Learning Causal Orderings for In-Context Tabular Prediction

Sascha Xu[◦]
CISPA Helmholtz Center
sascha.xu@cispa.de

Sarah Mameche[◦ *]
ISTA
sarah.mameche@ist.ac.at

Jilles Vreeken
CISPA Helmholtz Center
jv@cispa.de

Abstract

In-context learning for tabular data sets strong predictive standards in observational settings; it however primarily relies on correlational structure, which becomes unreliable under distribution shift or intervention. While established methods to discover causal structure exist, they are often focused on structure identifiability and decoupled from the predictive architectures that could benefit from them.

To bridge these perspectives, we study how to simultaneously infer and enforce causal structure in the form of topological variable orderings into tabular prediction. Unlike standard architectures, our model TABORDER uses causal order-constrained attention, basing predictions only on features that precede a target under a learned causal order. Similar to causal discovery methods, TABORDER learns the optimal variable ordering in an unsupervised manner through a likelihood-based objective. We justify this choice under standard functional model classes and also study how sample missingness, a common challenge in tabular data, interacts with causal direction identification. Empirically, we confirm that TABORDER recovers accurate variable orderings while addressing prediction and imputation tasks and provides insight into real-world biological data under intervention.

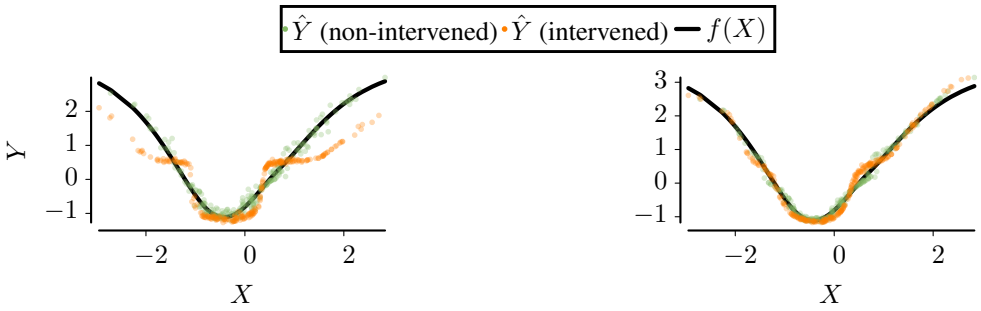
1 Introduction

In-context learning architectures such as Tabular foundation models (TFMs) achieve strong performance in predictive learning from tabular data through large-scale pre-training [Müller et al., 2022, Hollmann et al., 2023]. However, these models are primarily associative and do not explicitly account for causal directionality [Pearl, 2009] among features, such that their predictions could rely on non-causal associations sensitive to distribution shifts. We illustrate this effect in Figure 1 in a three-variable chain $X \to Y \to Z$, comparing an observational regime (green) to an intervention on the mechanism generating Z (orange). This affects the predictive performance of a state-of-the-art TFM (Figure 1a), suggesting that it relies on the non-causal statistical association between Z and Y. In contrast, predictions with our proposed architecture remain robust (Figure 1b) as the model relies only on the direct causal relationship between X and Y.

Identifying such directional relationships is the primary aim of causal discovery [Pearl, 2009], usually with focus on recovering the underlying directed acyclic graph (DAG) from observational data. However, classical methods [Spirtes et al., 2001] are typically dataset-specific and detached from large-scale predictive architectures. Recent amortized approaches to causal discovery [Lorch et al., 2022, Ke et al., 2023] apply an in-context learning paradigm to causal discovery, relying on supervised training over large collections of known causal graphs. However, these works focus on a different question of whether causal structure recovery can be improved by supervised learning over large synthetic samples [cf. Montagna et al., 2025], whereas the question of how to systematically use causal structure in predictive architectures is relatively underexplored.

[◦]Equal contribution. [*]Work done during PhD at CISPA Helmholtz Center for Information Security.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).



(a) Prediction with TABPFN. (b) Prediction with TABORDER.

Figure 1: **Prediction under Intervention with and without Order Constrained Attention**. For prediction of a mediator Y in a chain $X \to Y \to Z$, we measure test error before and after intervention on $Y \to Z$. The generating mechanism $f(X)$ is shown in black. TABPFN (a) accurately models $X \to Y$ when no intervention is present (green), but fails under intervention (orange). TABORDER (b) remains accurate in both settings by learning and leveraging the causal order.

In this work, we propose to connect the research perspectives behind predictive tabular models on the one hand, and causal structure discovery on the other hand, within a transformer-based framework. Unlike standard architectures, our model TABORDER uses causal order-constrained attention, modeling the joint distribution as an ordered factorization under a learned causal ordering. In contrast to classical causal discovery, TABORDER infers this order efficiently within a single forward pass; in addition, most existing causal discovery methods address a fully observed regime, while we allow for missing samples with direct mechanisms for imputation. Finally, different from amortized causal discovery, we maintain an unsupervised approach without requiring ground-truth causal orders during training.

Given that recovering causal structure from purely observational data is impossible without further assumptions, we design our architecture around a likelihood-based objective that we justify in Additive Noise Model (ANM) regimes under standard assumptions [Bühlmann et al., 2014]. We study how the prediction of arbitrary missing values can be used to learn causal orderings, and show that the presence of missing samples can actually facilitate causal direction identification (Section 3). Experiments across different classes of generating processes (Section 5) confirm that TABORDER reliably infers accurate causal orderings, while maintaining competitive predictive performance on downstream tasks such as missing value imputation.

2 Causal Order-Constrained Prediction

We begin by introducing the problem setting, then explain the features of our architecture.

Let $\mathbf{X} = (X_1, \ldots, X_d)$ denote a collection of d real-valued random variables drawn from an unknown joint distribution $P(\mathbf{X})$. We observe a dataset $\mathcal{D} = \{\mathbf{x}^{(r)}\}_{r=1}^{n}$ over n samples, where each row $\mathbf{x}^{(r)} = (x_1^{(r)}, \ldots, x_d^{(r)}) \in \mathbb{R}^d$ is an independent realization of $\mathbf{X}$, and where some entries $x_i^{(r)}$ are missing, indicated by a binary mask $M = \{m_i^{(r)}\}_{r=1}^{n}$ with $m_i^{(r)} \in \{0, 1\}$. Our objective is to learn the structured conditional dependencies between variables in $\mathbf{X}$ to enable accurate and robust cell-wise predictions.

We base the architecture design on **tabular foundation models** (TFMs) [Hollmann et al., 2023], which model the posterior predictive distribution of a target variable $x_i^{(r)}$ in row r as $p_\theta(x_i \mid \mathbf{x}_{i^*}^{(r)}, \mathcal{D})$, where $\mathbf{x}_{i^*}^{(r)}$ denotes all entries in row r except $x_i^{(r)}$.

This approach however allows the use of all available features as predictors for $x_i^{(r)}$ without accounting for causal directionalities. To address this, we propose modeling the joint distribution of a row $\mathbf{x}^{(r)}$

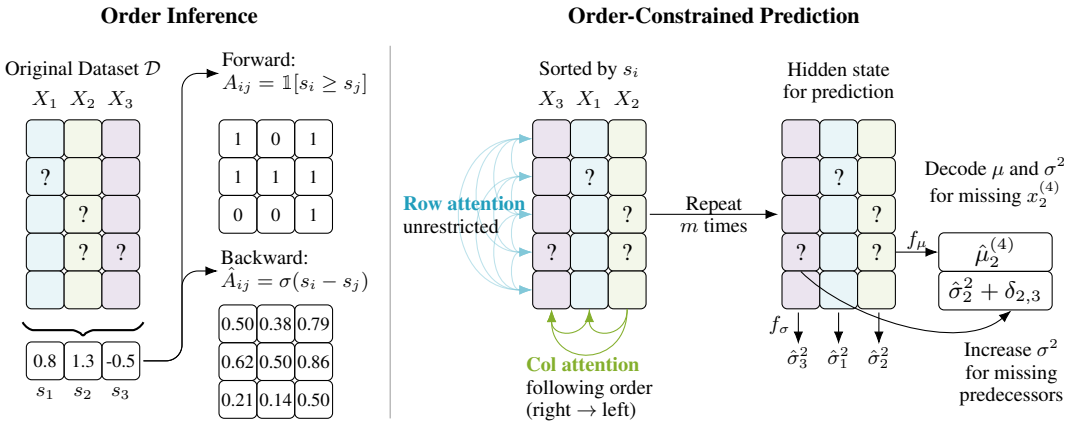


Figure 2: **Overview of TABORDER.** For a dataset $\mathcal{D}$ we map each column i to an order score s_i. From these scores, we construct a hard/soft attention mask to constrain attention according to the learned order (left). To predict missing entries in $\mathcal{D}$, we alternate row-wise attention (unrestricted) and column-wise causal attention (order-constrained). From the resulting cell embeddings, we decode the conditional mean and a variance term accounting for missing potential causes (right).

using an **ordered factorization** of the form

$$p_\theta(\mathbf{x}^{(r)} \mid \mathcal{D}, \pi) = \prod_{k=1}^{d} p_\theta\left(x^{(r)}_{\pi(k)} \mid x^{(r)}_{\pi(1:k-1)}, \mathcal{D}\right) . \tag{1}$$

Under this factorization, the prediction for an entry $x^{(r)}_{\pi(k)}$ is conditioned strictly on the variables preceding it in the order $\pi \in S_d$, where S_d is the set of all possible variable permutations and $\pi(k)$ denotes the index of the variable at position k. This induces an asymmetric dependency structure, where each variable is predicted only from its predecessors. If the learned order aligns with the underlying causal structure, the model is prevented from exploiting anti-causal or spurious correlations that do not generalize under intervention.

We are specifically interested in **causal orders** where π is compatible with the underlying cause-effect relationships of the data-generating process, i.e., corresponds to a topological sorting of an underlying Directed Acyclic Graph (DAG) [Pearl, 2009]. However, ground-truth causal orders π are often unknown or only partially observed in real-world settings [e.g., Sachs et al., 2005]. Therefore, we design the architecture to learn an optimal ordering $\hat{\pi}$ end-to-end, without requiring ground-truth causal labels during training. We explain this approach in the following, then explain its correspondence with inferring causal orders in Section 3.

2.1 TABORDER Architecture

TABORDER is a dual-transformer architecture that infers causal orders and performs order-constrained predictions in a single end-to-end pass. The model operates through two primary stages. Given a table $\mathcal{D} \in \mathbb{R}^{n \times d}$ with missing entries indicated by $M \in \{0,1\}^{n \times d}$, each cell $x_i^{(r)}$ and its missingness indicator $m_i^{(r)}$ are embedded through a learned linear map

$$\mathbf{e}_i^{(r)} = W_{\text{emb}} \begin{bmatrix} x_i^{(r)} \\ m_i^{(r)} \end{bmatrix} + \mathbf{b}_{\text{emb}} , \qquad W \in \mathbb{R}^{h \times 2} . \tag{2}$$

These embeddings are processed by alternating row-wise and column-wise self-attention layers in two specialized transformer branches: an unconstrained branch that induces a variable ordering $\hat{\pi}$ via scalar scores $\mathbf{s}$ (*Order Inference*), and a masked branch that models the conditional distribution of each cell, using a mask constructed from the learned scores $\mathbf{s}$ (*Order-Constrained Prediction*). We provide a high-level overview of TABORDER in Fig. 2 and explain its components in detail below.

Order Inference. To infer the ordering $\hat{\pi}$, we apply a standard multi-head self-attention transformer [Vaswani et al., 2017] to the initial cell embeddings, alternating between row- and column-wise attention for m steps. This results in a hidden state $H^{\text{ord}} \in \mathbb{R}^{n \times d \times h}$.

We model each variable's X_i position in the order via a scalar score $s_i \in \mathbb{R}$, where lower scores indicate an earlier position. We decode these scores from the hidden state H^{ord} by applying a two-layer MLP $f_{\text{ord}} : \mathbb{R}^h \to \mathbb{R}$ to each cell representation, and averaging across rows as per

$$s_i(\mathcal{D}) = \frac{1}{n} \sum_{r=1}^{n} f_{\text{ord}} \left(H^{\text{ord}}_{r,i} \right) \ . \tag{3}$$

To enforce the order imposed by the scores $\mathbf{s}$ in the *Order-Constrained Prediction* branch, we construct a causal attention mask $A^{\mathbf{s}}$, where $A^{\mathbf{s}}_{i,j} = \mathbb{1}(s_j \leq s_i)$, meaning that column i may only attend to columns with lower score. To maintain differentiability, we use a straight-through relaxation in the backward pass as

$$\hat{A}^{\mathbf{s}}_{i,j} = \text{sigmoid}\left(\frac{s_i - s_j}{\tau} \right) \ , \tag{4}$$

where the temperature τ controls the sharpness of the ranking.

The *Order Inference* transformer thus produces a differentiable ordering that controls the information flow in the *Order-Constrained Prediction* branch. Since both components are trained jointly, the model is encouraged to place variables earlier in the order when they improve prediction of downstream variables. For causal mechanisms with lower prediction error in the causal direction [Blöbaum et al., 2018], incorrect orderings incur systematically higher error and are therefore suboptimal.

Order-Constrained Prediction. We now model the structured conditional dependencies using the causal mask $A^{\mathbf{s}}$. We apply a second transformer to the initial cell embeddings, alternating row- and column-wise attention for m steps. While row-wise attention remains unconstrained to allow for in-context information sharing across samples, column-wise attention is strictly restricted by $A^{\mathbf{s}}$, ensuring that the hidden state in column i depends only on its predecessors in the learned order $\hat{\pi}$.

This yields predictive hidden states $H^{\text{pred}} \in \mathbb{R}^{n \times d \times h}$ that parameterize the conditional factorization of Eq. (1). We model each variable's conditional distribution using **additive noise models** (ANMs)

$$X_i = f(X_{pa(i)}) + N_i, \ N_i \perp\!\!\!\perp X_{pa(i)} \ , \tag{5}$$

with Gaussian noise $N_i \sim \mathcal{N}(0, \sigma_i^2)$. To that end, we decode the conditional mean and the variance, averaged per variable, via two MLPs f_μ and f_σ as

$$\hat{\mu}_i^{(r)} = f_\mu \left(H^{\text{pred}}_{r,i} \right), \quad \hat{\sigma}_i^2 = \frac{1}{n} \sum_{r=1}^{n} f_\sigma \left(H^{\text{pred}}_{r,i} \right) \ . \tag{6}$$

Prediction under Partial Observation. Since the model predicts potentially multiple missing entries per row, missing causal parents affect the estimation of the functional relationships in Eq. (5), leading to increased uncertainty when relevant inputs are unobserved. We capture this effect via a learnable variance increment $\delta_{i,j} \in \mathbb{R}^+$, decoded from H^{pred}, such that the point-wise variance for a cell $x_i^{(r)}$ is given by

$$\hat{\sigma}_i^{2(r)} = \hat{\sigma}_i^2 + \sum_{j \in [d]} \delta_{i,j} \cdot \mathbb{1}(s_j < s_i) \cdot m_j^{(r)} \ . \tag{7}$$

Intuitively, the model increases predictive uncertainty when variables that precede X_i in the learned order are missing. For example, in a collider structure $X_1 \to X_3 \leftarrow X_2$, missing either parent increases the uncertainty in predicting X_3 (Fig. 3). Moreover, we show that this mechanism amplifies the likelihood gap between correct and incorrect orderings, thus facilitating the learning of accurate causal orders (Section 3).

