhyay[1]*, Sudeshna Sarkar[2]**
**[1]TCS Research, TATA Consultancy Services Limited, Kolkata, India**
**soma.bandyopadhyay@tcs.com**,
**[2]Department of Computer Science and Engineering, IIT Kharagpur, India**
**sudeshna@cse.iitkgp.ac.in**

## Abstract

In this work, we propose a novel generative method to identify the causal impact and apply it to prediction tasks. We conduct causal impact analysis using interventional and counterfactual perspectives. First, applying interventions, we identify features that have a causal influence on the predicted outcome, which we refer to as causally sensitive features, and second, applying counterfactuals, we evaluate how changes in the cause affect the effect. Our method exploits the Conditional Variational Autoencoder (CVAE) to identify the causal impact and serve as a generative predictor. We are able to reduce confounding bias by identifying causally sensitive features. We demonstrate the effectiveness of our method by recommending the most likely locations a user will visit next in their spatiotemporal trajectory influenced by the causal relationships among various features. Experiments on the large-scale GeoLife [Zheng *et al.*, 2010] dataset and the benchmark Asia Bayesian network validate the ability of our method to identify causal impact and improve predictive performance.

## 1 Introduction

Determining causal impact is an important need of causal reasoning to understand the causal path among different features.

In this work, we aim to identify the causal impact in a prediction task through interventional and counterfactual analysis without assuming any prior causal graph, which we name **causal sensitivity identification**. We define the objectives of causal sensitivity identification as follows:

1. Identifying features that have a causal influence on the prediction outcome, and play a key role in preserving causal relationships [Bandyopadhyay and Sarkar, 2023], we refer to these features as **causally sensitive** features.
2. Assessing the impact of changes in causes on their effects.
3. Identifying causal paths (e.g., $X \rightarrow Y$) between input features and the prediction target, without assuming any prior knowledge of the causal graph.

We propose a **generative method** for causal impact analysis using Conditional Variational Autoencoder (CVAE) [Doersch, 2021] as generative predictor.

Causality operates on three main levels of the ladder of causation [Pearl and Mackenzie, 2018]: association, intervention, and counterfactuals. The lowest level captures statistical associations without causal interpretation. The middle level involves interventions, modeled through the do-calculus, which assigns values randomly to a variable, establishes the direct causal relation between cause and effect, and reduces confounding bias. The highest level represents the counterfactual [Kment, 2020], revealing cause and effect relationships and their dependence on counterfactuals [Pearl, 2019] or alternate situations. For example, could different past location sequences result in the same next location? Does a user's location history influence their next movement?

- **Intervention: Identifying causally sensitive feature**
  1. We determine the **causally sensitive** features ($F_{CS}$) that influence both the cause and the effect, potentially acting as confounders. To detect $F_{CS}$, we compare prediction performance on identical test data using two models, one trained on the original (factual) train data and the other trained on intervened data where candidate features are altered.
  2. Identified causally sensitive features are used to **condition** the generative prediction model to ensure the prediction is guided by a true causal relations.
- **Counterfactuals: Assessing impact of cause changes**
  1. We assess the change in effect when the cause has changed using counterfactual analysis and identify the **causal path**. In this scenario, we compare the performance of prediction using counterfactual test data obtaining the counterfactual latent representation, and **original (factual)** test data using the predictor trained in original (factual) train data.
  2. We obtain the counterfactual predictions / generations to determine outcomes in alternate situations.

A summary of our contribution is as follows:

*Contact Author

1. **Causal sensitivity identification framework**:
   - We propose a new method to determine causally sensitive features using interventional analysis for prediction tasks, exploiting a CVAE based generative predictor.
   - We identify causal sensitivity by assessing the impact of changes in causes on their effects using counterfactuals, and use this to identify the underlying causal path.
2. **Causally sensitive recommendation/prediction**:
   - We evaluate our method on the **Asia** Bayesian network from the BNLearn repository [Scutari, 2009], demonstrating its ability to recover known causal dependencies by identifying causally sensitive features and associated causal paths through interventions and counterfactual analysis.
   - We apply the proposed framework to the real-world **GeoLife** [Zheng *et al.*, 2010] human trajectory dataset, where the identified causally sensitive features (e.g., `start_time`) are used to condition next location prediction. We further assess the impact of counterfactual changes in past location sequences on future trajectory predictions.

## 2 Related Work

We explore the related works on causality focusing on applications of intervention and counterfactuals using neural models and on the trajectory predictions.

**Causal interventions and counterfactuals:** The use of interventions and counterfactuals to uncover cause-effect relationships has emerged as a key area of research, particularly under partially known or unknown causal structures.

- Yang et al. [Yang *et al.*, 2018] propose I-MEC to characterize interventional equivalence classes and reduce ambiguity via soft interventions.
- Ke et al. [Ke *et al.*, 2020] develop a neural approach for causal path discovery under unknown interventions without requiring explicit targets.
- Dai et al. [Dai *et al.*, 2025] address causal discovery under selection bias and latent interventions.
- Kung et al. [Zuo *et al.*, 2022] ensure counterfactual stability by assuming sensitive attributes lack ancestors, enforcing prediction consistency.
- `Neural generative models:`
  - VAE [Kingma and Welling, 2013] based generative models have been widely adopted for counterfactual reasoning and disentangled causal representation learning.
    - Louizos et al. [Louizos *et al.*, 2017] propose CEVAE, which models noisy proxy variables for unobserved confounders based on Pearl's backdoor criterion [Pearl, 2009].
    - Yang et al. [Yang *et al.*, 2021b] propose CausalVAE, which integrates a linear structural causal model (SCM) with a VAE for counterfactual generation using known causal structures.
  - Kuang et al. [Xia *et al.*, 2023] apply generative adversarial networks (GANs) [Goodfellow *et al.*, 2014] for counterfactual inference under known causal graphs.
- `Causal structure learning:` Numerous methods aim to recover causal structures from data, often by optimizing a score under acyclicity constraints. Among these, NOTEARS [Zheng *et al.*, 2018] is a widely used method that formulates DAG discovery as a continuous and differentiable optimization problem. It is commonly applied to identify pairwise causal relations and can complement intervention-based approaches. Ke et al. [Ke *et al.*, 2020] focus on causal discovery under unknown intervention targets, aiming to recover the DAG without explicitly addressing prediction performance or counterfactual inference.

**Sequence modeling for trajectory prediction:** Trajectory prediction based on GPS data is widely studied as a sequence modeling task, especially in mobility applications where past behavior influences future outcomes.