X_1	X_2	$X_3 = X_1 + (X_2)^2 + N$
1	2	$\hat{\mu}_3^{(1)} = 5, \ \hat{\sigma}_3^{2(1)} = 1$
0	?	$\hat{\mu}_3^{(2)} = 0, \ \hat{\sigma}_3^{2(2)} = 1 + 2$
?	-1	$\hat{\mu}_3^{(3)} = 1, \ \hat{\sigma}_3^{2(3)} = 1 + 1$

Figure 3: **Effect of Missingness.** In an additive noise model, $X_3 = X_1 + (X_2)^2 + N$ where $N, X_1, X_2 \sim \mathcal{N}(0, 1)$, when either parent of X_3 is missing, the model increments the variance $\hat{\sigma}^2$ (Eq. (7)) by a learned amount (blue for X_1, green for X_2) that reflects the effect of missingness of causal parents.

Training. To train our model, we optimize all components jointly by maximizing the log-likelihood of masked entries under the ordered conditional model. We assume access to a distribution over datasets $\mathcal{D} \sim \mathcal{P}$, from which training instances are sampled. For each dataset, we generate a prediction task by masking entries completely at random with probability $q = 20\%$. Given a masked dataset $(\mathcal{D}, M)$, we first infer an attention mask $A^{\mathbf{s}}$ via the order induction module. Conditioned on this mask, the predictive transformer produces cell-wise mean $\hat{\mu}_i^{(r)}$ and variance estimates $\hat{\sigma}_i^{2(r)}$. The training objective is then to maximize the average log-likelihood $\mathcal{L}$ of the masked entries,

$$\mathcal{L} = \frac{1}{|M|} \sum_{m_i^{(r)}=1} \log \left[p_{\mathcal{N}}\Big(x_i^{*(r)} \mid \hat{\mu}_i^{(r)}, \hat{\sigma}_i^{2(r)} \Big) \right]. \tag{8}$$

Maximizing Eq. (8) couples order learning and prediction: orderings that better explain masked entries receive higher likelihood. Together, the architecture and objective of TABORDER provide an in-context learning framework for inferring variable orderings and structured conditional dependencies.

We describe our exact implementation including relevant hyperparameter choices in Appendix B.

3 Causal Order Inference

In this section, we provide a theoretical perspective on TABORDER and examine under which conditions its architecture and likelihood-based training objective are aligned with recovering causal structure. In particular, we focus on learning variable orderings without requiring ground-truth causal graphs during training.

As is standard, we assume the data to be generated by a Structural Causal Model (SCM) [Pearl, 2009] with structural equations $X_i = f_i(pa_i, N_i)$, where pa_i denotes the set of direct causes (parents) of X_i, and N_i is an exogenous noise variable. The causal structure of the entire system is captured by a directed acyclic graph (DAG). A **topological ordering** of a DAG is any order in which all causal parents precede their children; for such an ordering π, the joint density factorizes as

$$p(\mathbf{x}) = \prod_{i=1}^{d} p\big(x_{\pi(i)} \mid x_{\pi(1:i-1)}\big) = \prod_{i=1}^{d} p\big(x_i \mid x_{pa_i}\big) \ . \tag{9}$$

Hence we refer to $\pi \in S_d$ as a causal order for X if it corresponds with a topological ordering of the underlying causal DAG. We make standard assumptions of the causal Markov Condition, faithfulness, and causal sufficiency, listed in detail in Appendix A.1.

Order Inference via Likelihood-Based Objectives. Identifying the causal graph and thereby a topological ordering is feasible for certain classes of functional mechanisms [Shimizu et al., 2006, Hoyer et al., 2008, Zhang and Hyvärinen, 2009]. As our learning objective is based on minimizing conditional error variance, we focus on **causal additive models** (CAM) [Bühlmann et al., 2014],

$$X_i = \sum_{j \in pa_i} f_{ij}(X_j) + N_i, \tag{10}$$

with smooth functions f_{ij} and independent noise $N_i \perp\!\!\!\perp X_{pa_i}$. For this class, the likelihood-optimal order is consistent with the topological orders of the true DAG, such that scoring criteria based on residual variances consistently recover valid causal orders [Bühlmann et al., 2014]. As TABORDER jointly predicts a variable order $\hat{\pi}$ as well as models conditional expectations under the order-constrained SCM (Eq. 5) via likelihood maximization, this steers training toward orderings that are consistent with the underlying causal structure, without requiring ground-truth graphs.

Order Inference under Partial Observation. In the presence of missing values, TABORDER includes an explicit mechanism to capture the effect on the predictive uncertainty in the additive noise model in Eq. (5) (see Figure 3). We now show that in certain cases, we can leverage an asymmetry in the effect of missingness to further increase likelihood terms in the causal direction.

To illustrate, consider the bivariate cause effect pair $X \to Y$. Let $M \sim \text{Bernoulli}(q)$ indicate missingness of the parent, and for direction τ, define $v_{0,\tau}$ as the residual variance when the parent is observed ($M = 0$) and $v_{1,\tau}$ when it is missing ($M = 1$).

A Gaussian model with a single variance parameter in direction τ must match the mask-averaged residual variance $\bar{\sigma}^2_\tau = (1-q)\, v_{0,\tau} + q\, v_{1,\tau}$, yielding population log-likelihood $\mathcal{L}^\star_{\tau,\text{fix}} = -\frac{1}{2}\log \bar{\sigma}^2_\tau +$ const . A missingness-aware model fits separate variances for $M = 0$ and $M = 1$, giving $\mathcal{L}^\star_{\tau,\text{inc}} = -\frac{1}{2}\big[(1-q)\log v_{0,\tau} + q \log v_{1,\tau}\big] + \text{const}$. Since $-\log$ is strictly convex, if $v_{0,\tau} \neq v_{1,\tau}$ then Jensen's inequality implies $\mathcal{L}^\star_{\tau,\text{inc}} > \mathcal{L}^\star_{\tau,\text{fix}}$, that is accounting for missingness strictly improves the likelihood in either direction.

The above suggests that the variance increment in Eq. (7) helps the model distinguish cause from effect. We now examine conditions under which this is the case. To that end, we compare

$$\Delta_{\text{fix}} := \mathcal{L}^\star_{\text{fwd,fix}} - \mathcal{L}^\star_{\text{bwd,fix}} \text{ and } \Delta_{\text{inc}} := \mathcal{L}^\star_{\text{fwd,inc}} - \mathcal{L}^\star_{\text{bwd,inc}} \,. \tag{11}$$

Then, we can show the following.

Theorem 1 (Effect of Missingness on Likelihood Asymmetry)**.** *Consider an additive noise model $Y = f(X) + N$ with $N \perp\!\!\!\perp X$ and $\mathrm{Var}(N) > 0$ and missingness indicator $M_X, M_Y \sim \mathrm{Bernoulli}(q)$ with $q \in (0,1)$. Then, the difference in likelihood is amplified iff the forward signal-to-noise ratio $\mathrm{Var}(f(X))/\mathrm{Var}(N)$ exceeds the backward ratio $\mathrm{Var}(\mathbb{E}[X \mid Y])/\mathbb{E}[\mathrm{Var}(X \mid Y)]$, i.e.,*

$$\Delta_{\text{inc}} > \Delta_{\text{fix}} \quad \Longleftrightarrow \quad \frac{\mathrm{Var}(f(X))}{\mathrm{Var}(N)} > \frac{\mathrm{Var}(\mathbb{E}[X \mid Y])}{\mathbb{E}[\mathrm{Var}(X \mid Y)]} \,. \tag{12}$$

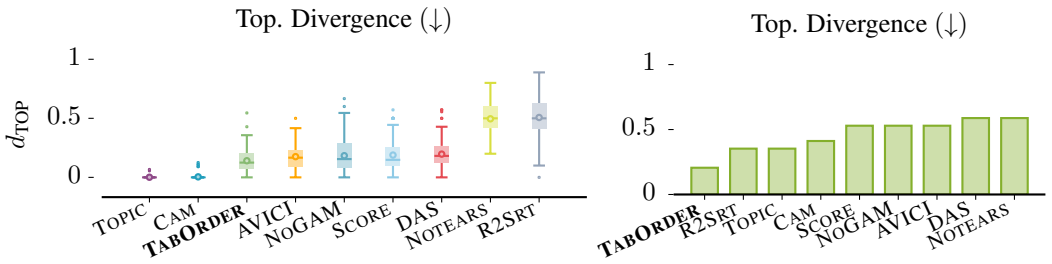
We defer the proof to Appendix A.3. The result shows that optimizing the likelihood with the variance increment in Eq. (7) further increases the likelihood gap, facilitating learning of causally aligned orders in identifiable settings such as CAMs. More generally, in settings where causal structure and likelihood-optimal factorizations coincide [Wendong et al., 2025], likelihood maximization provides a principled approach to learning variable orderings without requiring ground-truth graphs. TABORDER instantiates this principle in an in-context learning framework.

4 Related Work

Tabular Foundation Models. Recent deep learning approaches for tabular data adopt an in-context learning paradigm using transformer-based models trained on large collections of tables [Müller et al., 2022]. TABPFN introduced a transformer pre-trained on synthetic tasks drawn from a prior over structural mechanisms, achieving state-of-the-art performance without finetuning [Hollmann et al., 2023, 2025]. Subsequent work improves scalability [Qu et al., 2025] and retrieval [Ma et al., 2026]. Existing TFMs primarily focus on predicting a single target variable and do not impose structural constraints on feature–feature relationships. Work on causality in TFMs has so far focused on treatment effect estimation [Robertson et al., 2025, Balazadeh et al., 2026], typically relying on amortized learning over a synthetic SCM prior rather than inferring dataset-specific causal structure.

Missing Value Imputation. Classical imputation methods include mean imputation, k-nearest neighbors, and Multiple Imputation by Chained Equations (MICE) [Van Buuren and Groothuis-Oudshoorn, 2011], which iteratively fits conditional models. MIRACLE [Kyono et al., 2021] explicitly models structured missingness by conditioning on estimated causes of missingness, improving robustness under non-random missingness. This is complementary to our approach, which focuses on learning causal relationships between features to improve imputations. Deep generative models such as GAIN [Yoon et al., 2018] and MIWAE [Mattei and Frellsen, 2019] learn joint feature distributions, while transformer-based approaches such as NAIM [Caruso et al., 2024] model conditional distributions directly. TABIMPUTE [Feitelberg et al., 2025] leverages pretraining on synthetic missingness tasks but does not impose structural constraints.

Causal Discovery. Identifying causal structure from observational data requires assumptions on functional forms or noise distributions [Shimizu et al., 2006, Hoyer et al., 2008, Zhang and Hyvärinen, 2009]. TABORDER is conceptually related to order-based discovery methods [Bühlmann et al., 2014, Rolland et al., 2022, Montagna et al., 2023a,b, Xu et al., 2025] and gradient-based DAG learners [Zheng et al., 2018, Lachapelle et al., 2020]. However, these methods solve dataset-specific optimization problems to infer causal structure from a single dataset. In contrast, amortized approaches like AVICI [Lorch et al., 2022] learn to infer graphs across datasets through supervised training on ground-truth graphs [Ke et al., 2023], although it has been suggested that this supervised approach may be sensitive to synthetic priors [Montagna et al., 2025]. TABORDER uniquely combines amortized learning with unsupervised, likelihood-based order discovery for downstream prediction.



(a) Synthetic Data (ANM).

(b) Real-World Data [Sachs et al., 2005].

Figure 4: **Causal Order Inference.** Shown is the topological divergence (d_{TOP}, lower is better) of estimated causal orders on synthetic datasets generated from nonlinear Gaussian process ANMs (**a**) and on the real-world causal discovery benchmark by Sachs et al. [2005] (**b**).

5 Experiments

We evaluate whether TABORDER achieves a favorable trade-off between I. causal order learning and II. tabular prediction and imputation tasks.

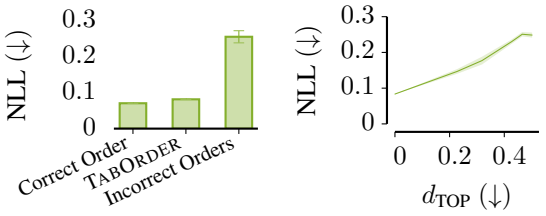
We consider both synthetic data with known causal structure, and real-world datasets with and without ground-truth graphs. Synthetic experiments assess the accuracy of learned orders under a range of nonlinear mechanisms (additive and non-additive), noise levels, and sample sizes (Appendix C.1). Real-world experiments evaluate predictive performance in missing value imputation and downstream tasks, including robustness under distribution shift and intervention. We provide a public implementation of TABORDER and the experimental setup at `https://github.com/Schascha1/TabOrder`.

5.1 Causal Order Inference (Trade Off I)

We first evaluate the quality of the causal orders discovered by TABORDER. Given a learned order $\hat{\pi}$ and ground-truth DAG G, we measure topological divergence $d_{\text{TOP}}(\hat{\pi}, G) = \sum_{i=1}^{d} \sum_{j:\hat{\pi}(i)\geq\hat{\pi}(j)} \mathbb{I}[(i,j) \in G]$, where lower values indicate fewer edge direction disagreements of $\hat{\pi}$ with G. For comparison, we consider a selection of topological-ordering based causal discovery algorithms, CAM [Bühlmann et al., 2014], SCORE [Rolland et al., 2022], DAS [Montagna et al., 2023a], NOGAM [Montagna et al., 2023b], NOTEARS [Zheng et al., 2018] and TOPIC [Xu et al., 2025], as well as AVICI [Lorch et al., 2022] and the baseline R2SRT [Reisach et al., 2023].

Figure 4 shows the topological divergence d_{TOP} across methods. On synthetic data (Figure 4 (a)), TABORDER discovers causal orderings of matching quality compared to the outputs of structure learning algorithms tailored to this task. It discovers such orders without explicit regression and graph learning (CAM, TOPIC) or their associated scalability constraints (Appendix E.2), while enabling near-instant order inference and, unlike AVICI, requiring no ground-truth graphs during training.

Order Impact. Next we assess predictive performance in relation to the quality of the learned causal order. We impose different orders when evaluating TABORDER, including a correct causal order, the learned order, and multiple sampled incorrect orders. To show the effect on missing value imputation, we report the Negative Log-Likelihood (NLL, smaller is better). As Figure 5a shows, the NLL of the correct and learned order is close and much smaller than for random or reversed orders. For these incorrect orders, we show in Figure 5b how the NLL behaves as the divergence d_{TOP} of corrupted orders from the true order increases, showing it worsens mono-



(a) Replacing the order with correct vs. incorrect orders.