- LSTM-based models, often combined with attention mechanisms [Vaswani *et al.*, 2017; Yang *et al.*, 2021c], effectively capture spatiotemporal dependencies in trajectory prediction.
- Deep learning has been widely applied to trajectory modeling [Wang *et al.*, 2022], with surveys and benchmarks in [Rudenko *et al.*, 2020; Nezhadettehad *et al.*, 2024].
- Generative models such as VAEs [Salzmann *et al.*, 2020] support distributional forecasting in frameworks like [Feng *et al.*, 2020; Liu and others, 2021], enabling uncertainty-aware predictions.
- **Our emphasis:** We emphasize the following related works to compare our method against theirs on human trajectory prediction using GPS trajectory data.
  - **LSTM**: The authors [Krishna *et al.*, 2018] show that LSTM-based models outperform HMM and hierarchical HMM baselines [MacDonald and Zucchini, 1997] for activity recognition and duration estimation.
  - **LSTM + Attention**: The authors [Li *et al.*, 2020] introduce hierarchical attention mechanisms to improve long-term mobility pattern prediction using LSTM architectures.
  - **DeepMove**: [Feng *et al.*, 2018] jointly embeds multimodal factors such as time, user ID, and location, enhanced with a historical attention module to improve location transition modeling.
  - **MHSA**: [Hong *et al.*, 2023] proposes a multi-head self-attention (MHSA) model that leverages spatiotemporal features (e.g., visit time, duration) for location sequence modeling.

Despite their success in sequence learning, these models largely ignore causal reasoning and do not consider how interventions or counterfactual variations in inputs affect trajectory predictions.

## 3 Methodology

We present the proposed causal sensitivity identification method using a CVAE-based generative model as a generative predictor. Our framework incorporates interventional and counterfactual analysis expanding upon the objectives stated in the introduction.

As illustrated in Figure 1, the functional components of the proposed method comprises the generative predictor, with the factual path shown is blue solid line, the interventional path shown in red dotted line, counterfactuals path shown in green dotted line along with the counterfactual latent representation $Z_{CF}$, and the input data $D(X, Y)$.

The proposed framework integrates a generative predictor based on CVAE to evaluate causal sensitivity through three complementary paths: factual, interventional, and counterfactual. The factual path assesses the model's baseline predictive performance on unaltered data. The interventional path enables the identification of causally sensitive features by measuring performance changes when specific variables are intervened upon (e.g., $do(X_i = x')$). Finally, the counterfactual path estimates the effect of hypothetical changes in the input by generating counterfactual outcomes using the factual latent representations and altered test instances. Together, these components allow for a systematic analysis of causal influence and sensitivity, providing insight into which variables are most critical for accurate prediction and robust decision-making.

The details of the proposed method are described below.

*Notation:* $X_{\text{train}}$ and $X_{\text{test}}$ denote the factual (unaltered) train and test data, respectively. All equations using $Y_{t+1}$ apply analogously for $Y$ in non-sequential settings.

### 3.1 Generative Model

We use Conditional Variational Autoencoder (CVAE) as a generative predictor (GP) with an encoder and decoder components. We use sparse-categorical cross-entropy loss ($\mathcal{L}_{\text{rec}}$) (depicted in Equation (1)) which represents the negative log probability of the Y, as true label in the training data given the input features (X), averaged over all samples (N) for sequence prediction tasks.

$y_i$ represents the ith true level of the $x_i \in X$ sequence up to time step $t-1$: $y_i = f(X_{1:t-1})$ ; (for non-sequential data, $y_i$ simply corresponds to the label for $\mathbf{x}_i$).

$$\mathcal{L}_{\text{rec}} = -\frac{1}{N}\sum_{i=1}^{N} \log p_\theta(y_i \mid \mathbf{x}_i); y_i \epsilon Y; x_i \epsilon X \tag{1}$$

For prediction of binary-valued targets, we use binary cross-entropy loss $\mathcal{L}_{\text{bce}}$ (depicted in Equation (2))

$$\mathcal{L}_{\text{bce}} = -\frac{1}{N}\sum_{i=1}^{N} \left[y_i \log p_\theta(y_i \mid \mathbf{x}_i) + (1-y_i)\log(1-p_\theta(y_i \mid \mathbf{x}_i))\right] \tag{2}$$

$$c_{\text{max}} = \max(Y_{\text{train}}) + 1 \tag{3}$$

**Encoder:** The encoder learns the latent space representation (Z). Z is obtained by computing the mean $\mu$ and log-variance $\sigma$ with the reparameterization trick to sample from the latent space, first $\epsilon$ is sampled from $N(0, 1)$ and then z is computed as $z = \mu(Y|X) + \sigma^{1/2}(Y|X) * \epsilon$.

**Decoder:** The decoder takes Z as input and repeats across the time step which is the maximum sequence length in the training data and combines this with the conditional input X, considering X as temporal data. Now this Z conditioned on X is passed to the next neural layers as used in Encoder. The final output of the decoder goes to a dense layer. This final dense layer has $c_{max}$ (defined in Equation (3)) number of nodes with softmax activation to predict the next location.

**Generator:** The decoder model is employed to generate new data samples of Y by passing the latent samples sampled from a Gaussian distribution and using conditional inputs. For **sequence prediction tasks** such as next-location modeling, we use the following CVAE loss (Equation 4). The reconstruction loss uses sparse categorical cross-entropy (Equation 1. We apply **KL annealing** [Li *et al.*, 2019], kl_weight (a gradually increasing weight) to multiply the KL divergence term to counter KL-vanishing during the initial training.

$$\begin{aligned}\mathcal{L}(\theta, \phi; \mathbf{Y}_t, \mathbf{X}_{t-1:t-n}) = \\ -\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{Y}_t, \mathbf{X}_{t-1:t-n})}\left[\log p_\theta(\mathbf{Y}_t|\mathbf{z}, \mathbf{X}_{t-1:t-n})\right] \\ +\text{kl_weight} \cdot \text{KL}\left(q_\phi(\mathbf{z}|\mathbf{Y}_t, \mathbf{X}_{t-1:t-n}) \parallel p_\theta(\mathbf{z}|\mathbf{X}_{t-1:t-n})\right)\end{aligned} \tag{4}$$

$\theta$, $\phi$ are the parameters of the decoder, and encoder network respectively. $\mathbf{Y}_t$ is the target output $\mathbf{X}_{t-1:t-n}$ is the input sequence comprising different features. $q_\phi(\mathbf{z}|\mathbf{Y}_t, \mathbf{X}_{t-1:t-n})$ is the approximate posterior distribution, $p_\theta(\mathbf{Y}_t|\mathbf{z}, \mathbf{X}_{t-1:t-n})$ is the likelihood of the data given the latent variable and the conditional input. $p_\theta(\mathbf{z}|\mathbf{X}_{t-1:t-n})$ is the prior distribution of the latent variable given the conditional input. KL denotes the Kullback-Leibler divergence [Hershey and Olsen, 2007]. For non-sequential prediction tasks the same loss Eq. (4) is used, where the reconstruction term follows binary cross-entropy (Eq. (2)), with $y_i$ corresponding directly to the label for $\mathbf{x}_i$.

### 3.2 Causal Sensitivity Identification

The proposed method determines the causal sensitivity considering two perspectives.

1. **Causally sensitive feature ($Z_{CF}$) and reduction of confounding bias**: We propose the following steps to determine the causal sensitivity of a feature without considering any prior knowledge of causal graph. We aim to reduce the confounding effect by blocking the backdoor path (depicted in Figure 2) which connects X and Y with at least one common ancestor or common cause $X \leftarrow Z \rightarrow Y$, where X is not the cause of Z.
   (a) **Factual training**: Train the generative predictor (GP) model using the factual input $X_{\text{train}}$ to create GP-F (Baseline) as shown in Equation 5.
   (b) **Intervention**: Apply the intervention, ($do(X = X_{\text{altered}})$). Train the GP using $X_{\text{train_Altered}}$ to obtain GP-I.

$$P(Y_{t+1} \mid (X_t)) = \sum_{F_{\text{CS}}} P(Y_{t+1} \mid X_t) \tag{5}$$

$$P(Y_{t+1} \mid \text{do}(X = X_{\text{altered}})) = \sum_{F_{\text{CS}}} P(Y_{t+1} \mid X_{\text{altered}}, F_{\text{CS}}) P(F_{\text{CS}}) \tag{6}$$

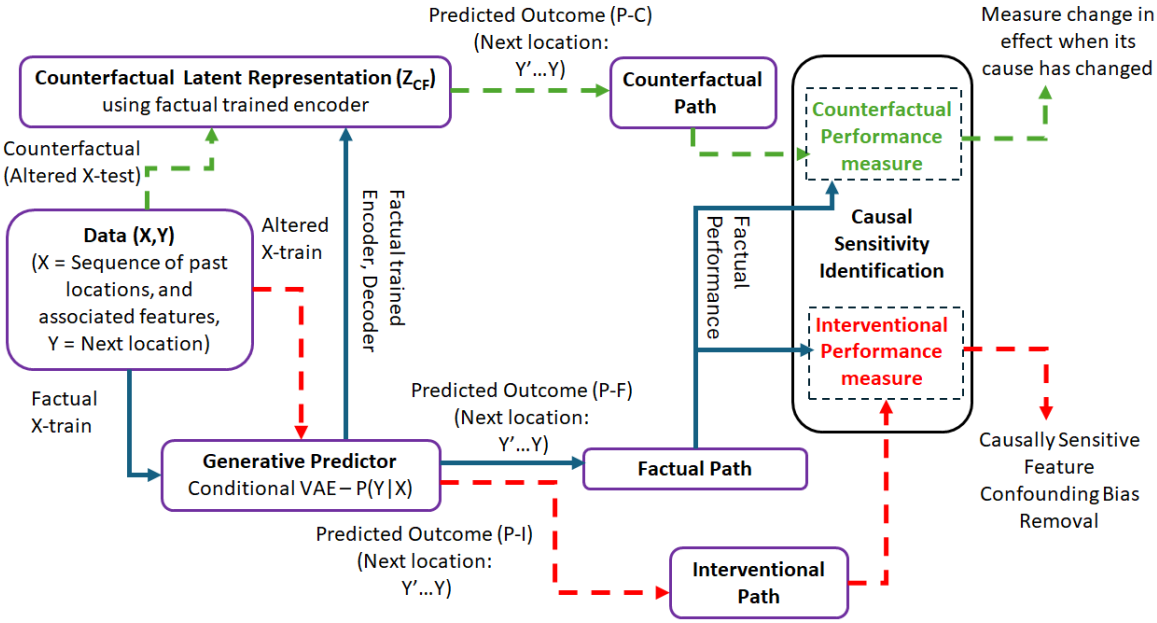



Figure 1: Functional components of the proposed causal sensitivity identification framework with the factual (blue), interventional (red), and counterfactual (green) to evaluate causal influence in prediction tasks.

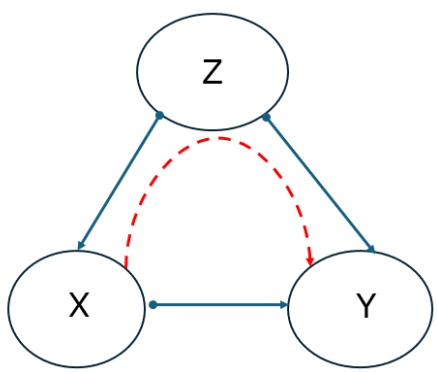



Figure 2: Causal graph depicting backdoor path

(c) **Factual prediction**: Test the GP-F using $X_{\text{test}}$.

$$Y_{(t+1)\text{factual}} = \text{GP-F.decoder}(Z_{FC}, X_{\text{test}}) \quad (7)$$

(d) **Interventional prediction**: Test the GP-I using $X_{\text{test}}$.

$$Y_{(t+1)\text{Interventional}} = \text{GP-I.decoder}(Z_{IF}, X_{\text{test}}) \quad (8)$$

(e) **Features having causal influence:** If the prediction error is higher (i.e., accuracy is lower) in the factual scenario (c) than in the interventional scenario (d), this indicates, that the feature is causally sensitive and acts as a common influencer, as blocking of backdoor path enables the direct causal path between X and Y and reduces the confounding bias in the causal effect of X on Y. We present this in terms of the difference in prediction accuracy, $\Delta$Acc. (Examples Acc: Acc@1, MRR $\in$ Acc defined in the section 4.2 Performance Measure). The difference in accuracy is defined as:

$$\Delta\text{Acc} = \text{Acc}_{\text{interventional}} - \text{Acc}_{\text{factual}}; \Delta\text{Acc} > 0 \quad (9)$$

2. **Counterfactuals to measure the change in effect when the cause has changed**

(a) **Obtain factual latent representation, and GP-F**: Get $\mathbf{z_f}$: Train the GP using factual $X_{\text{train}}$, $Y_{\text{train}}$.

$$\mathbf{z_f} \sim q_\phi(Y_t, \mathbf{X}_{t-1:t-n})_factual. \quad (10)$$

The trained encoder of GP-F is used to obtain $Z_{\text{FC}}$, the factual latent representation from $X_{\text{test}}$

$$Z_{\text{FC}} = \text{GP-F.encoder}(X_{\text{test}})$$

(b) **Obtain counterfactual latent representation $Z_{\text{CF}}$:** $Z_{\text{CF}}$ follows counterfactual probability which is computed based on the equation (11)

$$P(Y_{X=x'} \mid X = x, Y = y) \approx \int P(Y \mid X = x', \mathbf{z})\, q_\phi(\mathbf{z} \mid X = x, Y = y)\, d\mathbf{z} \quad (11)$$

Obtain the counterfactual latent representation from the $X_{test_altered}$.

$$Z_{\text{CF}} = \text{GP-F.encoder}(X_{\textbf{test_altered}}) \quad (12)$$

$Z_{\text{CF}}$ is the counterfactual latent representation.

(c) **Generate factual and counterfactual predictions:** Use the decoder of GP-F to predict the outcome $Y_{\text{t+1}}$ for the factual scenario:

$$Y_{(t+1)\text{factual}} = \text{GP-F.decoder}(Z_{FC}, X_{\text{test}}) \quad (13)$$

$$Y_{(t+1)\text{counterfactual}} = \text{GP-F.decoder}(Z_{CF}, X_{\textbf{test_altered}}) \quad (14)$$

We use counterfactuals as described above on the test data and also generate counterfactuals $Y_{(t+1)\text{counterfactual}}$. The difference in accuracy between counterfactual and factual scenarios $\Delta\text{Acc} < 0 (\Delta\text{Acc} = \text{Acc}_{\text{counterfactual}} - \text{Acc}_{\text{factual}})$, signifies causal path $X \rightarrow Y$. The proposed generative causal sensitivity identification method is presented in Algorithm 1, without using any prior knowledge of causal graph and applying any causality constraints during learning.