(b) NLL vs. divergence between used and true order.

Figure 5: **Order Impact.** Shown is the effect of replacing learned orders at test time. Incorrect orders degrade prediction (NLL) as the considered order diverges from the true causal order (d_{TOP}).

Setting	d_{TOP}	Δ
In-prior reference	0.181	-
Polynomial mechanisms	0.205	0.024
MNAR, 20% missingness	0.202–0.212	0.021–0.031
MNAR, 50% missingness	0.286–0.404	0.105–0.223
Latent confounding	0.308	0.127

Table 1: **Assumption violations.** Topological divergence (d_{TOP}, lower is better).

Environment	Correct edges	d_{TOP}
REGED0 (observational)	620/823	0.247
REGED1 (interventional)	568/823	0.310
REGED2 (interventional)	547/823	0.335

Table 2: **REGED benchmark.** Topological divergence (d_{TOP}, lower is better).

tonically. Additional ablations show that the missingness-aware variance increment improves both likelihood and order recovery, while unrestricted attention has a much larger NLL increase under intervention (0.175–0.237 versus 0.041–0.056 with order-constrained attention); detailed results are in Appendix E.1.

Assumption Violations. The guarantees for causal order recovery rely on identifiable functional models and causal sufficiency. We therefore evaluate shifts away from these conditions. Order recovery remains close to the in-prior result under polynomial mechanisms and moderate MNAR masking, but degrades under severe value-dependent missingness and latent confounding (Table 1). At 20% MNAR masking, value-dependent selection has little effect on the inferred order; at 50%, it causes a clear loss of accuracy. Latent confounding also increases divergence, consistent with the fact that unobserved common causes violate the assumptions used for order identification.

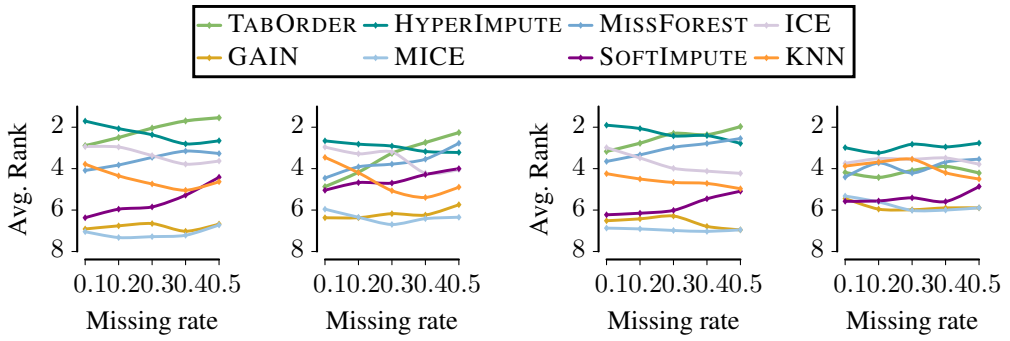
Order Discovery on Real-World Data. We also evaluate order discovery on a common standard real-world benchmark consisting of multiparameter single-cell data [Sachs et al., 2005] where a given causal DAG reflects causal cell signaling pathways among 11 variables. The main results (Figure 4 (b)) show that TABORDER is the only method to discover causal order with $d_{\text{TOP}} \approx 0.21$ divergence. We show the divergence for the main reference dataset (condition `cd3cd28`) but besides this, multiple interventional datasets are available, for which TABORDER consistently achieves topological divergences of $0.12 - 0.29$, with the exception of one condition where a perturbation on the hub node PKC was applied (condition `g0076`). All results are shown in detail in Appendix D.2.

We further evaluate whether an order learned from the synthetic training prior transfers to the REGED gene-expression benchmark, which provides a known causal network and observational and interventional environments [Statnikov et al., 2015]. We apply the pretrained model without fine-tuning to a 499-gene induced subgraph. Table 2 shows that the inferred order agrees with 620 of 823 directed edges in the observational environment (75.3%), and retains substantial agreement when the data come from two environments with unknown gene interventions. Order inference for a new table requires only a forward pass, allowing the same pretrained model to scale to this 499-gene setting. Appendix D.3 provides the setup and smaller-network results.

Summary. Across synthetic data, controlled component ablations, and two real-world benchmarks, our pre-trained TABORDER model recovers useful causal order information without order supervision. Its performance is robust to moderate departures from the training prior, while severe missingness and unobserved confounding remain important limitations.

5.2 Predictive Performance (Trade Off II)

We now evaluate the predictive performance of TABORDER on real-world data, measuring both imputation accuracy and downstream predictive performance. We use the OpenML CC18 classification and CTR23 regression benchmarks [Bischl et al., 2021, Fischer et al., 2023]. We consider the datasets with at least 80% real-valued inputs, yielding 33 CC18 and 23 CTR23 datasets, listed in Appendix C.2. We apply a split into 80% train and 20% test data as well as mask 10–50% of entries uniformly at random. We compare against standard imputers KNN, ICE, MICE [Van Buuren and Groothuis-Oudshoorn, 2011], MISSFOREST [Stekhoven and Bühlmann, 2012], neural methods (GAIN [Yoon et al., 2018]), and HYPERIMPUTE [Jarrett et al., 2022]; additional baselines are reported in Appendix D.4. For the trained TABORDER, we perform fine-tuning on the training data only. For downstream evaluation, we train an XGBOOST model [Chen and Guestrin, 2016]



(a) CTR23: Imputation (b) CTR23: Prediction (c) CC18: Imputation (d) CC18: Prediction

Figure 6: **Predictive Performance (Trade Off II)**. Shown is the average rank of imputation (a, c) and downstream prediction (b, d) performance on the CTR23 (left) and CC18 (right) datasets. TabOrder is the most accurate imputation method for $\geq 40\%$ missingness.

on imputed data (classification and regression), except that for CTR23 we directly impute missing targets with TabOrder at test time.

Figure 6 reports average ranks across missingness levels. TabOrder achieves the best imputation performance at higher missingness (CTR23 for $q \geq 30\%$, CC18 for $q \geq 40\%$), HyperImpute at lower missingness. For downstream tasks, TabOrder is competitive on CC18 and outperforms on CTR23 at $q \geq 40\%$, benefiting from directly imputing values without retraining.

Intervention-Robust Prediction. We first illustrate the effect in the chain SCM $X \to Y \to Z$ introduced in Figure 1, predicting the mediator Y from X and Z. Changing the mechanism of Z affects methods that use this downstream association, while an order-constrained prediction remains stable (Figure 7).

We then test the same idea on 50 independently sampled SCMs with eight variables each. For each SCM, we predict a mediator with both parents and descendants, and change one child mechanism at test time. Table 3 shows the trade-off: the standard tabular foundation models achieve lower error when the distribution is unchanged, whereas the learned TabOrder mask has a substantially smaller average increase under intervention. The oracle order is unchanged, as expected.

Unknown Biological Interventions. Last, to study the effects of interventions outside the synthetic setting, we revisit the real-world data by Sachs et al. [2005]. We take the causal order inferred for the near-observational dataset (`cd3cd28`) as a reference, and compare it to the orders inferred by TabOrder for two interventional datasets (`g0076`, `psitect`). We summarize the footprint of an intervention by counting how many pairwise relative positions involving each variable change from the

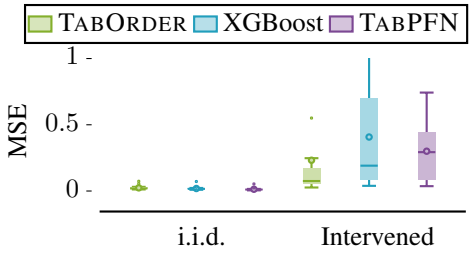
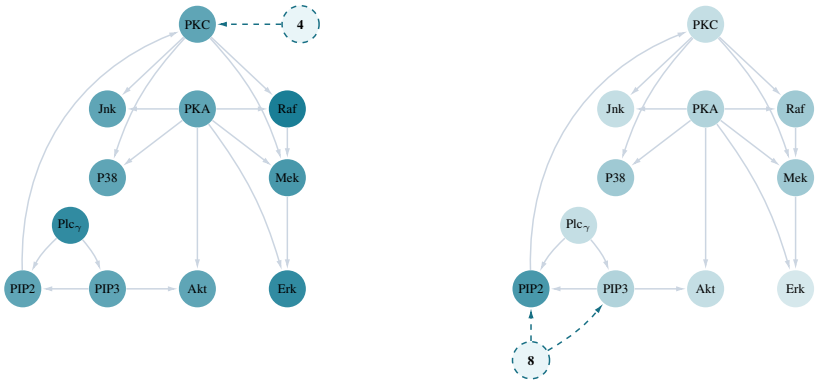


Figure 7: **Intervention-Robust Prediction.** MSE for predicting Y in the three-variable chain on i.i.d. and intervened samples. TabOrder remains accurate under the mechanism shift via its learned causal order, while TabPFN and XGBoost degrade.

Method	i.i.d.	Intervened	Δ
TabOrder, true order	0.604	**0.604**	**0.000**
TabOrder, learned order	0.599	**0.681**	**0.082**
TabPFN 2.5	**0.381**	0.862	0.482
TabICL v2	**0.383**	0.794	0.411
TabDPT	**0.383**	0.790	0.406
XGBoost	0.427	0.747	0.320

Table 3: **Prediction under a child-mechanism intervention.** Mean RMSE across 50 eight-variable SCMs. Lower Δ indicates greater stability from i.i.d. to intervened data.



(a) Interventional Condition `cd3cd28_g0076`. (b) Interventional Condition `cd3cd28_psitect`.

Figure 8: **Changes in Inferred Orders under Intervention.** In the Sachs et al. [2005] data, we compare two interventional conditions (a, b) to the reference condition, and count pairwise relative-order flips involving each node in the order by TABORDER (darker for more flips). An upstream intervention (a) leads to global changes in the order; a downstream one (b) to a local change of one target.

reference order (see Appendix E.3 for the full analysis). In Figure 8, darker nodes have more changed relative positions. The upstream PKC perturbation is associated with broad reorganization, while the PIP2/PIP3 perturbation has a more localized footprint. Such comparisons can help summarize intervention effects when their full system-wide consequences are not known in advance.

6 Conclusion

Towards addressing the disconnect between associative in-context learning and causal structure learning, we integrate causal order learning within a transformer-based framework for tabular data. Our architecture TABORDER jointly infers and uses causal orderings to constrain predictive attention. While the model is not limited to specific functional forms during training or inference, causal structure identification is not generally possible from observational data without further assumptions, such that we provided theoretical justification for our approach within the class of additive noise models. Our experiments cover both causal order discovery and predictive tasks, where TABORDER achieved reasonable performance across both dimensions. On real-world data, the inferred topological orderings align very closely with domain knowledge, and the discovered orders across interventional conditions give insights into the localized effects of the perturbations.

Limitations and Future Work. Regarding current limitations, the implementation focuses primarily on learning topological orderings rather than full directed acyclic graphs as we consider this sufficient for the predictive and interventional tasks under consideration. This is both a limitation and an upside of TABORDER: order ensures robustness to changes in non-causal dependencies, while attention implicitly takes over the task of identifying which variables are relevant. Specifying/estimating a full graph on the other hand is a much more difficult task and can thus incur additional error, both of which are bypassed by our approach. The refinement of these orders into sparse graph structures is well explored using pruning via sparse regression, but future work could address this aspect more directly within the transformer architecture. Furthermore, investigating broader training priors and systematically studying their effect on amortized learners is not the focus of this work, but an important consideration for improving the quality of pretraining. In this context, the use of explicitly interventional data during training is likely particularly promising given the aim to learn causal structure, although this requires access to richer training data where such interventions are present. These research efforts could better connect the areas of structure learning and predictive modeling toward more interpretable and intervention-robust in-context learning.

Acknowledgments and Disclosure of Funding

This work was supported by the CISPA Helmholtz Center for Information Security. Sascha Xu is a Ph.D. student at the Saarbrücken Graduate School for Computer Science. The authors declare no competing interests. The paper itself has many results that are naturally competing for the interest of the reader.

References

Vahid Meresht Balazadeh, Hamidreza Kamkari, Valentin Thomas, Junwei Ma, Bingru Li, Jesse Cresswell, and Rahul Krishnan. Causalpfn: Amortized causal effect estimation via in-context learning. *Advances in Neural Information Processing Systems*, 38:154945–154984, 2026.

Bernd Bischl, Giuseppe Casalicchio, Matthias Feurer, Pieter Gijsbers, Frank Hutter, Michel Lang, Rafael Gomes Mantovani, Jan N. van Rijn, and Joaquin Vanschoren. OpenML benchmarking suites. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021. URL https://openreview.net/forum?id=OCrD8ycKjG.

Patrick Blöbaum, Dominik Janzing, Takashi Washio, Shohei Shimizu, and Bernhard Schölkopf. Cause-effect inference by comparing regression errors. In *International Conference on Artificial Intelligence and Statistics*, pages 900–909. PMLR, 2018.

Peter Bühlmann, Jonas Peters, and Jan Ernest. Cam: Causal additive models, high-dimensional order search and penalized regression. *The Annals of Statistics*, 42(6):2526–2556, 2014.

Camillo Maria Caruso, Paolo Soda, and Valerio Guarrasi. Not another imputation method: A transformer-based model for missing values in tabular datasets. *arXiv preprint arXiv:2407.11540*, 2024.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939785. URL https://doi.org/10.1145/2939672.2939785.

Jacob Feitelberg, Dwaipayan Saha, Kyuseong Choi, Zaid Ahmad, Anish Agarwal, and Raaz Dwivedi. Tabimpute: Accurate and fast zero-shot missing-data imputation with a pre-trained transformer. *arXiv preprint arXiv:2510.02625*, 2025.

Sebastian Felix Fischer, Matthias Feurer, and Bernd Bischl. Openml-ctr23–a curated tabular regression benchmarking suite. In *AutoML Conference 2023 (Workshop)*, 2023.

Trevor Hastie, Rahul Mazumder, Jason D Lee, and Reza Zadeh. Matrix completion and low-rank svd via fast alternating least squares. *The Journal of Machine Learning Research*, 16(1):3367–3402, 2015.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. In *The Eleventh International Conference on Learning Representations*, 2023.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

Patrik Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. *Advances in neural information processing systems*, 21, 2008.

Noah Isakov and Amnon Altman. Protein kinase c in t cell activation. *Annual Review of Immunology*, 2002. URL https://api.semanticscholar.org/CorpusID:82352757.

Daniel Jarrett, Bogdan C Cebere, Tennison Liu, Alicia Curth, and Mihaela van der Schaar. Hyperimpute: Generalized iterative imputation with automatic model selection. In *International Conference on Machine Learning*, pages 9916–9937. PMLR, 2022.

Nan Rosemary Ke, Silvia Chiappa, Jane Wang, Anirudh Goyal, Jorg Bornschein, Melanie Rey, Theophane Weber, Matthew Botvinic, Michael Mozer, and Danilo Jimenez Rezende. Learning to induce causal structure. In *International Conference on Learning Representations*, 2023. URL `https://openreview.net/forum?id=hp_RwhKDJ5`.

Trent Kyono, Yao Zhang, Alexis Bellot, and Mihaela van der Schaar. Miracle: Causally-aware imputation via learning missing data mechanisms. *Advances in Neural Information Processing Systems*, 34:23806–23817, 2021.

Sébastien Lachapelle, Philippe Brouillard, Tristan Deleu, and Simon Lacoste-Julien. Gradient-based neural dag learning. In *International Conference on Learning Representations*, 2020.

David Lake, Sonia Corrêa, and Jurgen Muller. Negative feedback regulation of the erk1/2 mapk pathway. *Cellular and Molecular Life Sciences*, 73, 12 2016. doi: 10.1007/s00018-016-2297-8.

Lars Lorch, Scott Sussex, Jonas Rothfuss, Andreas Krause, and Bernhard Schölkopf. Amortized inference for causal structure learning. *Advances in Neural Information Processing Systems*, 35: 13104–13118, 2022.

Junwei Ma, Valentin Thomas, Rasa Hosseinzadeh, Alex Labach, Jesse Cresswell, Keyvan Golestan, Guangwei Yu, Anthony L Caterini, and Maks Volkovs. Tabdpt: Scaling tabular foundation models on real data. *Advances in Neural Information Processing Systems*, 38:172692–172722, 2026.

Pierre-Alexandre Mattei and Jes Frellsen. Miwae: Deep generative modelling and imputation of incomplete data sets. In *International conference on machine learning*, pages 4413–4423. PMLR, 2019.

Osman A Mian, Alexander Marx, and Jilles Vreeken. Discovering fully oriented causal networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8975–8982, 2021.

Francesco Montagna, Nicoletta Noceti, Lorenzo Rosasco, Kun Zhang, and Francesco Locatello. Scalable causal discovery with score matching. In *Conference on Causal Learning and Reasoning*, pages 752–771. PMLR, 2023a.

Francesco Montagna, Nicoletta Noceti, Lorenzo Rosasco, Kun Zhang, and Francesco Locatello. Causal discovery with score matching on additive models with arbitrary noise. In *Conference on Causal Learning and Reasoning*, pages 726–751. PMLR, 2023b.

Francesco Montagna et al. Demystifying amortized causal discovery with transformers. *Transactions on Machine Learning Research*, 2025.

Joris M Mooij, Sara Magliacane, and Tom Claassen. Joint causal inference from multiple contexts. *Journal of Machine Learning Research*, 21:1–108, 2020.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do bayesian inference. In *International Conference on Learning Representations*, 2022.

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2009.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. Tabicl: A tabular foundation model for in-context learning on large data. In *ICML 2025-Forty-Second International Conference on Machine Learning*, 2025.

Alexander Reisach, Myriam Tami, Christof Seiler, Antoine Chambaz, and Sebastian Weichwald. A scale-invariant sorting criterion to find a causal order in additive noise models. *Advances in Neural Information Processing Systems*, 36:785–807, 2023.

Jake Robertson, Arik Reuter, Siyuan Guo, Noah Hollmann, Frank Hutter, and Bernhard Schölkopf. Do-PFN: In-context learning for causal effect estimation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=0aNbl9b56B.

Paul Rolland, Volkan Cevher, Matthäus Kleindessner, Chris Russell, Dominik Janzing, Bernhard Schölkopf, and Francesco Locatello. Score matching enables causal discovery of nonlinear additive noise models. In *International Conference on Machine Learning*, pages 18741–18753. PMLR, 2022.

Karen Sachs, Omar Perez, Dana Pe'er, Douglas Lauffenburger, and Garry Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, pages 523–9, 2005.

Shohei Shimizu, Patrik O Hoyer, Aapo Hyvärinen, Antti Kerminen, and Michael Jordan. A linear non-gaussian acyclic model for causal discovery. *Journal of Machine Learning Research*, 7(10), 2006.

Peter Spirtes, Clark Glymour, and Richard Scheines. *Causation, prediction, and search*. MIT press, 2001.

Alexander Statnikov, Sisi Ma, Mikael Henaff, Nikita Lytkin, Efstratios Efstathiadis, Eric R. Peskin, and Constantin F. Aliferis. Ultra-scalable and efficient methods for hybrid observational and experimental local causal pathway discovery. *Journal of Machine Learning Research*, 16(100): 3219–3267, 2015.

Daniel J Stekhoven and Peter Bühlmann. Missforest—non-parametric missing value imputation for mixed-type data. *Bioinformatics*, 28(1):112–118, 2012.

Stef Van Buuren and Karin Groothuis-Oudshoorn. mice: Multivariate imputation by chained equations in r. *Journal of statistical software*, 45:1–67, 2011.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Igor Vivanco and Charles Sawyers. The phosphatidylinositol 3 kinase akt pathway in human cancer. *Nature reviews. Cancer*, 2:489–501, 08 2002. doi: 10.1038/nrc839.

Liang Wendong, Simon Buchholz, and Bernhard Schölkopf. Algorithmic causal structure emerging through compression. In *Causal Learning and Reasoning*, pages 201–242. PMLR, 2025.

Sascha Xu, Sarah Mameche, and Jilles Vreeken. Information-theoretic causal discovery in topological order. In *International Conference on Artificial Intelligence and Statistics*, pages 2008–2016. PMLR, 2025.

Jinsung Yoon, James Jordon, and Mihaela Schaar. Gain: Missing data imputation using generative adversarial nets. In *International conference on machine learning*, pages 5689–5698. PMLR, 2018.

Kun Zhang and Aapo Hyvärinen. On the identifiability of the post-nonlinear causal model. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, page 647–655, Arlington, Virginia, USA, 2009. AUAI Press. ISBN 9780974903958.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. Dags with no tears: Continuous optimization for structure learning. *Advances in neural information processing systems*, 31, 2018.

A Detailed Theoretical Results

A.1 Background and Assumptions

Our analysis builds on a set of standard assumptions commonly used in causal structure learning. We give a shortened overview below and refer to, e.g., Pearl [2009] for a detailed exposition.

(i) Causal Markov Condition. The joint distribution factorizes according to the true causal DAG $G^\star$, i.e., each variable X_i is conditionally independent of its non-descendants given its parents pa_i.

(ii) Faithfulness. All conditional independencies in the data distribution are entailed by the causal DAG. That is, there are no independencies arising from exact cancellations of parameters.

(iii) Causal Sufficiency. There are no unobserved confounders: all common causes of observed variables are included in the model.

(iv) Additive Noise Model (ANM). Data are generated according to a structural causal model of the form

$$X_i = \sum_{j \in pa_i} f_{ij}(X_j) + N_i, \tag{13}$$

where the noise variables N_i are jointly independent and the functions f_{ij} are nonlinear and sufficiently smooth.

Under these assumptions, causal directions become identifiable in many settings [Shimizu et al., 2006, Hoyer et al., 2008, Zhang and Hyvärinen, 2009, Bühlmann et al., 2014], and likelihood- or variance-based scores are consistent for recovering valid topological orders of the underlying DAG.

A.2 Identifiability of Causal Orders

Identifying the causal graph and thereby recovering a valid topological ordering is feasible for certain classes of functional mechanisms [Shimizu et al., 2006, Hoyer et al., 2008, Zhang and Hyvärinen, 2009]. Because our learning objective is based on minimizing conditional error variance, we focus on **causal additive models** [Bühlmann et al., 2014], where

$$X_i = \sum_{j \in pa_i} f_{ij}(X_j) + N_i, \tag{14}$$

with smooth functions f_{ij} and independent noise. Bühlmann et al. [2014] show that likelihood-/residual-variance–based scores consistently recover orders that respect the data-generating causal DAG. A simplified version of their main result reads as follows.

Theorem 2 (Order Consistency in Causal Additive Models; Bühlmann et al., 2014)**.** *In a causal additive model with non-linear functions and independent, non-zero noise, let $\hat{\pi}$ be the order minimizing residual variance. Then,*

$$\mathbb{P}(\hat{\pi} \in \Pi_0) \to 1 \quad \textit{as } n \to \infty, \tag{15}$$

where Π_0 denotes the set of all topological orders of the true causal DAG. Any order violating a causal edge incurs a strictly larger expected residual variance.

For this particular class of ANMs, likelihood-optimal orders are consistent with the topological orders of the causal DAG. This property is key for TABORDER: maximizing the likelihood naturally guides the optimization toward orders that match the underlying causal structure, because conditioning on the true parents yields the overall lowest-variance conditionals. In this way, the likelihood itself provides a causal inductive bias, steering the learned order toward compatibility with the generative process, without requiring any explicit supervision or information about the causal graph.

A.3 Formal Proof of Theorem 1

Theorem 1 (Effect of Missingness on Likelihood Asymmetry)**.** *Consider an additive noise model $Y = f(X) + N$ with $N \perp\!\!\!\perp X$ and $\mathrm{Var}(N) > 0$ and missingness indicator $M_X, M_Y \sim \mathrm{Bernoulli}(q)$ with $q \in (0, 1)$. Then, the difference in likelihood is amplified iff the forward signal-to-noise ratio $\mathrm{Var}(f(X))/\mathrm{Var}(N)$ exceeds the backward ratio $\mathrm{Var}(\mathbb{E}[X \mid Y])/\mathbb{E}[\mathrm{Var}(X \mid Y)]$, i.e.,*

$$\Delta_{\mathrm{inc}} > \Delta_{\mathrm{fix}} \quad \Longleftrightarrow \quad \frac{\mathrm{Var}(f(X))}{\mathrm{Var}(N)} > \frac{\mathrm{Var}(\mathbb{E}[X \mid Y])}{\mathbb{E}[\mathrm{Var}(X \mid Y)]}\,. \tag{12}$$

Proof. Fix a direction τ (either forward or backward). Let $v_{0,\tau}$ be the residual variance when the parent is observed ($M = 0$) and $v_{1,\tau}$ when the parent is missing ($M = 1$), with $\Pr(M = 1) = q$.

We first consider the log-likelihood under fixed variance versus missingness-aware variance. An optimal Gaussian conditional model that uses a single variance parameter must match the mask-averaged residual variance

$$\bar{\sigma}_\tau^2 = (1-q)\, v_{0,\tau} + q\, v_{1,\tau}, \tag{16}$$

hence achieves population log-likelihood

$$\mathcal{L}^\star_{\tau,\text{fix}} = -\tfrac{1}{2} \log \bar{\sigma}_\tau^2 + C, \tag{17}$$

for a constant C independent of τ. A model that distinguishes the two regimes fits them exactly and achieves

$$\mathcal{L}^\star_{\tau,\text{inc}} = -\tfrac{1}{2}\big[(1-q)\log v_{0,\tau} + q \log v_{1,\tau}\big] + C. \tag{18}$$

Likelihood Gain. We now characterize the improvement in log-likelihood when using missingness-aware variance. We define the (nonnegative) gain as

$$G_\tau := \mathcal{L}^\star_{\tau,\text{inc}} - \mathcal{L}^\star_{\tau,\text{fix}}. \tag{19}$$

By substituting the expressions above and canceling C, we obtain for the gain

$$G_\tau = -\tfrac{1}{2}\big[(1-q)\log v_{0,\tau} + q \log v_{1,\tau}\big] + \tfrac{1}{2}\log\big((1-q)\, v_{0,\tau} + q\, v_{1,\tau}\big).$$

Our objective is now to express the gain in terms of the variance ratio $R_\tau := v_{1,\tau}/v_{0,\tau}$ and the missingness rate q. We factor $v_{0,\tau}$ out of the logarithm as

$$(1-q)\, v_{0,\tau} + q\, v_{1,\tau} = v_{0,\tau}\big((1-q) + q\, R_\tau\big). \tag{20}$$

Therefore, we substitute for the logarithm of the single variance model

$$\log\big((1-q)\, v_{0,\tau} + q\, v_{1,\tau}\big) = \log v_{0,\tau} + \log\big((1-q) + qR_\tau\big). \tag{21}$$

By expanding $\log v_{1,\tau} = \log(v_{0,\tau} R_\tau) = \log v_{0,\tau} + \log R_\tau$ and plugging both into G_τ yields

$$\begin{aligned}
G_\tau &= -\tfrac{1}{2}\Big[(1-q)\log v_{0,\tau} + q\big(\log v_{0,\tau} + \log R_\tau\big)\Big] + \tfrac{1}{2}\Big[\log v_{0,\tau} + \log\big((1-q) + qR_\tau\big)\Big] && (22)\\
&= -\tfrac{1}{2}\Big[\big((1-q)+q\big)\log v_{0,\tau} + q \log R_\tau\Big] + \tfrac{1}{2}\Big[\log v_{0,\tau} + \log\big((1-q) + qR_\tau\big)\Big] && (23)\\
&= -\tfrac{1}{2}\Big[\log v_{0,\tau} + q \log R_\tau\Big] + \tfrac{1}{2}\Big[\log v_{0,\tau} + \log\big((1-q) + qR_\tau\big)\Big] && (24)\\
&= \tfrac{1}{2}\Big[\log\big((1-q) + qR_\tau\big) - q \log R_\tau\Big]. && (25)
\end{aligned}$$

Thus G_τ depends on $(v_{0,\tau}, v_{1,\tau})$ only through the ratio R_τ and the missingness rate q:

$$G_\tau = G(R_\tau, q) := \frac{1}{2}\Big[\log\big((1-q) + qR_\tau\big) - q \log R_\tau\Big]. \tag{26}$$

We now differentiate $G(R, q)$ with respect to R to establish monotonicity under masking rates of $q \in (0, 1)$

$$\frac{\partial G}{\partial R} = \frac{1}{2}\left(\frac{q}{(1-q) + qR} - \frac{q}{R}\right) = \frac{q(1-q)(R-1)}{2R\big((1-q) + qR\big)}. \tag{27}$$

If the ratio $R > 1$, then all terms in the numerator and denominator are positive, and therefore the derivative is positive, so $G(R, q)$ is strictly increasing in R on $(1, \infty)$, i.e.

$$\boxed{R_1 > R_2 > 1 \implies G(R_1, q) > G(R_2, q).} \tag{28}$$

The case when $R = 1$ corresponds to no gain, since then missingness does not affect the residual variance. This is only achievable in pathological cases (e.g., constant functions). We have now established general conditions under which the gain in one direction exceeds that in the other. We now move to provide explicit conditions in the ANM case.