**Algorithm 1** Causal Sensitivity Identification Method

**Input**: $X_{\text{train}}, Y_{\text{train}}, X_{\text{train_altered}}, X_{\text{test}}, X_{\text{test_altered}}$
**Output**: $\Delta$Acc , Causally sensitive features, Causal path, Counterfactual prediction

Train GP-F on $X_{\text{train}}, Y_{\text{train}}$ to learn encoder $q_\phi$ and decoder $p_\theta \rightarrow$ Eq. (4)
Train GP-I on $X_{\text{train_altered}} \rightarrow$ Eq. (6)
Compute factual latent:
$Z_{\text{FC}} = \text{GP-F.encoder}(X_{\text{test}})$
Compute interventional latent:
$Z_{\text{IF}} = \text{GP-I.encoder}(X_{\text{test}})$
Factual prediction:
$Y_{(t+1)\text{factual}} = \text{GP-F.decoder}(Z_{\text{FC}}, X_{\text{test}}) \rightarrow$ Eq. (13)
Interventional prediction:
$Y_{(t+1)\text{interventional}} = \text{GP-I.decoder}(Z_{\text{IF}}, X_{\text{test}}) \rightarrow$ Eq. (8)
$\Delta\text{Acc} = \text{Acc}_{\text{interventional}} - \text{Acc}_{\text{factual}} \rightarrow$ Eq. (9)
**if** $\Delta\text{Acc} > 0$ **then**
    Feature is causally sensitive
**end if**
Counterfactual latent: $Z_{\text{CF}} = \text{GP-F.encoder}(X_{\text{test_altered}}) \rightarrow$ Eq. (12)
Counterfactual prediction: $Y_{(t+1)\text{counterfactual}} = \text{GP-F.decoder}(Z_{\text{CF}}, X_{\text{test_altered}}) \rightarrow$ Eq. (14)
$\Delta\text{Acc} = \text{Acc}_{\text{counterfactual}} - \text{Acc}_{\text{factual}} \rightarrow$ Eq. (11)
**if** $\Delta\text{Acc} < 0$ **then**
    Infer causal path: $X \rightarrow Y$
**end if**

### 3.3 Causally sensitive recommendation/prediction:

The identified $F_{CS}$ (Algorithm 1) is applied to condition the prediction task.We refer to this as generative causally sensitive prediction (**GCSP**) presented in Algorithm 2.

Our method integrates causal sensitivity identification with generative prediction and is applicable to both general and sequential prediction tasks, such as next-location prediction. In contrast to prior works, it addresses causal impact analysis with the following key features:

- Identification of causally sensitive features through interventional analysis and quantification of their influence using a generative predictor;
- Assessment of the impact of changes in causes on predicted outcome, enabling identification of causal path;
- A unified prediction framework that operates without prior knowledge of the causal graph or structural constraints (such as acyclicity, as required in methods like NOTEARS [Zheng *et al.*, 2018]).

**Algorithm 2** Generative Causally Sensitive Prediction (GCSP)

**Input:** Factual training data $(X_{\text{train}}, Y_{\text{train}})$, test data $X_{\text{test}}$
**Output:** Prediction $Y_{(t+1)\text{factual}}$ using causally sensitive conditioning

Train GP-F (Generative Predictor - Factual) using CVAE on $(X_{\text{train}}, Y_{\text{train}})$
Identify causally sensitive features $F_{CS}$ via intervention analysis (Algorithm 1)
Condition the model on $F_{CS}$
Encode test data to obtain factual latent representation:

$$Z_{\text{FC}} = \text{GP-F.encoder}(X_{\text{test}})$$

Generate predictions using decoder conditioned on $F_{CS}$:

$$Y_{(t+1)\text{factual}} = \text{GP-F.decoder}(Z_{\text{FC}}, X_{\text{test}}) following Eq. (13)$$

**return** $Y_{(t+1)\text{factual}}$ as the causally conditioned next prediction

## 4 Evaluation of Proposed Method

In this section, we demonstrate causal sensitivity identification using the proposed method on the **Asia** dataset [Lauritzen and Spiegelhalter, 1988] and on the **GeoLife** [Zheng *et al.*, 2010] data.

It is important to note that the underlying causal structure for the Asia dataset is known, which allows for explicit validation of the causal paths identified by our method. In contrast, the GeoLife dataset does not have a ground-truth causal graph available for validation.

### 4.1 Causal Sensitivity Identification on The Asia Dataset

To demonstrate the proposed causal sensitivity identification method we first apply it to the **Asia** dataset.

**Data**: Asia real world data set of Bayesian Network Repository (BnLearn) [Scutari, 2009] contains 8 binary variables (e.g., `smoke`, `lung`, `bronc`, `dysp`). The directed acyclic graph (DAG) structure of the Asia data is predefined and presents known causal relationships among the variables, and is widely used in causal learning and inference tasks.

**Causal sensitivity analysis:** We focus our causal sensitivity analysis on predicting the target variable `dysp` (shortness of breath) while evaluating how conditioning on subsets of features and intervening on variables like `either` affects predictive accuracy. Our method is used to identify minimal sufficient sets, isolate confounders such as `smoke`, and study performance shifts under interventions like `do(either=1)`. We implement a CVAE to model the distribution of the binary target variable `dysp` conditioned on various subsets of input variables. Generative predictor CVAE is trained using binary cross-entropy loss combined Eq. (2) with a KL divergence regularization term. The encoder takes as input the conditioning features (e.g., `either`, `smoke`, `bronc`) and the target `dysp`, and outputs the latent mean and log-variance. The decoder reconstructs **dysp** from samples drawn from the latent space and the same conditioning input.

We use a multi-layer perceptron (MLP) based encoder with 16 hidden units to map the concatenated input of the target variable and conditioning variables ($[target, conditioning features]$) map to a latent space of dimension 2. The model is trained for 400 epochs using the Adam optimizer with a learning rate of $10^{-3}$, optimizing the binary cross-entropy reconstruction loss along with a KL divergence regularization term.

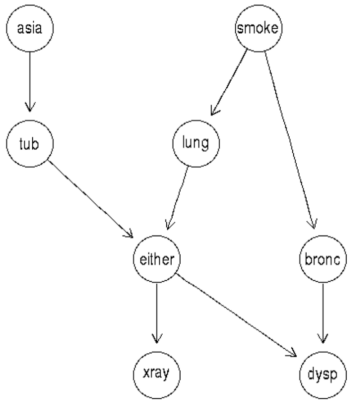



Figure 3: Causal graph - Asia [Ke *et al.*, 2020]

Table 1: Asia: Identification of causally sensitive features for `dysp` under factual and interventional (Intv) scenarios using the proposed method.

| Conditioning Set | Scenario | Accuracy |
|---|---|---|
| [either] | Factual | 0.660 |
| [either] | Intv | 0.580 |
| [either, bronc] | Factual | 0.815 |
| [either, bronc] | Intv | 0.850 |
| [either, bronc, lung] | Factual | 0.825 |
| [either, bronc, lung] | Intv | 0.845 |
| [either, bronc, lung, tub] | Factual | 0.835 |
| [either, bronc, lung, tub] | Intv | 0.845 |
| [either, smoke, bronc] | Factual | 0.850 |
| [either, smoke, bronc] | Intv | **0.855** |
| [either, smoke, bronc, tub] | Factual | 0.850 |
| [either, smoke, bronc, tub] | Intv | 0.850 |
| [either, smoke, bronc, lung] | Factual | 0.845 |
| [either, smoke, bronc, lung] | Intv | 0.850 |
| [either, smoke, bronc, lung, tub] | Factual | 0.850 |
| [either, smoke, bronc, lung, tub] | Intv | 0.850 |

Table 2: Asia: Counterfactual (CF) sensitivity analysis for `dysp` using the proposed method.

| Counterfactual | Factual Acc. | CF Acc. | Delta |
|---|---|---|---|
| either | 0.83 | 0.59 | **–0.240** |
| smoke | 0.83 | 0.84 | +0.010 |
| bronc | 0.83 | 0.365 | **–0.465** |
| lung | 0.83 | 0.85 | +0.020 |
| tub | 0.83 | 0.855 | +0.025 |

Based on the proposed method (Algorithm 1), we evaluate the causal sensitivity of the features on the prediction of the target variable `dysp` using the Asia dataset presented in Table 1. Our results indicate that `bronc` and `smoke` are causally sensitive variables for `dysp`, as there is a significant improvement in the accuracy of interventional scenarios. Furthermore, conditioning `smoke` with `bronc` while intervening do(either = 1) gives the highest performance indicates the possibility of `smoke` being a confounder, contributing to backdoor paths like `smoke` $\rightarrow$ `bronc` $\rightarrow$ `dysp` and `smoke` $\rightarrow$ `lung` $\rightarrow$ `either` $\rightarrow$ `dysp`. The ground truth causal graph is depicted in figure 3.

In Table 2 we present our findings highlighting the utility of counterfactual inference in uncovering both direct and indirect causal relationships without prior knowledge of the underlying graph. Specifically, intervening on `bronc` and `either` resulted in a decline in prediction accuracy by $-0.465$ and $-0.240$, respectively, thereby validating the existence of direct causal paths: `bronc` $\rightarrow$ `dysp` and `either` $\rightarrow$ `dysp`. We compare our method with prior approaches such as Ke et al. [Ke *et al.*, 2020], who apply a neural causal model to the Asia dataset and successfully identify key paths. Their method achieves high structural accuracy without requiring knowledge of intervention targets. However, their method focuses primarily on recovering the causal graph structure and does not explicitly quantify the effect of individual variables on prediction outcomes.

In contrast our generative approach identifies causally sensitive features and their effects on the prediction of `dysp` through performance deviations under factual, interventional, and counterfactual scenarios, and identifies the direct causal path, validating its effectiveness in capturing structural and functional causal dependencies.

We further compare our method against CausalVAE [Yang *et al.*, 2021a].

Table 3: Asia: counterfactual sensitivity analysis for `dysp` applying CausalVAE.

| Counterfactual | Factual Acc. | CF Acc. | Delta |
|---|---|---|---|
| tub | 0.62 | 0.62 | 0.000 |
| smoke | 0.62 | 0.62 | 0.000 |
| lung | 0.62 | 0.62 | 0.000 |
| bronc | 0.62 | 0.62 | 0.000 |
| either | 0.62 | 0.62 | 0.000 |

Table 3 presents the counterfactual evaluation of CausalVAE on the Asia dataset for predicting `dysp`. Notably, the model shows no significant variation in accuracy across counterfactual scenarios, indicating a lack of sensitivity to causal structure. This contrasts with our method (Table 2), which identifies `bronc` and `either` as causally sensitive features. This supports the claim that our approach better captures the underlying causal relationships necessary for meaningful counterfactual reasoning.

## 4.2 Causal Sensitivity Identification on The GeoLife Data

We apply the proposed method to predict the next location of human trajectory using GeoLife [Zheng *et al.*, 2010] data.

**Data:** GeoLife is a human trajectory dataset collected by 182 users in a period of over three years (from April 2007 to

August 2012) under the GeoLife project, Microsoft Research Asia. This comprises the GPS trajectory a sequence of time-stamped points, each of which contains the information of latitude, longitude and altitude having diverse sampling rate. This dataset has 17,621 trajectories covering a total distance of 1.2 million kilometers and more than 48,000 hours of duration. This trajectory dataset includes a wide range of users with diverse outdoor movements, like shopping, sightseeing, dining etc., along with their life routines like go home and go to work.

**Factual data:** The unaltered GeoLife data is exploited for factual analysis.

**LS:** The sequence of past location visits.

**Altered data:** We create altered versions of the data by modifying the sequence of location visits to conduct intervention and counterfactual analysis, as follows:

$LS_1$**:** Replace the most frequently visited location ID with the third most frequent.

$LS_1$**:** Replace the most frequently visited location ID with location ID 0.

We follow these steps to exploit the proposed method and perform experimental analysis:

1. Preprocessing of data
2. High level feature extraction
3. Evaluate causal sensitivity identification using interventions, counterfactuals
4. Next location prediction in GPS trajectory.

### Preprocessing of Data

In this step trajectories are processed to extract staypoints and locations. Staypoints are a subset of trajectories where the user stays for a minimum duration of time. We follow [Martin *et al.*, 2022] for the preprocessing to form locations. We use the preprocessed data to identify the causal-sensitivity and there after predicting the next location in the trajectory.

### High Level Feature Extraction

We use the open source Python library Trackintel [Martin *et al.*, 2022] to process and analyze the GeoLife movement data as considered by authors in [Hong *et al.*, 2023] and extract the various high-level mobility features. The high-level features considered are as follows:

Unique user identifier(UID), sequence of past location visits(LS), activity duration(DS), start minute (Smin), day of the week(W). The Smin feature adds a finer level of temporal granularity by indicating the specific start times of activity (location visit) within an hour or day. Feature W adds the perspective of the user's daily life visits and the other out-of-routine location visits.

### Performance Measure

The following performance metrics are used:

**Accuracy (Acc@k):** Measures how often the true location is in the top-k predictions.

- $P \in \mathbb{R}^{N \times C}$: Predicted probabilities matrix ($N$: samples, $C$: classes),
- $y \in \{1, \ldots, C\}^N$: is the vector of true labels.

$$\text{Top-k Accuracy} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{1}\left(y_i \in \text{Top-k}(P_i)\right)$$

where Top-k$(P_i)$ is the set of $k$ classes with the highest predicted probabilities for sample $i$, and $\mathbb{1}(\cdot)$ is 1 if true, 0 otherwise. We report **Top-k Accuracy** $\times\, 100\%$.