Gain in Additive Noise Models. Let $\Delta_{\text{fix}} := \mathcal{L}^\star_{\text{fwd,fix}} - \mathcal{L}^\star_{\text{bwd,fix}}$ and $\Delta_{\text{inc}} := \mathcal{L}^\star_{\text{fwd,inc}} - \mathcal{L}^\star_{\text{bwd,inc}}$. Then

$$\Delta_{\text{inc}} - \Delta_{\text{fix}} = \big(\mathcal{L}^\star_{\text{fwd,inc}} - \mathcal{L}^\star_{\text{fwd,fix}}\big) - \big(\mathcal{L}^\star_{\text{bwd,inc}} - \mathcal{L}^\star_{\text{bwd,fix}}\big) = G(R_{\text{fwd}}, q) - G(R_{\text{bwd}}, q). \quad (29)$$

By strict monotonicity of $G(\cdot, q)$ on $R > 1$,

$$\Delta_{\text{inc}} > \Delta_{\text{fix}} \quad \Longleftrightarrow \quad R_{\text{fwd}} > R_{\text{bwd}}. \quad (30)$$

It remains to derive specific terms for R_{fwd} and R_{bwd} in ANMs.

Forward direction ($X \to Y$): if X is observed, the optimal residual is N, so $v_{0,\text{fwd}} = \text{Var}(N)$. If X is missing, the optimal predictor is the mean of Y, so $v_{1,\text{fwd}} = \text{Var}(Y) = \text{Var}(f(X)) + \text{Var}(N)$ (since $N \perp\!\!\!\perp X$). Thus

$$R_{\text{fwd}} = \frac{v_{1,\text{fwd}}}{v_{0,\text{fwd}}} = \frac{\text{Var}(f(X)) + \text{Var}(N)}{\text{Var}(N)} = 1 + \frac{\text{Var}(f(X))}{\text{Var}(N)}. \quad (31)$$

Backward direction ($Y \to X$): if Y is observed, the optimal residual variance is $v_{0,\text{bwd}} = \mathbb{E}[\text{Var}(X \mid Y)]$. If Y is missing, the best predictor is the mean of X, so $v_{1,\text{bwd}} = \text{Var}(X) = \mathbb{E}[\text{Var}(X \mid Y)] + \text{Var}(\mathbb{E}[X \mid Y])$. Hence

$$R_{\text{bwd}} = \frac{v_{1,\text{bwd}}}{v_{0,\text{bwd}}} = \frac{\mathbb{E}[\text{Var}(X \mid Y)] + \text{Var}(\mathbb{E}[X \mid Y])}{\mathbb{E}[\text{Var}(X \mid Y)]} = 1 + \frac{\text{Var}(\mathbb{E}[X \mid Y])}{\mathbb{E}[\text{Var}(X \mid Y)]}. \quad (32)$$

Therefore $R_{\text{fwd}} > R_{\text{bwd}}$ is equivalent to

$$1 + \frac{\text{Var}(f(X))}{\text{Var}(N)} > 1 + \frac{\text{Var}(\mathbb{E}[X \mid Y])}{\mathbb{E}[\text{Var}(X \mid Y)]} \quad \Longleftrightarrow \quad \frac{\text{Var}(f(X))}{\text{Var}(N)} > \frac{\text{Var}(\mathbb{E}[X \mid Y])}{\mathbb{E}[\text{Var}(X \mid Y)]}. \quad (33)$$

Using Eq. (30), we can thus conclude that the likelihood gap is amplified iff the forward signal-to-noise ratio exceeds the backward ratio, which proves the claim. □

B Model and Training Details

This section summarizes the model components and hyperparameters used in the experiments.

We implement the attention mask as an additive bias to the attention scores before applying the softmax. Forward, we use $(1 - A_{ij}) \cdot \beta$, where β is annealed from -5.0 to -20.0 during training; backward, we use the sigmoid form $\hat{A}_{ij} = \sigma((s_i - s_j)/t)$ where t is annealed from 1.0 to 0.1.

- We embed every cell to a vector of dimension $d_{\text{embedding}} = 128$.
- Transformer blocks: 6 repetitions for inducing order and 4 repetitions for cell prediction. All row- and column-attention modules use four heads. We use PyTorch's `nn.TransformerEncoderLayer` with GELU activations, feed-forward dimension of $2 \cdot d_{\text{embedding}}$, dropout 0.1 and prenorm.
- Optimization: Adam with base learning rate 2×10^{-4}, warmup ratio 0.03 and weight decay.
- Batch size 4 with 25000 training steps for a total of 100,000 datasets seen. Datasets are generated with $5 - 10$ variables each and $512 - 1024$ samples. Pretraining takes approximately 41 minutes on one NVIDIA A100 GPU with 40GB of memory.
- Masking during training: random masking with `masking_percentage`=0.2. For each dataset, the query fraction is sampled uniformly from $[0.05, 0.5]$.

C Detailed Experimental Setups

C.1 Synthetic Data Generation

For each synthetic example, we sample the number of variables $d \sim \text{Unif}_{\text{Int}}(5, 10)$ and construct a directed acyclic graph (DAG) $G = (V, E)$ over $V = \{1, \ldots, d\}$ by first drawing a random topological ordering and then sampling parent sets subject to an in-degree constraint $|pa_i| \in \{1, 2, 3\}$.

Given G, we generate observations according to a nonlinear structural causal model (SCM). For each node $i = 1, \ldots, d$, variables are generated as

$$X_i \;=\; f_i(X_{pa_i}) + \varepsilon_i,$$

where the noise terms are mutually independent with $\sigma_i \sim \mathrm{Unif}(0.05, 0.25)$ and $\varepsilon_i \sim \mathcal{N}(0, \sigma_i^2)$.

For root nodes ($pa_i = \varnothing$), we sample X_i from a mixture of simple base distributions: with probability $1/2$, a scaled Gaussian $X_i = sZ$ where $Z \sim \mathcal{N}(0,1)$ and $s \sim \mathrm{Unif}(0.5, 2.0)$; otherwise, a scaled uniform $X_i = (U - \frac{1}{2})a$ where $U \sim \mathrm{Unif}(0,1)$ and $a \sim \mathrm{Unif}(1.0, 5.0)$. Values are lightly clipped to avoid extreme outliers.

In the non-additive setting, each structural function f_i is a smooth nonlinear function of all parents jointly, sampled from an RBF Gaussian process prior approximated by random Fourier features (RFF). Let $x \in \mathbb{R}^P$ denote the standardized parent vector, where $P = |pa_i|$. We sample a lengthscale $\ell \sim \exp\big(\mathrm{Unif}(\log 0.3, \log 1.0)\big)$, and define

$$\phi(x) \;=\; \sqrt{\frac{2}{H}}\cos\big(W^\top x + b\big), \qquad W \in \mathbb{R}^{P\times H},\; b \in \mathbb{R}^H,$$

with $H = 256$ features, $W_{jk} \sim \mathcal{N}(0, \ell^{-2})$, and $b_k \sim \mathrm{Unif}(0, 2\pi)$. With weights $a \sim \mathcal{N}(0, I_H)$, we set $f_i(x) \;=\; a^\top \phi(x)$. This yields smooth nonlinear mechanisms that are explicitly non-additive across parents.

Finally, for each DAG we draw batches with sequence length $r \sim \mathrm{Unif}_{\mathrm{Int}}(512, 1024)$ from the resulting SCM.

C.2 Real-World Datasets

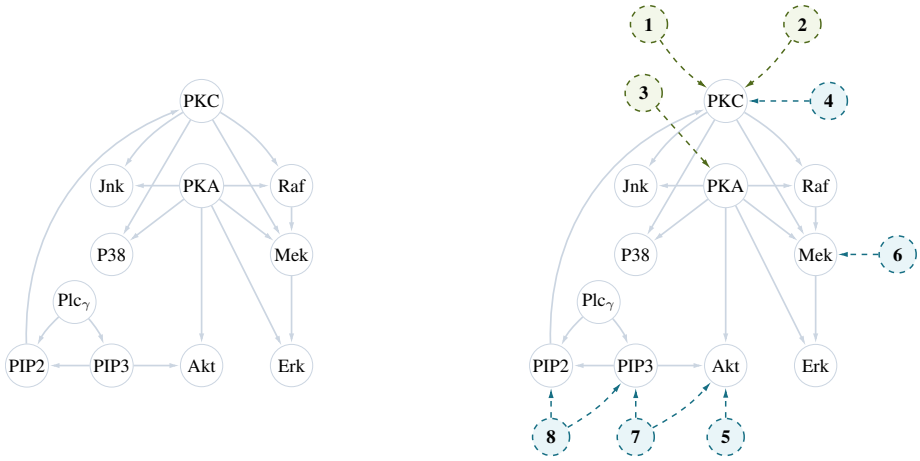
Sachs Datasets. We evaluate our method on the protein signaling dataset of Sachs et al. [2005]. The data consist of single-cell measurements of intracellular signaling proteins in primary human CD4$^+$ T cells, obtained via multiparameter flow cytometry.

The dataset contains measurements of 11 phosphoproteins and phospholipids (`Raf`, `Mek`, `Erk`, `Plcg`, `PIP2`, `PIP3`, `PKC`, `PKA`, `Akt`, `Jnk`, `P38`) under multiple experimental conditions, where as is common we disregard the condition `b2camp` and treat the condition `cd3cd28` as the main reference condition, shown in the main experiments (Figure 4b). The conditions correspond to different combinations of external stimulation (e.g., TCR/CD28 activation and ICAM-2 co-stimulation) and inhibition (e.g., of MEK, PKC, PI3K, and Akt, as well as perturbations of phosphoinositide signaling).

For the evaluations, we use a reduced Sachs signaling graph over the measured variables consistent with commonly used benchmark variants in prior causal discovery work [e.g., Zheng et al., 2018]. The used network closely follows the validated Sachs benchmark network distributed within the `bnlearn` R package, replacing the edges Erk $\rightarrow$ Akt and PKC$\rightarrow$ PKA with the biologically established reduced-pathway edges PIP3 $\rightarrow$ Akt and PIP2 $\rightarrow$PKC, both of which Sachs et al. [2005] identify as signaling influences. We show this graph in Figure 9, depicting the consensus graph we use in the experiments (a) and the activating resp. inhibiting perturbations (b). We show the correspondence of these interventions with the available experimental conditions in Table 4.

No.	Type	Condition	Intervention
1	Activator	`cd3cd28`	TCR/CD28 stimulation
2	Activator	`cd3cd28_icam2`	TCR/CD28 + ICAM-2
3	Activator	`pma`	PKA/PKC-side activation proxy
4	Inhibitor	`cd3cd28_g0076`	PKC inhibition
5	Inhibitor	`cd3cd28_aktinhib`	Akt inhibition
6	Inhibitor	`cd3cd28_u0126`	MEK inhibition
7	Inhibitor	`cd3cd28_ly`	PI3K/PIP3/Akt-axis inhibition
8	Inhibitor	`cd3cd28_psitect`	PIP2/PIP3-axis perturbation

Table 4: **Interventional Conditions in the Data by Sachs et al. [2005].** Listed is each interventional condition and the type of perturbation applied therein.



(a) **Causal DAG.** (b) **Intervention Targets.**

Figure 9: **Consensus Causal Structure for the Data by Sachs et al. [2005].** We show the causal DAG that we consider in the evaluations (a), as well as the effects of perturbations applied in each experimental condition (b). Dashed colored nodes correspond to activators (green) and inhibitors (blue), cf. Table 4.

REGED Datasets. We additionally evaluate on REGED, a gene-expression benchmark derived from human lung-cancer microarray data with known directed networks [Statnikov et al., 2015, Mian et al., 2021]. REGED0 is the unmanipulated environment; REGED1 and REGED2 contain gene manipulations that are not supplied to the model. We use three subnetworks with 5, 15, and 499 observed genes. The 499-gene analysis uses the induced subgraph available in the online benchmark, containing 823 directed edges.

CC18 and CTR23. For reference, we include statistics on the real-world datasets CC18 and CTR23 used in Figure 6 in Table 5.

Table 5: **Real-World Datasets for Predictive Tasks.** CC18 and CTR23 datasets used in our experiments.

suite	dataset	rows	cols	target_name
CC18	GesturePhaseSegmentationProcessed	9873	33	Phase
CC18	MiceProtein	552	78	class
CC18	analcatdata_authorship	841	71	Author
CC18	balance-scale	625	5	class
CC18	banknote-authentication	1372	5	Class
CC18	blood-transfusion-service-center	748	5	Class
CC18	breast-w	683	10	Class
CC18	churn	5000	17	class
CC18	climate-model-simulation-crashes	540	19	outcome
CC18	diabetes	768	9	class
CC18	electricity	45312	8	class
CC18	first-order-theorem-proving	6118	52	Class
CC18	ilpd	583	10	Class
CC18	jm1	10880	22	defects
CC18	jungle_chess_2pcs_raw_endgame_complete	44819	7	class
CC18	kc1	2109	22	defects
CC18	kc2	522	22	problems
CC18	mfeat-morphological	2000	7	class
CC18	ozone-level-8hr	2534	73	Class

Continued on next page

suite	dataset	rows	cols	target_name
CC18	pc1	1109	22	defects
CC18	pc3	1563	38	c
CC18	pc4	1458	38	c
CC18	phoneme	5404	6	Class
CC18	qsar-biodeg	1055	42	Class
CC18	satimage	6430	37	class
CC18	segment	2310	17	class
CC18	spambase	4601	58	class
CC18	steel-plates-fault	1941	28	target
CC18	vehicle	846	19	Class
CC18	vowel	990	11	Class
CC18	wall-robot-navigation	5456	25	Class
CC18	wdbc	569	31	Class
CC18	wilt	4839	6	class
CTR23	QSAR_fish_toxicity	908	7	LC50
CTR23	abalone	4177	8	rings
CTR23	airfoil_self_noise	1503	6	sound_pressure
CTR23	california_housing	20640	9	medianHouseValue
CTR23	cars	804	18	Price
CTR23	concrete_compressive_strength	1030	9	strength
CTR23	cpu_activity	8192	22	usr
CTR23	energy_efficiency	768	9	heating_load
CTR23	fifa	19178	28	wage_eur
CTR23	forest_fires	517	11	area
CTR23	geographical_origin_of_music	1059	117	latitude
CTR23	grid_stability	10000	13	stab
CTR23	kin8nm	8192	9	y
CTR23	kings_county	21613	18	price
CTR23	miami_housing	13932	16	SALE_PRC
CTR23	naval_propulsion_plant	11934	15	gt_compressor_decay_ state_coefficient
CTR23	physiochemical_protein	45730	10	RMSD
CTR23	pumadyn32nh	8192	33	thetadd6
CTR23	red_wine	1599	12	quality
CTR23	sarcos	48933	22	V22
CTR23	space_ga	3107	7	ln_votes_pop
CTR23	superconductivity	21263	82	critical_temp
CTR23	white_wine	4898	12	quality

D Extended Experiments

D.1 Causal Order Inference on Synthetic Data (Fig. 4a)

Our baselines for causal order discovery include topological sorting methods that use regression together with regularized likelihood or AMC-based scores, CAM [Bühlmann et al., 2014] and TOPIC [Xu et al., 2025], or perform sorting based on the score of the data distribution using score matching, namely SCORE [Rolland et al., 2022] and extensions DAS [Montagna et al., 2023a] and NOGAM [Montagna et al., 2023b], and finally the neural method NOTEARS [Zheng et al., 2018]. We use their publicly available implementations, among others included in the `causal-learn` resp. `do-discover` Python packages, using default parameter choices as suggested in the original implementations. A suite of methods, such as the classical PC algorithm [Spirtes et al., 2001], return only partially oriented structures (CPDAGs or PDAGs) that do not correspond to a unique causal order. Therefore, we omit them from the comparison. In the evaluation, we use batch sizes of $|B| = 4$ for $n_B = 30$ batches, resulting in 120 runs per method. All methods return a fully oriented causal DAG with the exception of AVICI, which outputs edge probabilities from which we extract a causal order by thresholding.

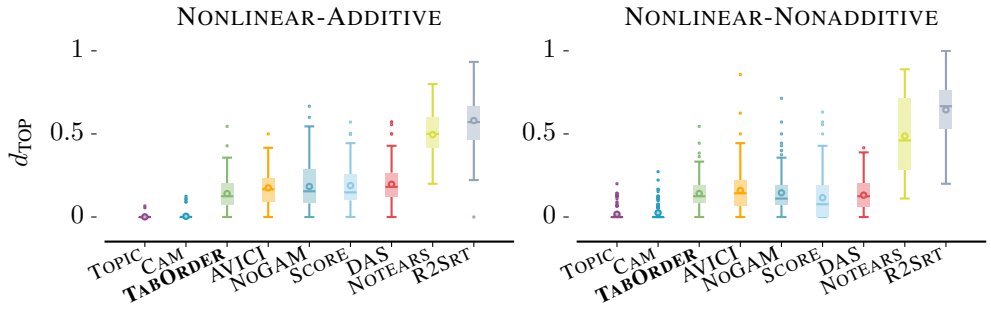


Figure 10: **Causal Order Inference (extends Fig. 4a).** Shown is the average topological divergence (d_{TOP}, lower is better) between ground truth and estimated causal orders on synthetic datasets, generated from a nonlinear additive noise model with functions drawn from a Gaussian process, with both a variant where functional mechanisms are additive (left), respectively non-additive (right), in the causal parents.

We assess the causal order learning using the topological divergence d_{TOP}, which measures the discrepancy between the predicted causal order $\hat{\pi}$ and the ground-truth order implied by the true DAG $G^\star$ by counting edge disagreements $d_{\text{TOP}}(\hat{\pi}, G^\star) = \sum_{i=1}^{d} \sum_{j:\pi(i)\geq\pi(j)} \mathbb{I}[(i,j) \in G^\star]$, with lower values indicating better alignment with the true causal ordering.

We plot the topological divergence in Figure 10, including both additive mechanisms as well as non-additive mechanisms.

D.2 Causal Order Inference on Real-World Data (Fig. 4b)

We evaluate causal order discovery on the Sachs protein-signaling benchmark [Sachs et al., 2005], using the consensus graph in Figure 9. We focus on the `cd3cd28` condition as the primary reference condition, since it corresponds to TCR/CD28 stimulation without additional inhibitors and is commonly used as the closest proxy to an observational setting. We compare TABORDER again with the order-based and graph-based causal discovery baselines CAM, TOPIC, SCORE, DAS, NOGAM, NOTEARS, R2SRT, and AVICI.

In addition to the reference condition reported in Figure 4b, the Sachs dataset contains multiple interventional conditions corresponding to targeted perturbations of signaling pathways. We therefore also evaluate TABORDER across all Sachs conditions, besides `b2camp` which we exclude from the analysis as in previous works [Mooij et al., 2020]. The full results are shown in the heatmap in Figure 11. Across most conditions, TABORDER obtains topological divergences in the range 0.12–0.29, indicating that the inferred orders remain broadly compatible with the reference signaling graph under moderate interventions. The main exception is the PKC inhibition condition `cd3cd28_g0076`, where the divergence is substantially larger. This is biologically plausible, since `PKC` acts as a hub in the reduced graph and its perturbation can induce widespread changes across downstream MAPK and stress-signaling branches.

Overall, these results suggest that TABORDER recovers meaningful causal order information not only on synthetic data, but also on a real biological benchmark with known intervention structure. In Section E.3, we further analyze how individual interventions affect the inferred orders through rank shifts and pairwise order flips.

D.3 Causal Order Inference on REGED

REGED provides a complementary transfer setting: its observations and mechanisms do not come from our synthetic prior, and the model is not told which genes are manipulated. We apply the pretrained model without fine-tuning. For the 499-gene subgraph, we infer an order from three disjoint 64-row subsets and average their score vectors before ranking the genes. Table 2 reports the resulting edge agreement in the observational environment and in both manipulated environments.

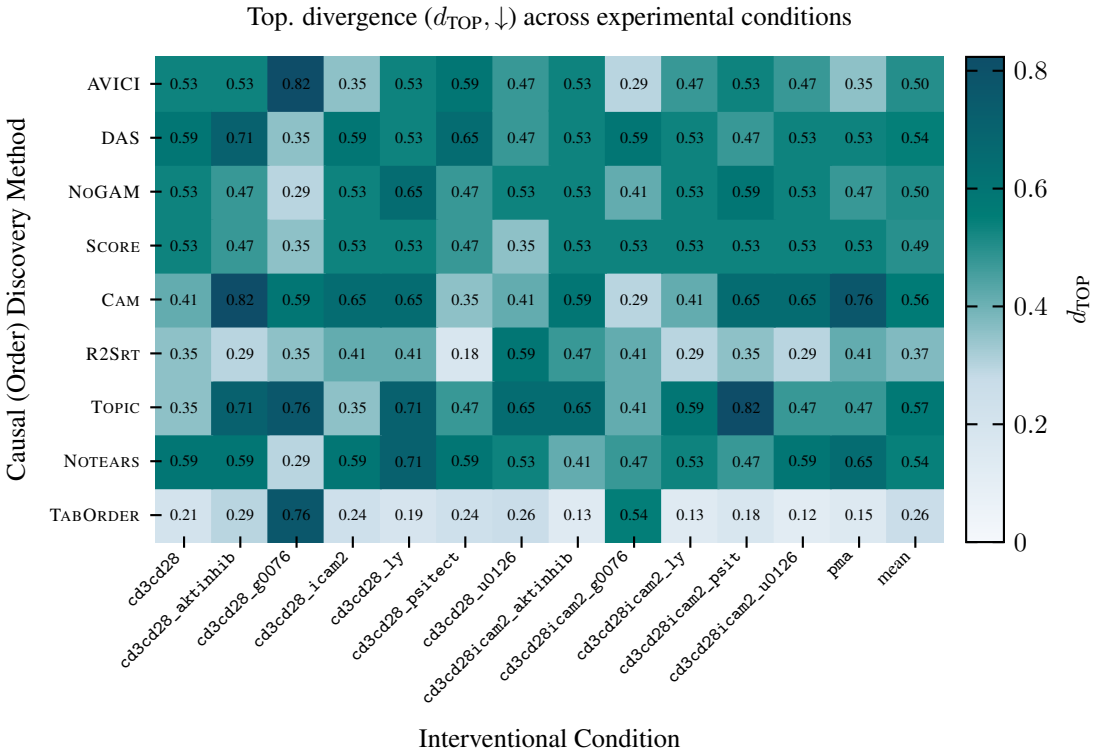


Figure 11: **Causal Order Inference in Real-World Data (extends Fig. 4b).** Shown is the average topological divergence (d_{TOP}, lower is better) between ground truth and estimated causal orders on the real-world causal discovery benchmark by Sachs et al. [2005]. For each available dataset c (horizontal) corresponding to one experimental condition, we show each method (vertical) and how the causal order $\hat{\pi}_c$ it discovers in dataset c differs from the consensus graph G_{c_0}. The consensus graph G_{c_0} represents a summary of causal relationships that likely best represents the near-observational condition `cd3cd28`. Figure 4b reports the corresponding bar-plot results for this condition.

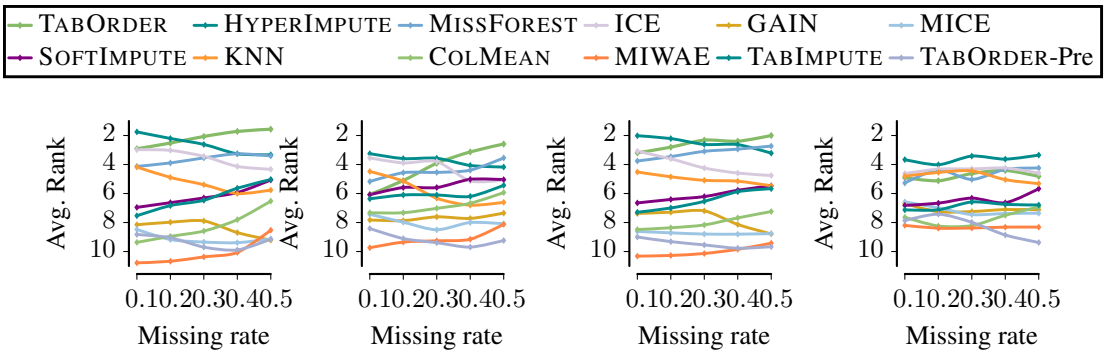
Environment	Genes	Correct edges	d_{TOP}
REGED0	499	620/823	0.247
REGED1	499	568/823	0.310
REGED2	499	547/823	0.335

Table 6: **REGED evaluation details.** The order is inferred separately in each environment; intervention targets are not provided to the model.

The 15-gene subnetwork gives 10 correctly ordered edges out of 14 in each environment ($d_{\text{TOP}} = 0.286$). The 5-gene subnetwork is inconclusive, ordering only one or two of its four edges correctly. These smaller results delineate the scope of the transfer result rather than being used to select a favorable subnetwork.

D.4 Imputation Experiments (Fig. 6)

We use a variety of baselines for missing value imputation: conventional methods such as column mean (ColMean), KNN, and SoftImpute [Hastie et al., 2015], the iterative approaches ICE, MICE [Van Buuren and Groothuis-Oudshoorn, 2011], and MissForest [Stekhoven and Bühlmann, 2012], neural baselines GAIN [Yoon et al., 2018] and MIWAE [Mattei and Frellsen, 2019], and HyperImpute [Jarrett et al., 2022], which automatically selects the best imputation model per feature. Lastly, we compare to TabImpute [Feitelberg et al., 2025], a tabular foundation model specifically designed for missing value imputation. Due to memory constraints we use maximum



(a) CTR23 Imputation Rank (b) CTR23 Prediction Rank (c) CC18 Imputation Rank (d) CC18 Prediction Rank

Figure 12: **Predictive Performance (extends Fig. 6).** Shown is the average rank of imputation (a, c) and downstream prediction (b, d) performance on the CTR23 and CC18 datasets of all tested methods.

batch sizes of 512 data points for TabImpute and limit the maximum number of variables to 15, as larger tables caused out-of-memory errors on our 40GB GPU hardware. We evaluate TabOrder, which is fine-tuned on the training set, as well as the pretrained TabOrder without fine-tuning (denoted TabOrder-Pre).

We plot the average rank of imputation and downstream prediction performance across all datasets and missingness rates in Figure 12. We observe that TabOrder is the most accurate imputation method for $\geq 40\%$ missingness, outperforming all baselines including TabImpute. TabOrder-Pre does not perform competitively, indicating that fine-tuning or alternatively training on different synthetic data is necessary to achieve strong imputation performance.

D.5 Intervention-Robust Prediction (Fig. 7)

For this experiment, we generate data from a three-variable chain $X \to Y \to Z$. X is sampled from a standard normal distribution. Y is generated from an additive noise model $Y = f(X) + N_Y$ where f is a GP sample with RBF kernel or spline function and $N_Y \sim \mathcal{N}(0, 0.1^2)$. Z is generated similarly as $Z = g(Y) + N_Z$ where g is an independent GP/spline and $N_Z \sim \mathcal{N}(0, 0.1^2)$. We generate training datasets with 5000 samples and test datasets with 2500 samples. We use the default configuration of TabPFN obtained via huggingface as well as the default XGBoost regressor with 100 estimators. We provide the full intervention MSE plots in Figure 13.

For the eight-variable benchmark in Table 3, we sample 50 SCMs with at most three parents per node (edge probability 0.34), independent Gaussian noise standard deviations in $[0.2, 0.5]$, and nonlinear additive mechanisms. We retain only graphs with a mediator target having both parents and descendants, use 512 context and 512 query samples per SCM, and replace one child mechanism at test time. We evaluate TabPFN v2.5 with its standard regressor configuration, TabICLv2 with eight estimators and key–value caching, TabDPT with eight ensemble members, and XGBoost with 400 estimators and maximum depth five. Thus, "TabPFN 2.5" in Table 3 denotes the v2.5 checkpoint/configuration used for this experiment.

E Additional Experiments

E.1 Robustness Outside the Training Assumptions

We additionally investigate the robustness of TabOrder to shifts away from the additive-noise and causal-sufficiency assumptions used in the theoretical analysis. Table 7 includes changes in functional mechanisms, value-dependent missingness, and unobserved confounding. At moderate shifts, order quality remains close to the in-prior reference. It degrades more clearly with severe MNAR masking and latent confounding, as expected when the causal assumptions are violated.

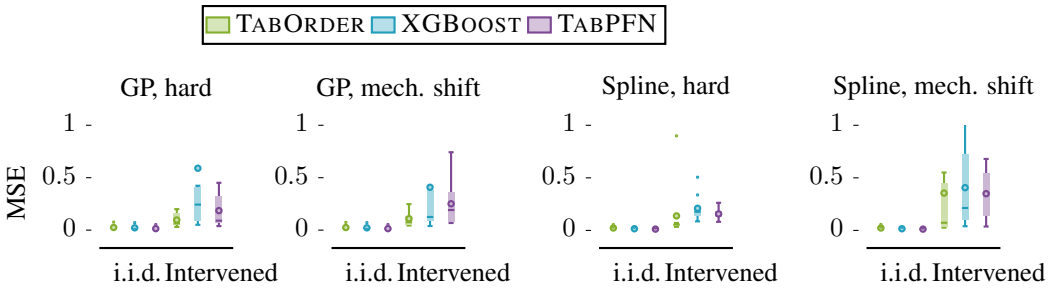


Figure 13: **Intervention-Robust Prediction (extends Fig.7).** Test MSE for predicting Y in the three-variable chain, split by mechanism family (GP vs. spline) and intervention type.