**Mean Reciprocal Rank (MRR):** Computes the average reciprocal rank of the true label:

$$\text{Reciprocal Rank}(i) = \frac{1}{\text{rank}_i(y_i)},$$

$$\text{MRR} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{\text{rank}_i(y_i)}$$

where $\text{rank}_i(y_i)$ is the position of $y_i$ in the sorted predicted probabilities (rank $= 1$ for the highest probability). We report **MRR** $\times\, 100\%$.

**Jensen-Shannon Divergence(JSD):** Measures the similarity between two probability distributions P and Q.

$$\text{JSD}(P \| Q) = \frac{1}{2} D_{\text{KL}}(P \| M) + \frac{1}{2} D_{\text{KL}}(Q \| M),$$

$$\text{where } M = \frac{1}{2}(P + Q)$$

The terms $D_{\text{KL}}(P \| M)$ and $D_{\text{KL}}(Q \| M)$ represent the KL divergence of $P$ and $Q$ with respect to $M$, respectively. The JSD is symmetric and bounded between 0 and 1, with lower values indicating higher similarity between $P$ and $Q$.

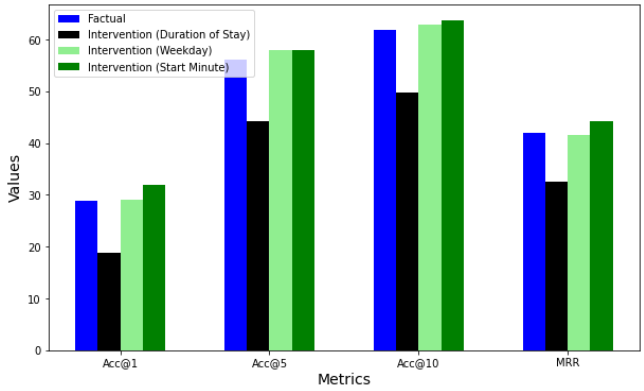
### Results

We conduct extensive experimental analysis to validate our method. Results are obtained considering the data for the 45 selected users in GeoLife as considered by the authors in MHSA [Hong *et al.*, 2023]. These users have observation periods of more than 50 days to provide a longer observational time for getting a meaningful temporal pattern. Average accuracy Acc@k where k = 1, 5, 10 and average MRR values are computed across the users. JSD measures the similarity between factual and counterfactual latent space distribution.

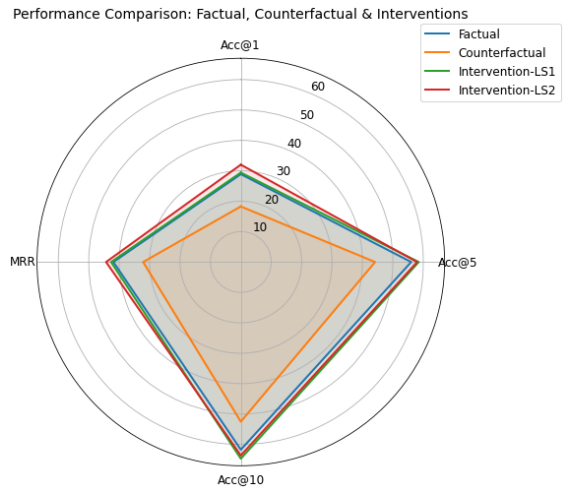
**Generative model configuration:** Our method exploits CVAE with multilayer LSTM-based architecture and self-attention modules.

**Encoder:** The encoder comprises two LSTM layers. First one has 60 hidden units and outputs the full sequence, which is regularized by dropout layer, (dropout = 0.3) to prevent overfitting. The output from this layer is passed through a **self-attention** layer with sigmoid activation, followed by the second LSTM layer with 40 hidden units, the final output is mapped to a latent space (Z) conditioned on X.

**Decoder:** The decoder takes Z as input and repeats it across the maximum sequence length in the training data. This Z, conditioned on X is passed to the LSTM layer with 40 hidden units, which outputs the full sequence. Dropout layer (dropout = 0.3) is applied to the output. The output from the dropout layer is passed to a **self-attention** layer with sigmoid activation, followed by the next LSTM layer with 60 hidden units, and finally, the output from this LSTM is passed to a dense layer with $c_{max}$ (defined in Equation (3)) units of nodes

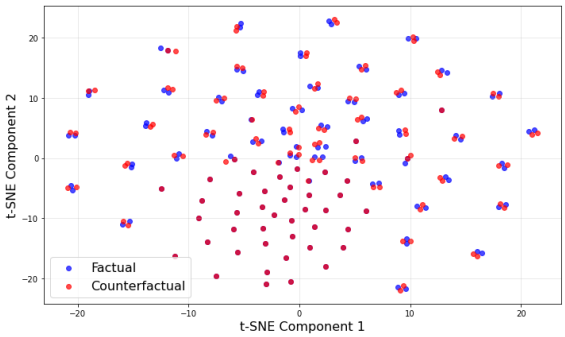



(a) Causally sensitive features

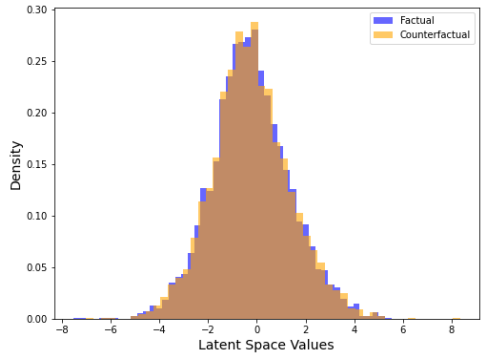



(b) Performance: Factual (F), Intervention (I), Counterfactual (C)

Figure 4: Causal Sensitivity Identification: a: Causally sensitive features, b: Performance across factual, intervention, and counterfactual scenarios.




(a) t-SNE visualization of factual ($Z_{FC}$) and counterfactual ($Z_{CF}$) latent spaces [van der Maaten and Hinton, 2008]




(b) Distribution comparison using JSD between $Z_{FC}$ and $Z_{CF}$ (JSD = 0.4244)

Figure 5: Comparison of latent spaces $Z_{FC}$ and $Z_{CF}$ from the GP-F model: (a) shows the t-SNE projection; (b) shows the Jensen-Shannon divergence between their distributions.

with softmax activation to predict the next location. The reconstruction loss uses sparse categorical cross-entropy.

We apply **KL annealing**, with kl_weight (a gradually increasing weight) as described in the methodology we consider kl_start epoch as 10, so up to 10 epochs the learning focuses only on the reconstruction error, kl_annealtime = 20. We use Adam optimizer, batch size = 32, latent dimension 50, and train our model with 500 epochs.

1. **Causally sensitive variable and establishing cause-effect relationship:**
   - We obtain GP-F, the factual baseline model following equation (4) considering only LS, and measure the performance using factual test data.
   - We intervene LS in $X_train$ by replacing the highest occurring location in LS with LS1 and LS2, resulting as $X_{\text{train_Altered}}$, and then train the interventional model, GP-I, using $X_{\text{train_Altered}}$ conditioning on different candidate features (day of week $W$, start time $Smin$, and duration of stay DS) to be identified as causally sensitive, denoted as $F_{\text{CS}}$.
   - We compute the performance of both GP-F and GP-I on $X_{test}$ to assess changes in next-location prediction performance under intervention.
   - We evaluate W, Smin, and DS as $F_{\text{CS}}$ considering equation (9).

In this scenario, as discussed in section 3.2, conditioning on $Smin$ and $W$, we observe improvement in the best average performance, across all users, i.e., $\Delta\text{Acc} > 0$, (equation (9)), establishing them as causally sensitive features representing the causal path as $LS \leftarrow F_{\text{CS}} \rightarrow Y$, where $F_{\text{CS}}$ acts as a common cause for $LS$ and $Y$.

For duration of stay (DS), we observe average $\Delta\text{Acc} <$

0; and for the best average performance approximately equal. This indicates DS does not have significant causal influence to the LS and the Y next location. Figure 4a depicts the obtained results.

2. **Measure the change in effect when its cause has changed:**
   - We use GP-F, the factual baseline model, to encode the factual test data $X_{\text{test}}$ and obtain the latent representation $Z_{FC}$.
   - To assess counterfactual effects, we alter the sequence of past location visits in $X_{\text{test}}$ to form $X_{\text{test_Altered}}$ (Equation (11)) and compute the counterfactual latent $Z_{\text{CF}}$ (Equation (12)).
   - Using these, we generate both factual (Equation (13)) and counterfactual (Equation (14)) predictions, representing alternate trajectory outcomes.
   - We compare performance metrics of factual and counterfactual predictions to evaluate the impact of changes in causes.

   Figure 4b depicts the counterfactual(C), factual(F) and interventional(I) scenarios where we find the average performance of counterfactual scenario is less than the Factual best one As discussed in section 3.2 i.e., $\Delta\text{Acc} < 0$ in this scenario. This evolution helps to measure the change in effects on the next location visit when the sequence of previous location visits has changed, this further helps to generate the possible alternate trajectories. Figure 5 depicts the divergence of factual and counterfactual latent space distribution.

3. **Causally sensitive prediction of next location of the trajectory:**

   We evaluate the proposed causally sensitive generative predictor using factual data and conditioning on the causally sensitive feature, Smin. For each instance, we generate $n = 20$ samples and report the best-performing prediction. We summarize in Table 4 the obtained results using factual GeoLife data along with the results of relevant state-of-the-art (SoA) as discussed in the related work `Our emphasis`.

   We compare the best-performing results from SoA methods trained on combined user data with our method's results averaged across individual user-level models. Our approach achieves competitive Acc@1 performance, and notably, the GP-F model conditioned on the causally sensitive feature Smin outperforms others in terms of MRR.

4. **Ablation Study:**

   We perform an ablation study by conditioning on the causally sensitive feature and subsequently removing it, as presented in Table 5. In the Baseline scenario, the generative predictor is trained without conditioning on any causally sensitive features. The results demonstrate significant performance improvement when such features are used for conditioning. Specifically, conditioning on Smin increases **Acc@1** by 10.34% and improves the mean reciprocal rank (MRR) by 5.00%. For W, **Acc@1** increases by 3.77%, with MRR remaining nearly unchanged. In contrast, no improvement is observed for DS, confirming its minimal causal influence. Additionally, the impact of altered location sequence LS2 is moderately higher than that of LS1 on the original location sequence LS.

Table 4: GeoLife: Average prediction performance of next location across users.

| **Method** | **Acc@1** | **Acc@5** | **Acc@10** | **MRR** |
|---|---|---|---|---|
| LSTM | 28.4 | 55.8 | 59.1 | 19.3 |
| LSTM + Attention | 29.8 | 54.6 | 58.2 | 21.3 |
| DeepMove | 26.1 | 54.2 | 58.7 | 38.2 |
| MHSA | 31.4 | 56.4 | 60.8 | 42.5 |
| **Proposed GCSP** | | | | |
| $F_{CS}$ =Smin | 31.9 | 59.2 | 64.2 | 43.9 |

Table 5: GeoLife: Ablation study on conditioning with causally sensitive features.

| **Scenario** | **Acc@1** | **Acc@5** | **Acc@10** | **MRR** |
|---|---|---|---|---|
| Baseline (No Conditioning) | 28.895 | 56.018 | 61.737 | 41.897 |
| Conditioning on **Start Minute (Smin)** | 32.018 | 57.980 | 63.647 | 44.310 |
| Conditioning on Weekday (W) | 29.092 | 58.002 | 62.798 | 41.636 |
| Conditioning on Duration of Stay (DS) | 18.859 | 44.205 | 49.755 | 32.576 |

Although not detailed in this paper, we have validated the proposed method on a cross-city mobility dataset, further confirming its ability to identify causally sensitive features across diverse spatiotemporal settings.

## 5 Discussion and Conclusion

We have presented a novel generative causal sensitivity identification method that combines intervention and counterfactual analysis to identify causal influence in prediction tasks.

The proposed method comprises two causal perspectives. The first is to identify causally sensitive features ($F_{CS}$) through interventional analysis, reducing confounding bias by blocking backdoor paths and establishing direct causal links between cause and effect when $F_{CS}$ acts as a confounder. The identified $F_{CS}$ are used as conditioning inputs in the CVAE-based generative predictor to obtain causally sensitive recommendation with improved factual prediction performance.

The second perspective is to assess the change in effect when the cause has changed using counterfactual analysis to identify the causal path, and determine the counterfactual predictions in alternate situations.

We validate our approach using the Asia Bayesian network benchmark. This dataset allows us to verify whether the proposed method can uncover known causal relationships under controlled conditions. We demonstrate that interventions on variables like, *either*, *bronc* lead to significant changes in prediction of downstream nodes such as *dysp*, confirming the method's ability to identify true causal paths. Counterfactual evaluations further highlight the impact of modify-

ing key variables, showing divergence in prediction behavior consistent with the known causal structure. Additionally, our method outperforms CausalVAE in counterfactual sensitivity analysis for the Asia dataset, more accurately identifying direct and indirect causal influences on the target variable.

Applied to the GeoLife GPS trajectory dataset, our method identifies day of the week and start time as causally sensitive features influencing both past and next locations, while duration of stay shows minimal impact. Counterfactual sensitivity is assessed by altering past visits, which reveals shifts in predictions and divergence in latent space. Conditioning on causally sensitive features yields the best performance in factual next location prediction, establishing their importance for this task. Compared to prior works using the same input structure, our method achieves competitive results.

While manual testing of individual features is possible, such empirical approaches lack guarantees of causal relevance and may reflect spurious correlations. Our method offers a unified solution by quantifying causal significance through interventions and counterfactuals.

In summary, our generative causal sensitivity identification method provides a generalizable and interpretable framework for analyzing causal relationships, particularly in prediction tasks where the causal graph is unknown and no structural constraints (such as acyclicity) are imposed during learning. This approach is not limited to human mobility and is extendable to a wide range of applications involving time-series prediction, classification, and personalized recommendation, offering the potential for both performance gains and causal interpretability.

## References


[Bandyopadhyay and Sarkar, 2023] S. Bandyopadhyay and S. Sarkar. Exploring causality aware data synthesis. In *Proc. ACM AIMLSystems*, 2023.