Setting	d_{TOP}	Δ
In-prior reference	0.181	0.000
Polynomial mechanisms	0.205	0.024
MNAR self-tail, 20%	0.202	0.021
MNAR self-high, 20%	0.212	0.031
MCAR, 50%	0.225	0.044
MNAR self-tail, 50%	0.286	0.105
MNAR self-high, 50%	0.404	0.223
Latent confounding	0.308	0.127

Table 7: **Order recovery outside the training assumptions.** Topological divergence (d_{TOP}, lower is better).

Besides standard additive Gaussian noise, we evaluate heteroskedastic noise, where the noise variance depends on the parent variables, and multiplicative noise, where noise scales the signal directly.

Table 8 reports the resulting topological divergence d_{TOP}. For illustration, we include both models trained on additive (CAM) functional models as well as non-additive ones. While the additive GP variant performs best under additive and heteroskedastic noise, the non-additive GP variant achieves the strongest performance under multiplicative noise and overall remains more stable across noise forms. These results suggest that TABORDER is reasonably robust to moderate deviations from the classical additive noise setting.

Method	Additive Noise	Heteroskedastic Noise	Multiplicative Noise
TABORDER (additive GP)	0.202	0.190	0.232
TABORDER (non-additive GP)	0.179	0.180	0.172

Table 8: **Non-Additive Noise.** Topological divergence (d_{TOP}) under different noise forms in the SCM.

Additional component results. Table 9 isolates the two architectural choices used by TABORDER on a held-out synthetic set; the ablation models were trained from three independent initializations (seeds 0, 1, and 2). Removing the missingness-aware variance increment worsens both likelihood and order recovery. The no-order model uses unrestricted attention, so we do not evaluate its learned scores as a causal order; its lower i.i.d. NLL instead reflects access to all associations. Under child interventions with MCAR, MAR, and MNAR masks, however, its NLL penalty is 0.175–0.237, compared with 0.041–0.056 for the full model. When the variance increment alone is disabled at test time, NLL increases from -0.234 to -0.034 (a $+0.200$ change, also shown in the table).

Model	i.i.d. NLL	d_{TOP}	Δ intervention NLL	Δ variance NLL
Full TABORDER	-0.025	0.179	0.041–0.056	–
Without variance increment	0.042	0.198	–	$+0.200$
Without order-constrained attention	-0.276	–	0.175–0.237	–

Table 9: **Architecture ablations.** I.i.d. NLL and d_{TOP} use a held-out 200-graph evaluation. Δ intervention NLL is the range across MCAR, MAR, and MNAR child-intervention settings; Δ variance NLL is the test-time increase when the missingness-aware variance increment is disabled (lower is better).

Feature-position control. Figure 5 replaces the causal order used to form TABORDER's attention mask; it does not permute the presented feature columns. As a complementary control, we consistently permuted the context and query columns for TabPFN, TabICLv2, and TabDPT on 30 eight-variable SCMs under oracle, corrupted, random, and reversed permutations. These TFMs were approximately invariant: for TabPFN, random and reversed permutations changed mean RMSE by -0.00010 and $+0.00010$ relative to the oracle permutation, and the change was unrelated to causal-order violations; TabICLv2 and TabDPT showed the same qualitative pattern.

E.2 Causal Order Inference: Scalability

We also demonstrate the scalability of TABORDER depending on (1) the number of features d and (2) sequence length of each batch n, otherwise using the standard data generating settings in Section C.1.

For each configuration, we generate 25 batches of size 4 (a total of 100 graphs) and report averaged metrics. We vary either the number of variables $d \in \{10, 20, 30, 40, 50\}$ at fixed sequence length $n = 1024$, or the sequence length $n \in \{128, 256, 512, 1024, 2048, 4096\}$ at fixed $d = 10$. All experiments were run on a single `NVIDIA A100` GPU (40GB VRAM) using `PyTorch` without distributed computation.

Figure 14 compares TABORDER against causal discovery baselines as the number of variables increases. We report topological divergence, evaluation runtime, and peak GPU memory usage. Some classical methods could not be evaluated beyond moderate dimensionalities due to computational limitations and are therefore omitted for larger d, in particular CAM for $d = 50$ (Figure 14 a).

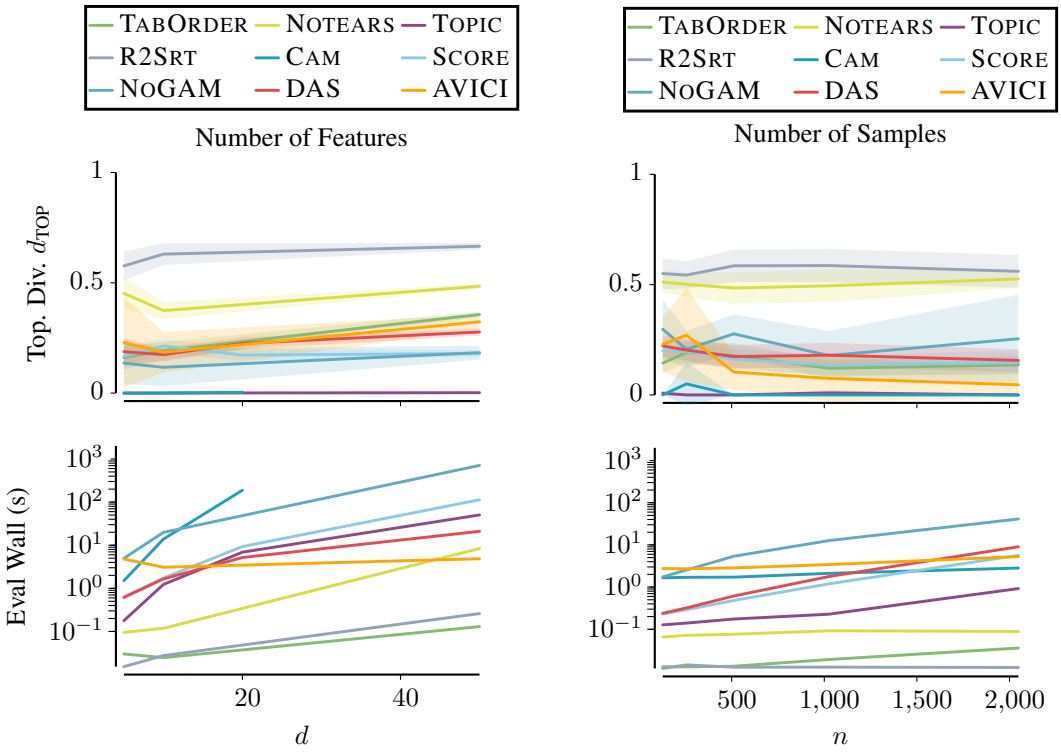
(a) **Scalability with Feature Size**. Shown are scalability metrics for TABORDER when varying the number of features d at fixed sample size $n = 1024$. Values are mean $\pm$ 95% CI.

d	Top. div. (d_{TOP})	Eval wall (s)	Peak mem (GB)
10	0.204096 ± 0.0179684	0.0458761 ± 0.00885146	$0.151067 \pm 0.0000251987$
20	0.236118 ± 0.0147771	0.0772007 ± 0.0137936	$0.293119 \pm 0.0000504157$
50	0.264468 ± 0.0105586	0.185035 ± 0.0112221	0.981129 ± 0.000124748
100	0.322675 ± 0.00883721	0.321426 ± 0.00937954	2.92858 ± 0.000259869
200	0.421040 ± 0.00577534	0.663079 ± 0.00304706	8.55309 ± 0.000527340

(b) **Scalability with Sample Size**. Shown are scalability metrics for TABORDER when varying the sample size n at fixed number of features $d = 10$. Values are mean $\pm$ 95% CI.

n	Top. div. (d_{TOP})	Eval wall (s)	Peak mem (GB)
128	0.254131 ± 0.0207305	0.0205390 ± 0.00128916	$0.0397114 \pm 0.00000329479$
256	0.229872 ± 0.0218224	$0.0196907 \pm 0.000812033$	$0.0560856 \pm 0.00000642391$
512	0.199187 ± 0.0206693	0.0268431 ± 0.00181145	$0.0891907 \pm 0.0000126822$
1024	0.204096 ± 0.0179684	0.0458761 ± 0.00885146	$0.151067 \pm 0.0000251987$
2048	0.197760 ± 0.0187264	0.149261 ± 0.00514696	$0.278104 \pm 0.0000855268$
4096	0.172316 ± 0.0184299	0.209578 ± 0.0120166	0.532027 ± 0.000107620

Table 10: **Scalability**. Shown are scalability metrics for TABORDER on causal order discovery. This experiment uses the same data generation and evaluation as in Figure 4 and varies the number of features respectively samples.



(a) Varying the number of features d at $n = 1024$.

(b) Varying the sequence length n at $d = 10$.

Figure 14: **Scalability (extends Fig. 4a).** Shown are scalability comparisons for TABORDER and baselines on causal order discovery. This experiment uses the same data generation and evaluation as in Figure 4 and varies the number of features d (left) and samples n (right). Shaded regions show 95% confidence intervals. For CAM, only $d <= 20$ is shown as the available implementation did not reasonably scale to $d = 50$.

We further assess the runtime and memory scaling of TABORDER to larger problem sizes where other methods time out. To that end, we scale d up to 200 and n up to 4096, and report the resulting topological divergence, evaluation runtime, and peak GPU memory usage in Table 10a and Table 10b. Computationally, runtime and memory usage scale predictably with problem size. Increasing the number of variables d has a substantially stronger effect on runtime and memory consumption than increasing the sequence length n. Despite being trained primarily on graphs with 5–10 variables, TABORDER generalizes to substantially larger settings up to $d = 200$ variables while maintaining moderate topological divergence.

Order quality degrades gradually as d increases, reflecting the increasing difficulty of predicting globally consistent orders in high-dimensional settings. In contrast, the topological divergence remains comparatively stable across sequence lengths and slightly improves for larger n, suggesting that the model is not strongly data-limited in this regime and can extract sufficient causal signal from relatively short sequences.

Inference of the causal order itself remains near-instant once the forward pass is completed.

E.3 Real-World Data: Analysis of Biological Intervention Effects

We additionally analyze how the causal orders inferred by TABORDER change across the interventional conditions of the Sachs signaling benchmark [Sachs et al., 2005]. Since the dataset contains multiple experimentally perturbed signaling conditions with known intervention targets, it provides a useful setting to study whether learned orderings reflect biologically meaningful system reorganization under intervention.

Setup. For each experimental condition c, we infer a topological ordering $\hat{\pi}_c$ over the 11 signaling variables and compare it to the reference condition $c_0 = \texttt{cd3cd28}$, which is commonly treated as the closest approximation to an observational baseline. The discovered orders and their topological divergences relative to the consensus graph are summarized in Table 11a and Table 11b. The evaluated conditions and intervention targets are summarized in Table 4, while the underlying consensus signaling graph is shown in Figure 9.

Metrics. Beyond the global topological divergence d_{TOP}, we analyze two local measures of intervention-dependent reorganization.

For each node X, we compute the rank shift as

$$\Delta_{\texttt{c}}(X) = \text{rank}(\hat{\pi}_{\texttt{c}}, X) - \text{rank}(\hat{\pi}_{\texttt{cd3cd28}}, X),$$

where positive (negative) values indicate downstream (upstream) movement in the inferred order.

We additionally compute pairwise order flips, i.e., changes in the relative ordering between node pairs across conditions. For each node, we aggregate the number of flipped pairwise relations relative to the baseline and normalize by $(p-1)$ to obtain a fraction in $[0, 1]$.

Figure 15 summarizes rank shifts and pairwise flips across the considered experimental conditions. Figure 16 visualizes pairwise flips on the consensus signaling graph, while Figure 17 shows some representative examples of how the inferred orders themselves change relative to the baseline condition.

Analysis. We observe clear intervention-dependent reorganizations in the inferred orders. MEK inhibition (`cd3cd28_u0126`) induces strong changes involving `Mek`, `Erk`, and `Raf`, consistent with perturbation of the MAPK pathway and known feedback within the Raf–MEK–ERK cascade [Lake et al., 2016]. PKC inhibition (`cd3cd28_g0076`) produces the strongest global reorganization across the signaling network, which is biologically plausible given the central role of `PKC` as a signaling hub [Isakov and Altman, 2002]. In contrast, phosphoinositide perturbation (`cd3cd28_psitect`) yields comparatively localized changes centered around `PIP2` and nearby signaling components.

Other interventions induce weaker or more diffuse effects. PI3K inhibition (`cd3cd28_ly`) produces only moderate changes involving `PIP3` and `Akt`, while Akt inhibition (`cd3cd28_aktinhib`) affects a broader set of MAPK-related nodes rather than remaining localized to the Akt pathway itself [Vivanco and Sawyers, 2002].

Overall, the inferred orders reflect biologically meaningful intervention-dependent reorganizations of the signaling system, suggesting that TABORDER captures pathway-specific changes beyond purely observational dependencies.

E.4 Order Reliability Diagnostics

Without a reference graph, order quality cannot be certified from observational data alone. We therefore evaluate a heuristic acceptance rule that uses no ground-truth structural information. It requires all of: (i) a favorable NLL margin relative to reversed, random, and corrupted orders; (ii) pairwise-order agreement across random half-samples; and (iii) agreement with three independently trained checkpoints. We calibrate each threshold as the 10th percentile on 60 independently generated in-prior reference datasets.

The rule accepts $45/60$ in-prior datasets but only $3/60$ shifted datasets, where mechanisms, noise, and dimensionality differ from the training prior. Across the diagnostic evaluation, accepted orders are more accurate than the unfiltered set ($d_{\text{TOP}} = 0.174$ versus 0.293). These diagnostics can flag unreliable extrapolation, but do not establish causal correctness without external knowledge or interventions.

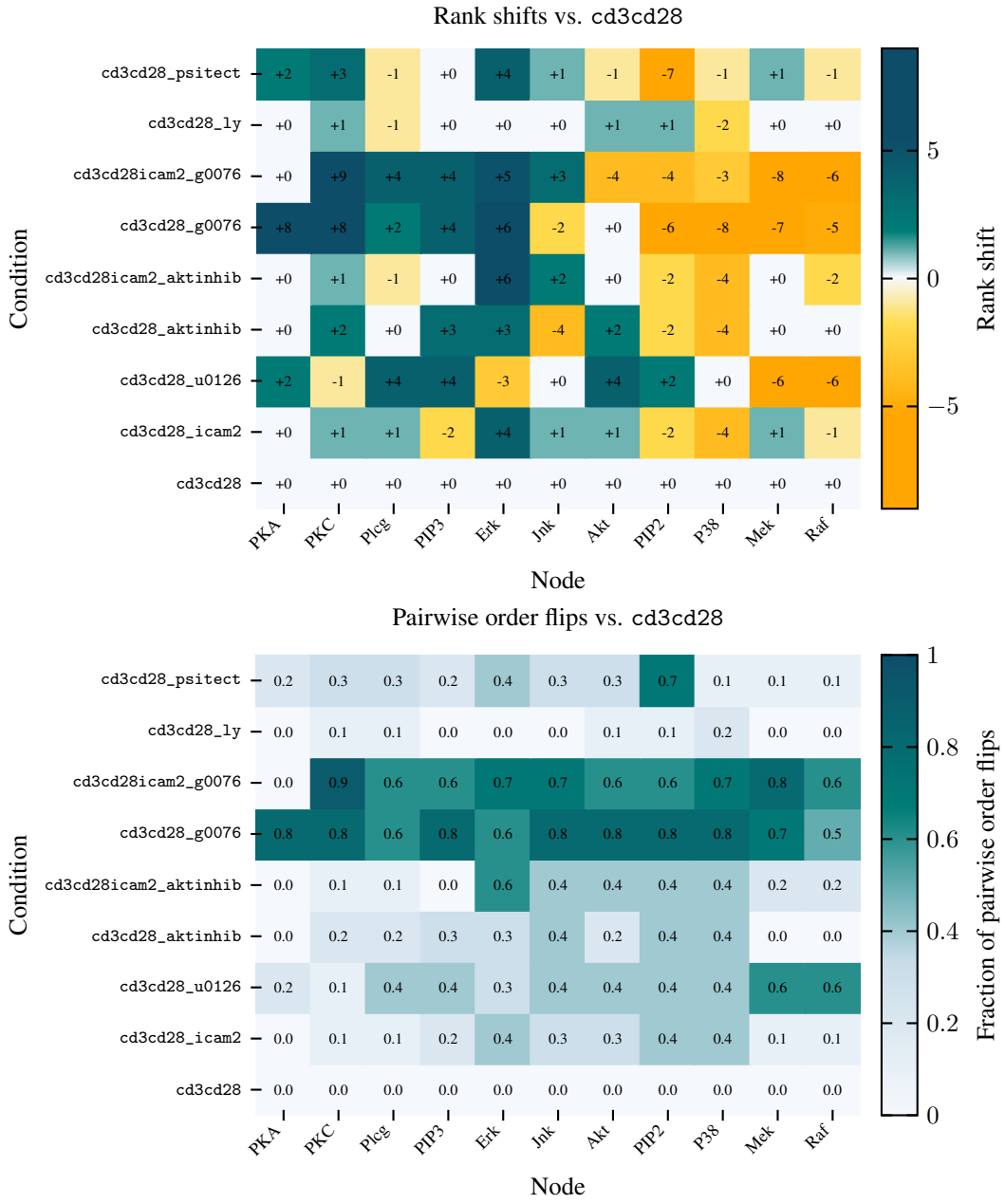
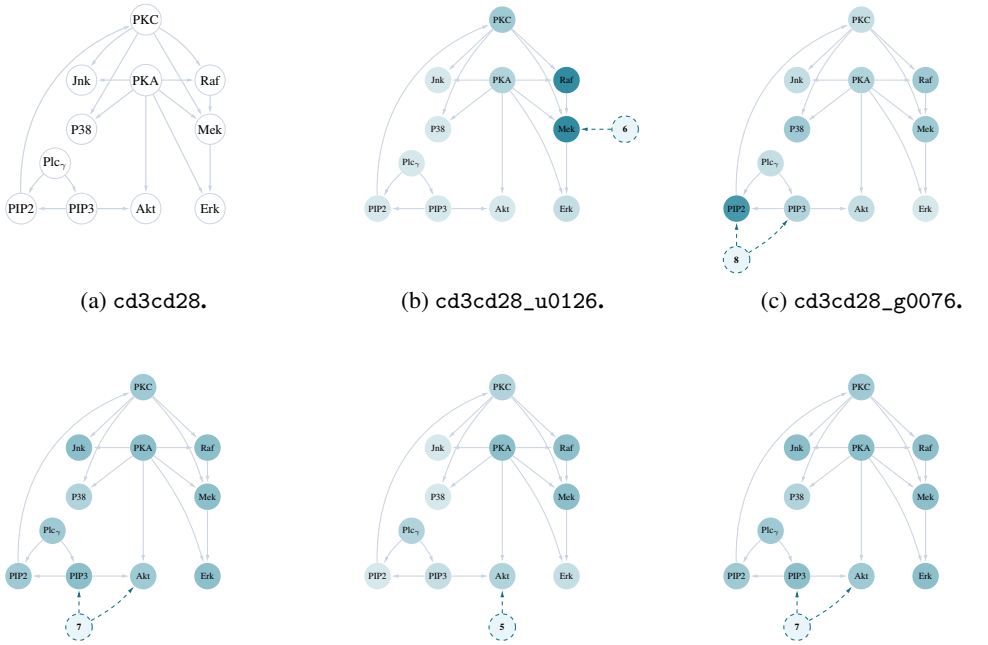


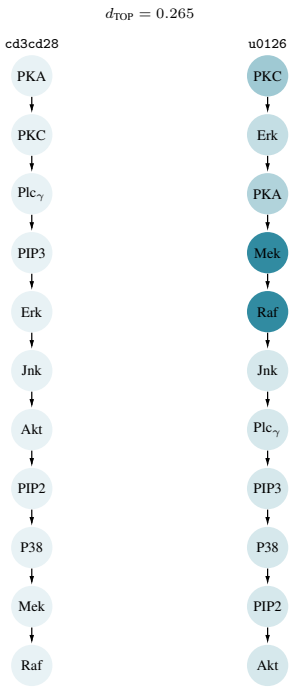
Figure 15: **Changes in Inferred Orders under Intervention.** The heatmaps compare the topological orders inferred by TABORDER across all selected Sachs experimental conditions relative to the reference condition `cd3cd28`. Top: node-wise rank shifts, where positive (negative) values indicate downstream (upstream) movement in the inferred order compared to the baseline condition. Bottom: fraction of pairwise order flips involving each node, measuring how often its relative ordering with other variables changes compared to the baseline. Rows correspond to experimental conditions and columns to signaling variables. Strong localized changes are observed for targeted pathway perturbations such as MEK inhibition (`cd3cd28_u0126`) and phosphoinositide perturbation (`cd3cd28_psitect`), while PKC inhibition (`cd3cd28_g0076`) induces broader global reorganization across the signaling network.



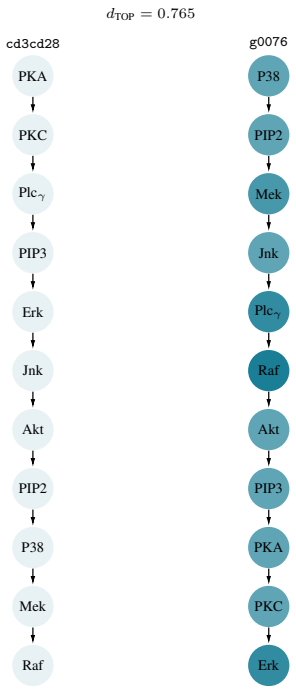
(a) `cd3cd28`. (b) `cd3cd28_u0126`. (c) `cd3cd28_g0076`.

(d) `cd3cd28_psitect`. (e) `cd3cd28_ly`. (f) `cd3cd28_aktinhib`.

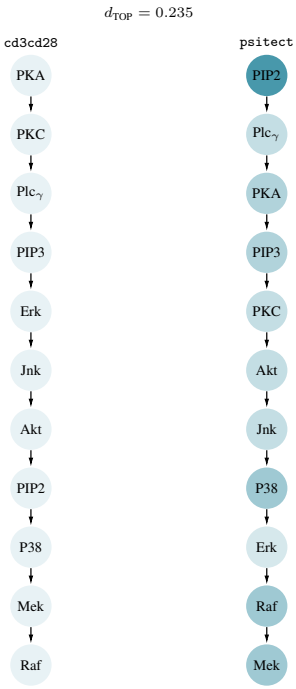
Figure 16: **Changes in Inferred Orders under Intervention (extends Fig. 8).** Nodes are colored by pairwise flip fraction relative to `cd3cd28`; edges show the fixed consensus graph. Activator and inhibitor markers indicate the corresponding experimental perturbations. Panel titles report the global topological divergence d_{TOP} for TABORDER where available.



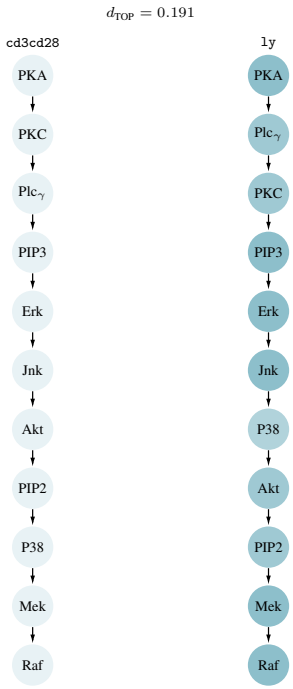
(a) u0126 (MEK inhibition).



(b) g0076 (PKC inhibition).



(c) psitect (Phosphoinositide perturbation).



(d) ly (PI3K inhibition).

Figure 17: **Example Changes in Inferred Orders.** Each panel compares the baseline order inferred by TABORDER on cd3cd28 to the order inferred under the intervention. Nodes in the baseline order are shown in light blue; nodes in the interventional condition are colored in dark blue depending on their pairwise flip fraction relative to the baseline.

(a) **Discovered Topological Orders per Condition.** Listed are the topological orders $\hat{\pi}_c$ discovered in c by TABORDER in each experimental condition c.

Condition c	Topological order $\hat{\pi}_c$
`cd3cd28`	PKA $\prec$ PKC $\prec$ Plcg $\prec$ PIP3 $\prec$ Erk $\prec$ Jnk $\prec$ Akt $\prec$ PIP2 $\prec$ P38 $\prec$ Mek $\prec$ Raf
`cd3cd28_icam2`	PKA $\prec$ PIP3 $\prec$ PKC $\prec$ Plcg $\prec$ P38 $\prec$ PIP2 $\prec$ Jnk $\prec$ Akt $\prec$ Erk $\prec$ Raf $\prec$ Mek
`cd3cd28_u0126`	PKC $\prec$ Erk $\prec$ PKA $\prec$ Mek $\prec$ Raf $\prec$ Jnk $\prec$ Plcg $\prec$ PIP3 $\prec$ P38 $\prec$ PIP2 $\prec$ Akt
`cd3cd28_aktinhib`	PKA $\prec$ Jnk $\prec$ Plcg $\prec$ PKC $\prec$ P38 $\prec$ PIP2 $\prec$ PIP3 $\prec$ Erk $\prec$ Akt $\prec$ Mek $\prec$ Raf
`cd3cd28icam2_aktinhib`	PKA $\prec$ Plcg $\prec$ PKC $\prec$ PIP3 $\prec$ P38 $\prec$ PIP2 $\prec$ Akt $\prec$ Jnk $\prec$ Raf $\prec$ Mek $\prec$ Erk
`cd3cd28_g0076`	P38 $\prec$ PIP2 $\prec$ Mek $\prec$ Jnk $\prec$ Plcg $\prec$ Raf $\prec$ Akt $\prec$ PIP3 $\prec$ PKA $\prec$ PKC $\prec$ Erk
`cd3cd28icam2_g0076`	PKA $\prec$ Mek $\prec$ Akt $\prec$ PIP2 $\prec$ Raf $\prec$ P38 $\prec$ Plcg $\prec$ PIP3 $\prec$ Jnk $\prec$ Erk $\prec$ PKC
`cd3cd28_ly`	PKA $\prec$ Plcg $\prec$ PKC $\prec$ PIP3 $\prec$ Erk $\prec$ Jnk $\prec$ P38 $\prec$ Akt $\prec$ PIP2 $\prec$ Mek $\prec$ Raf
`cd3cd28_psitect`	PIP2 $\prec$ Plcg $\prec$ PKA $\prec$ PIP3 $\prec$ PKC $\prec$ Akt $\prec$ Jnk $\prec$ P38 $\prec$ Erk $\prec$ Raf $\prec$ Mek

(b) **Discovered Topological Orders per Condition vs. Consensus Graph.** Edge Viol. and Top. Div. are the edge violation count and topological divergence (d_{TOP}), respectively, for the order $\hat{\pi}_c$ discovered by TABORDER, computed with respect to the consensus graph G_{c_0}.

Condition c	Edge viol.	d_{TOP}
`cd3cd28`	3	0.206
`cd3cd28_icam2`	3	0.235
`cd3cd28_u0126`	4	0.265
`cd3cd28_aktinhib`	5	0.294
`cd3cd28icam2_aktinhib`	1	0.132
`cd3cd28_g0076`	13	0.765
`cd3cd28icam2_g0076`	8	0.544
`cd3cd28_ly`	3	0.191
`cd3cd28_psitect`	3	0.235

Table 11: **Real-World Causal Order Inference.** Listed are the topological orders discovered by TABORDER in each experimental condition in the data by Sachs et al. [2005], as well as how they compare to the consensus network.

F Discussion

F.1 Limitations

Further limitations include that TABORDER currently supports only real-valued features, and our theoretical guarantees apply primarily to settings such as additive noise models. Extending the approach to mixed-type data, and establishing corresponding theoretical properties, presents an important direction for future work. Our method also focuses on learning causal orderings rather than full causal graphs. Extending the framework to learn sparser or more fine-grained causal structures is an interesting future direction but may require additional inductive biases or assumptions.

In practice, we observe that TABORDER pretrained solely on synthetic data does not yet match the performance of classical imputation methods, and fine-tuning on real-world datasets is required to achieve competitive results. Our current synthetic priors are best suited for generating mechanisms with known identifiability properties. Developing more realistic priors to improve pretraining quality is a particularly promising avenue for future research.

F.2 Broader impacts

TABORDER advances the state-of-the-art in tabular foundation models by enabling them to infer and integrate causal structure directly from data without structural supervision. This principled approach can improve robustness to distribution shifts and interventions, but it also introduces risks when variable orderings are misidentified or causal assumptions are incomplete. The structural inferences produced by TABORDER are hypotheses rather than ground truth and therefore require further validation, especially in high-stakes applications. As with other causal discovery methods, we recommend that users carefully assess the validity of the inferred structure, incorporate domain expertise, and perform appropriate sensitivity analyses before deploying TABORDER in critical settings.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction list our contributions and the claims accurately reflect the scope.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The main limitations of the work are explicitly mentioned in a dedicated section in the Conclusion.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The assumptions of the main theorem are listed and a complete proof provided in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We include source code that allows reproducing the main experimental results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Code is included in the supplementary material and will be made public under open access if accepted. The main experimental results are reproducible with that code.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The main training and test details are specified or included in the source code.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The main experimental results come with error bars.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The compute resources are described in the supplementary material.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The work proposes a new architecture and we see it as foundational research not specifically tied to particular applications or with a direct impact to negative applications that needs to be pointed out.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks and released pretrained models do not have a high risk of misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All code used to run the baselines was publically available online and we credit the owners by including citations and code links.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The code/model is documented.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The core method development does not involve LLMs for important or original components, and the use of LLMs is limited to standard components, boilerplate code and manuscript editing.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.