[Dai *et al.*, 2025] Haoyue Dai, Yaqi Xue, Krzysztof Chalupka, and Elias Bareinboim. When selection meets intervention: Additional complexities in causal discovery. In *International Conference on Learning Representations (ICLR)*, 2025.

[Doersch, 2021] C. Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:2111.10846*, 2021.

[Feng *et al.*, 2018] J. Feng, L. Yong, C. Zhang, F. Sun, F. Meng, A. Guo, and D. Jin. Deepmove: Predicting human mobility with attentional recurrent networks. In *Proc. WWW Conf.*, pages 1459–1468, 2018.

[Feng *et al.*, 2020] J. Feng, Z. Yang, F. Xu, H. Yu, M. Wang, and Y. Li. Learning to simulate human mobility. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3426–3433, 2020.

[Goodfellow *et al.*, 2014] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.

[Hershey and Olsen, 2007] John R. Hershey and Peder A. Olsen. Approximating the kullback–leibler divergence between gaussian mixture models. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 4, pages IV–317. IEEE, 2007.

[Hong *et al.*, 2023] Y. Hong, Y. Zhang, K. Schindler, and M. Raubal. Context-aware multi-head self-attentional neural network model for next location prediction. *Transp. Res. Part C: Emerg. Technol.*, 156, 2023.

[Ke *et al.*, 2020] Nan Rosemary Ke, Olexa Bilaniuk, Anirudh Goyal, Stephan Bauer, Hugo Larochelle, Chris Pal, and Yoshua Bengio. Learning neural causal models from unknown interventions. In *International Conference on Learning Representations (ICLR)*, 2020.

[Kingma and Welling, 2013] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[Kment, 2020] Boris Kment. Counterfactuals and causal reasoning. In *Perspectives on Causation: Selected Papers from the Jerusalem 2017 Workshop*, pages 463–482. Springer, 2020.

[Krishna *et al.*, 2018] Kalpit Krishna, Devendra Jain, Shobhit V. Mehta, and Shubham Choudhary. An lstm-based system for prediction of human activities with durations. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(4):147:1–147:31, 2018.

[Lauritzen and Spiegelhalter, 1988] Steffen L. Lauritzen and David J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2):157–194, 1988.

[Li *et al.*, 2019] Chunyuan Li, Xiujun Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. Cyclical annealing schedule: A simple approach to mitigating kl vanishing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 240–250. Association for Computational Linguistics, 2019.

[Li *et al.*, 2020] Fei Li, Zhen Gui, Zhili Zhang, Dawei Peng, Shuang Tian, Kai Yuan, Yafei Sun, Huayi Wu, Jing Gong, and Yinjie Lei. A hierarchical temporal attention-based lstm encoder-decoder model for individual mobility prediction. *Neurocomputing*, 403:153–166, 2020.

[Liu and others, 2021] X. Liu et al. Trajgans: Geo-privacy protection of trajectory data. In *GIScience*, 2021.

[Louizos *et al.*, 2017] Christos Louizos, Uri Shalit, Joris Mooij, David Sontag, Rich Zemel, and Max Welling. Causal effect inference with deep latent-variable models. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[MacDonald and Zucchini, 1997] Iain L. MacDonald and Walter Zucchini. *Hidden Markov and Other Models for Discrete-Valued Time Series*, volume 110 of *Monographs*

*on Statistics and Applied Probability*. CRC Press, Boca Raton, FL, 1997.

[Martin *et al.*, 2022] H. Martin, Y. Hong, N. Wiedemann, D. Bucher, and R. Martin. Trackintel: An open-source python library for human mobility analysis. *arXiv preprint arXiv:2206.03593*, 2022.

[Nezhadettehad *et al.*, 2024] Amin Nezhadettehad, Arkady Zaslavsky, Rafiq Abdur, S. A. Shaikh, Seng W. Loke, Guang-Li Huang, and Ali Hassani. Predicting next useful location with context-awareness: The state-of-the-art. *arXiv preprint arXiv:2401.08081*, 2024.

[Pearl and Mackenzie, 2018] Judea Pearl and Dana Mackenzie. *The Book of Why: The New Science of Cause and Effect*. Basic Books, 2018.

[Pearl, 2009] J. Pearl. *Causality: Models, reasoning, and inference*. Cambridge University Press, New York, 2nd edition, 2009.

[Pearl, 2019] J. Pearl. *Causal and counterfactual inference*. Springer, New York, 2019.

[Rudenko *et al.*, 2020] Andrey Rudenko, Luigi Palmieri, Michael Herman, Kris M. Kitani, Dariu M. Gavrila, and Kai O. Arras. Human motion trajectory prediction: A survey. *International Journal of Robotics Research*, 39(8):895–935, 2020.

[Salzmann *et al.*, 2020] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone. Trajectron++: Multi-agent generative trajectory forecasting with heterogeneous data for control. *CoRR*, abs/2001.03093, 2020.

[Scutari, 2009] Marco Scutari. The bnlearn dataset repository. https://www.bnlearn.com/bnrepository/, 2009. Accessed: 2025-05-25.

[van der Maaten and Hinton, 2008] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

[Vaswani *et al.*, 2017] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Proc. 31st NeurIPS*, pages 5998–6008. Curran Associates, 2017.

[Wang *et al.*, 2022] S. Wang, J. Cao, and P. S. Yu. Deep learning for spatio-temporal data mining: A survey. *IEEE Trans. Knowl. Data Eng.*, 34(8):3681–3700, 2022.

[Xia *et al.*, 2023] Kun Xia, Yang Pan, and Elias Bareinboim. Causal models for counterfactual identification and estimation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2023.

[Yang *et al.*, 2018] Karren Yang, Abigail Katcoff, and Caroline Uhler. Characterizing and learning equivalence classes of causal dags under interventions. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 5541–5550. PMLR, 2018.

[Yang *et al.*, 2021a] Mengyang Yang, Fanqian Liu, Zhiqian Chen, Xinyu Shen, Jingyao Hao, and Jun Wang. Causalvae: Disentangled representation learning via neural structural causal models. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9593–9602, 2021.

[Yang *et al.*, 2021b] Mengyang Yang, Feng Liu, Zhihui Chen, Xiaohui Shen, Jun Hao, and Jingdong Wang. Causalvae: Structured causal disentanglement in variational autoencoder. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9593–9602. IEEE, 2021.

[Yang *et al.*, 2021c] Xinrui Yang, Hongyang Zhang, Guosheng Qi, and Jianbo Cai. Causal attention for vision-language tasks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[Zheng *et al.*, 2010] Y. Zheng, X. Xie, and W.-Y. Ma. Geolife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull.*, 33(2):32–39, 2010.

[Zheng *et al.*, 2018] Xun Zheng, Bryon Aragam, Pradeep Ravikumar, and Eric P. Xing. Dags with no tears: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.

[Zuo *et al.*, 2022] A. Zuo, S. Wei, T. Liu, B. Han, K. Zhang, and M. Gong. Counterfactual fairness with partially known causal graph. In *Proc. AAAI Conference on Artificial Intelligence*, 2022.